**UNIVERSIDAD AUTONOMA GABRIEL RENÉ MORENO**

*********FACULTAD DE INGENIERÍA EN CIENCIAS DE LA COMPUTACIÓN Y TELECOMUNICACIONES***

**Integrantes:** Juan Jose Alanes Peres 2009023892

**Materia:** Taller de Grado I

**Grupo:** SC

**Fecha:** /11/13

**SANTA CRUZ – BOLIVIA**

Tabla de contenido

[**Índice de Figuras** 3](#_Toc448696642)

[Introducción 5](#_Toc448696643)

[**CAPÍTULO 1 - Problema de investigación** 6](#_Toc448696644)

[**1.** **Definición Del Problema** 6](#_Toc448696645)

[**1.1.** **Situación Problemática** 7](#_Toc448696646)

[**1.2.** **Situación Deseada** 7](#_Toc448696647)

[**2.** **Objetivos** 7](#_Toc448696648)

[**2.1.** **Objetivo General** 7](#_Toc448696649)

[**2.2.** **Objetivos Específicos** 7](#_Toc448696650)

[**3.** **Metodología** 8](#_Toc448696651)

[ **Fase de Inicio:** 8](#_Toc448696652)

[ **Fase de elaboración:** 8](#_Toc448696653)

[ **Fase de Construcción:** 9](#_Toc448696654)

[**4.** **Alcance** 9](#_Toc448696655)

[**4.1.** **Requisitos Funcionales y no funcionales:** 9](#_Toc448696660)

[**CAPÍTULO 2 - La Organización** 10](#_Toc448696661)

[**5.** **Establecimientos Visitados:** 10](#_Toc448696662)

[**5.1.** **Hospital Universitario de Veterinaria (HUV)** 10](#_Toc448696663)

[**5.1.1.** **Misión** 11](#_Toc448696664)

[**5.1.2.** **Visión** 11](#_Toc448696665)

[**5.1.3.** **Estructura Organizacional** 11](#_Toc448696666)

[**5.2.** **Veterinaria PIRAÍ** 13](#_Toc448696667)

[**5.2.1.** **Estructura Organizacional** 14](#_Toc448696668)

[**5.2.2.** **Políticas de negocio más importantes.** 14](#_Toc448696670)

[**CAPÍTULO 3–Envió de mensajes de texto y correos electrónicos** 22](#_Toc448696671)

[6. Envió de Notificaciones 22](#_Toc448696672)

[**6.1.** **Vía Correo Electrónico** 22](#_Toc448696673)

[**6.1.1.** **JavaMail.** 22](#_Toc448696674)

[**6.2.** **Mensajes de Texto** 27](#_Toc448696675)

[**6.2.1.** **Servicio SMS usando SMPP 3.4** 27](#_Toc448696676)

[FASE DE INICIO. 29](#_Toc448696677)

[CAPÍTULO 4 - Requisitos 29](#_Toc448696678)

[**7.** **Captura de Requisitos** 29](#_Toc448696679)

[**7.1.** **Descripción.** 29](#_Toc448696684)

[**7.2.** **Identificar actores y casos de uso.** 29](#_Toc448696685)

[**7.3.** **Priorización de casos de uso** 62](#_Toc448696686)

[**7.4.** **Diagrama general de casos de uso.** 64](#_Toc448696687)

[**FASE DE ELABORACION** 65](#_Toc448696690)

[**CAPÍTULO 5 - Análisis** 65](#_Toc448696691)

[8. Descripción: 65](#_Toc448696698)

[**8.1.** **Análisis Paquetes.** 65](#_Toc448696699)

[**8.1.1.** **Identificación de Paquetes.** 65](#_Toc448696700)

[**8.2.** **Análisis de la arquitectura.** 66](#_Toc448696701)

[**8.2.1.** **Vista de paquetes.** 66](#_Toc448696702)

[**8.3.** **Análisis de Casos de Uso.** 69](#_Toc448696703)

[**8.3.1.** **Diagramas de Comunicación.** 69](#_Toc448696704)

[CAPÍTULO 6 - Diseño 82](#_Toc448696705)

[8.1. Diseño conceptual de la Base de Datos. 82](#_Toc448696706)

[8.2. Diseño Lógico de la Base de Datos. 84](#_Toc448696707)

[**8.2.1.** **Mapeo.** 84](#_Toc448696708)

[**8.3.** **Diseño Físico de la Base de Datos.** 85](#_Toc448696709)

[**8.3.1.** **Tablas de Volumen.** 85](#_Toc448696710)

[**8.4.** **Diagrama de Clases** 90](#_Toc448696711)

[90](#_Toc448696712)

[91](#_Toc448696713)

[93](#_Toc448696714)

[**9.** **Bibliografía** 96](#_Toc448696715)

**Índice de Figuras**

[Figura 1 Estructura organizacional del Hospital Universitario de Veterinaria. Fuente: (Dr. Emilio Arce, 2013) 12](#_Toc367232707)

[Figura 2 Estructura organizacional de la veterinaria Santa Cruz. Fuente: (Dr. Marcelino Banegas, 2013) 14](#_Toc367232708)

[Figura 3 Proceso de registro de propietario, otorgado por Dr. Marcelino Banegas 15](file:///M:\Taller\Amelunge_V3.docx#_Toc367232709)

[Figura 4 Proceso de registrar paciente, otorgado por Dr. Marcelino Banegas 16](file:///M:\Taller\Amelunge_V3.docx#_Toc367232710)

[Figura 5 Proceso de realizar consulta clínica, otorgado por Dr. Marcelino Banegas Director general de la veterinaria Piraí. 17](file:///M:\Taller\Amelunge_V3.docx#_Toc367232711)

[Figura 6 Proceso de registro de veterinario, otorgado por Dr. Marcelino Banegas 18](file:///M:\Taller\Amelunge_V3.docx#_Toc367232712)

[Figura 7 Proceso de realizar un tratamiento, otorgado por Dr. Marcelino Banegas Director general de la veterinaria Piraí. 19](file:///M:\Taller\Amelunge_V3.docx#_Toc367232713)

[Figura 8 Proceso de registro de vacunación otorgado por Dr. Marcelino Banegas 20](file:///M:\Taller\Amelunge_V3.docx#_Toc367232714)

[Figura 9 Proceso de registro de Desparasitación otorgado por Dr. Marcelino Banegas Director general de la veterinaria Piraí. 21](file:///M:\Taller\Amelunge_V3.docx#_Toc367232715)

# 

# Introducción

En la actualidad existe un sin número de clínicas veterinarias distribuidas por toda la ciudad, de las cuales muchas de ellas no cuentan con un software que les permita automatizar sus funciones, esto se debe a que muchas de estas clínicas veterinarias no cuentan con suficientes recursos monetarios para comprar un software adecuado a sus políticas de negocios. La falta de un software de gestión en estas clínicas puede ocasionar un esfuerzo innecesario en el personal de la clínica, además de una lenta atención a los pacientes cuando se trata de revisar su historial clínico o registrar a un cliente, lo cual implica un costo monetario adicional para clínica veterinaria.

Por otro lado muchas personas descuidan a sus mascotas debido al tiempo que se invierte en acudir a la clínica veterinaria por lo que se debe encontrar una forma de facilitar al propietario de la mascota informarse sobre el estado de su mascota.

La solución a los problemas mencionados anteriormente seria realizar un software capaz de adecuarse a la variedad de especies y razas de animales que se atienden actualmente para cada clínica veterinaria y de esta manera el software podría ser adquirido por más de una clínica veterinaria a la vez, resultando de esta manera más accesible a aquellas clínicas veterinarias que no pueden costearse un software exclusivo para la variedad de especies atendidas.

El software mencionado anteriormente también deberá ser flexible ante los posibles crecimiento que se presenten en un futuro referentes a la atención de nuevas especies dentro de la clínica veterinaria, además de contar con un modulo que permita el envío de mensajes ya sea al correo electrónico y/o al celular del propietario de la mascota para mantenerlo informado de el estado de su mascota, de su tratamiento o realizar alguna consulta a la base de datos, con el fin de facilitarle a dicho cliente la interacción con la clínica veterinaria para ganar afinidad con los clientes potenciales. Dicho software se describirá en el presente documento.

**CAPÍTULO 1 - Problema de investigación**

1. **Definición Del Problema**

En la clínica veterinaria Piraí es muy frecuente la pérdida de clientes al no poder otorgar facilidades al momento de realizar consultas simples, por lo que dichos potenciales clientes optan por no acudir a la clínica ya que no disponen del tiempo para hacerlo. Además de que al no estar automatizados los procesos de la clínica, no están en la capacidad de brindar un servicio con la calidad requerida, por lo que se incurre en una pérdida de tiempo, costo y esfuerzo para la propia clínica veterinaria.

A lo anterior se suma el descuido de los clientes de no cumplir con fechas programadas para la realización de controles de vacunación, desparasitaciones y tratamientos en general.

* 1. **Situación Problemática**

Actualmente los clientes asisten a la clínica veterinaria Piraí, para realizar cualquier tipo de consultas, incluso para obtener información básica como los servicios que se prestan, además que se manejan los historiales de forma manual, con un tiempo considerable de atención por cliente, lo que en muchas ocasiones provoca la aglomeración de personas en busca de atención, y por tal motivo, los clientes esperan demasiado tiempo para ser atendidos.

* 1. **Situación Deseada**

Tener un mecanismo de interacción con el cliente, que le permita realizar consultas clínicas desde cualquier lugar evitándole tener que asistir a la clínica para hacerlo de forma personal. Además brindar al cliente notificaciones o recordatorios clínicos en cualquier momento sin que éste tenga que comunicarse o apersonarse a la clínica para averiguar o consultar acerca de ese tipo de información.

1. **Objetivos**
   1. **Objetivo General**

Desarrollar un software de gestión de clínicas veterinarias que permita la recepción, procesamiento y respuesta de consultas de clientes de la clínica a través de correos electrónicos (Email) y a través de SMS's.

* 1. **Objetivos Específicos**
* Recopilar información de la clínica, y definir los requerimientos del sistema.
* Realizar el análisis de los requerimientos definidos, con el propósito de entender el problema y describir de manera completa lo que el sistema debe hacer.
* Realizar el diseño del sistema con el objetivo de describir como deberá ser implementado el sistema, satisfaciendo los requisitos del mismo.
* Realizar la implementación del software de acuerdo al diseño realizado con las funciones bien establecidas en el lenguaje de programación JAVA.
* Realizar pruebas con el objetivo de encontrar errores.

1. **Metodología**

Se utilizará como estrategia y como métodos al Proceso Unificado de Desarrollo de Software(PUDS), **(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**(Jacobson, 2000)**.

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**
* Entrevistas al personal de la clínica veterinaria para recabar información acerca de la forma en la que se realizan las actividades.
* Identificar y describir actores y casos de uso.
* Priorización de casos de uso.
* Estructuración de 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 paquetes.
* Análisis de casos de uso.
* Análisis de la arquitectura.

**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. **Alcance**


5. 1. **Requisitos Funcionales y no funcionales:**

Se contara con los siguientes módulos y funciones:

* Registro.

Modulo encargado del registro, modificación y dar de baja o abandono de los actores que participan en la gestión de la clínica veterinaria.

* + Gestionar Propietario.
  + Gestionar Paciente.
  + Gestionar Personal.
* Datos Clínicos.

Modulo encargado de registrar y actualizar toda la información clínica que se requiere en la clínica veterinaria.

* + Gestionar Vacunación.
  + Gestionar Desparasitación.
  + Gestionar Cirugías.
  + Gestionar Internación.
  + Gestionar Diagnostico.
  + Gestionar Consultas clínicas.
  + Gestionar tratamiento.
* Usuarios.

Modulo encargado de adicionar, modificar y eliminar usuarios de un determinado grupo, así como también definir los privilegios de cada grupo de usuario.

* + Gestión de Usuarios.
  + Gestión de grupos de usuario.
* Seguridad.

Modulo encargado de mantener coherente la base de datos de la clínica veterinaria en caso de contingencias, realizando copias de seguridad y restauración.

* + Backup de la base de datos.
  + Restauración de la base de datos.
* Consultas.

Modulo que permite realizar consultas a la base de datos según se lo necesite.

* + - * + Consultar Paciente.
        + Consultar Desparasitación.
        + Consultar Cirugía.
        + Consultar Diagnostico.
        + Consultar Vacunación.
        + Consultar Tratamiento.
        + Consultar Propietario.
* Servicios adicionales.

Modulo que contiene funcionalidades adicionales para el usuario.

* + Gestionar Especies.
  + Consultar Especie.
  + Gestión Raza.
  + Consultar Raza.
  + Enviar notificaciones por correo electrónico.
  + Procesar Consulta por Correo electrónico.
  + Envío personal de mensajes al celular de los propietarios.
  + Envío automático de mensajes al celular de los propietarios.

**CAPÍTULO 2 - La Organización**

1. **Establecimientos Visitados:**
   1. **Hospital Universitario de Veterinaria (HUV)**

Según información brindada por el **Dr. Emilio Arce, (2013)**, el Hospital Universitario de Veterinaria (HUV) se encuentra ubicado en los Módulos universitarios entre av. Bush y av. Centenario del 2do anillo, teléfono 3542035.

El Hospital Universitario de Veterinaria dependiente de la facultad de Ciencias Veterinarias U.A.G.R.M , fue creado el 12 de junio de 1960, por gestiones del Decano Dr Jorge Vaca Pereira, con el objetivo de apoyar al Proceso Académico, la Investigación y Extensión en sanidad Animal, a nivel del área urbana y rural. Su funcionamiento empezó, en la Av. Ejército Nacional, con el nombre de Hospital Animal. En 1970, junto con la facultad de Medicina, Veterinaria se trasladó a su actual ubicación en el campus universitario, con el nombre de Hospital para Animales. A partir de 1999 lleva el nombre de Hospital Universitario de Veterinaria (HUV).

El HUV es una unidad pedagógica científica, desde su formación hasta la fecha ha ido cumpliendo sus objetivos por los que fue creado y se han visto en la necesidad de ir evolucionando paralelamente al progreso de la región, es así que en las diferentes administraciones se ha preocupado para modernizar, mejorar y ampliar sus servicios.

En la facultad se ha construido la infraestructura del nuevo Hospital, con una inversión de alrededor de 4 millones de bolivianos, situación que a corto plazo permitirá convertirse en uno de los hospitales más modernos de Sudamérica, no solo en la infraestructura, sino en la implementación de técnicas de diagnóstico con alto nivel tecnológico, que permitirá mejorar la efectividad y calidad de sus objetivos y así satisfacer las exigencias de la sociedad actual.

1. * 1. **Misión**

Contribuir al desarrollo armónico de la Medicina Veterinaria y Zootécnica, mediante la formación de profesionales Médicos Veterinarios Zootecnistas, de alta calidad con espíritu ético, científico y humanístico; desarrollando la interacción social e investigación aplicada a la realidad sociocultural y productiva para mejorar la calidad de vida de la población.

* + 1. **Visión**

El Hospital Universitario de Veterinaria se visualiza en un centro referencial departamental, nacional e internacional de medicina integral de alta calidad, que aporta al desarrollo sostenible de la ciencia animal, en sus potencialidades y vocaciones, mediante procesos de formación profesional, interacción social e investigación.

* + 1. **Estructura Organizacional**

Figura 1Estructura organizacional del Hospital Universitario de Veterinaria.  
Fuente: (Dr. Emilio Arce, 2013)

**Servicios que se ofrecen**

Los costos por los servicios que se presta son accesibles, efectivos con profesionalismo.

* **Consultorio externo.-** Consultas, tratamientos y vacunaciones. En animales domésticos y silvestres (de compañía).
* **Quirófanos.-** Intervenciones quirúrgicas en general, cirugías estéticas reparadoras.
* **Farmacia.-** Medicamentos y biológicos en general.
* **Rayos X.-** Estudios radiológicos.
* **Observación de rabia.-** Para control de animales sospechosos o mordedores.
* **Internado.-** Control y tratamiento de especies menores.
* **Servicios de Cremación.-** Cremación de animales y residuos en general.

**LAISVET (LACLIVET)**

Tiene como objetivos: realizar investigación aplicada para pre y post-grado. Mejorar la capacidad de diagnóstico de las enfermedades de animales en Bolivia. Mejorar la capacidad de los docentes de la Facultad de Ciencias Veterinarias, y de otros veterinarios de laboratorios estatales y privados. Mejorar la formación de los veterinarios clínicos. Ordenar los protocolos de los laboratorios de Bolivia y conformar una red nacional y regional de información. Sus líneas de acción son:

**Líneas de investigación**

En Parasitología, Microbiología, Virología, Caracterización antígena y molecular de microorganismo causantes de enfermedades en los animales y utilización de la biología molecular en la genética de bovinos y otras especies animales.

**Laboratorios de:** Histopatología, Microbiología, Parasitología, PROVETSUR (ELISA, PCR, Inmunoflorescencia, Virología y Serología), Patología clínica: (Química Sanguínea, Examen de Orina, Hematología, Citológicos, Dermatológicos y otros).

**Próximos Servicios**

* Diagnósticos a través de la técnica de ELISA, PCR e Inmunofluorescencia.
* Clínica y cirugía de animales mayores.
* Peluquería y preparación de mascotas.
* Otras especialidades; Endoscopia y Traumatología.
* Activación y fortalecimiento de la clínica ambulatoria.
* Fortalecimiento a la atención de animales silvestres (de compañía).

***Fuente: Dr. Emilio Arce. Director general del Hosp. Universitario de Veterinaria.***

* 1. **Veterinaria PIRAÍ**

Según información brindada por el **Dr. Marcelino Banegas, (2013)**, la veterinaria Piraí se encuentra ubicada en la Av. Piraí esq. Ibérica, U.V. 52, Mz. 47. Teléfono 3550609.

La veterinaria Piraí funciona en nuestro departamento desde el año 1997, a cargo del doctor Marcelino Banegas, quien actualmente continúa en el cargo de director y propietario de la veterinaria. Actualmente trabaja conjuntamente con su esposa, quien ocupa el puesto de secretaria en la clínica.

La veterinaria presta sus servicios a animales domésticos especializándose a especies caninas y felinas.

La clínica brinda los siguientes servicios a la comunidad:

* + Consultas clínicas.
  + Tratamientos.
  + Internación.
  + Cirugías.
  + Prevención de rabia.
  + Aseo y peluquería.
  1. 1. **Estructura Organizacional**

Figura 2Estructura organizacional de la veterinaria Santa Cruz.  
Fuente: (Dr. Marcelino Banegas, 2013)

* + 2. **Políticas de negocio más importantes.**

****

1. **Diagrama de actividades Organizado en calles. Registrar Propietario.**

Figura 3Proceso de registro de propietario, otorgado por Dr. Marcelino Banegas

Director general de la veterinaria Piraí.

1. **Registrar Paciente.**

Figura 4Proceso de registrar paciente, otorgado por Dr. Marcelino Banegas

***Director general de la veterinaria Piraí.***

1. **Realizar Consulta Clínica**

Figura 5Proceso de realizar consulta clínica, otorgado por Dr. Marcelino Banegas Director general de la veterinaria Piraí.

1. **Gestionar Veterinario.**

Figura 6Proceso de registro de veterinario, otorgado por Dr. Marcelino Banegas

Director general de la veterinaria Piraí.

1. **Gestionar tratamiento.**

Figura 7 Proceso de realizar un tratamiento, otorgado por Dr. Marcelino Banegas Director general de la veterinaria Piraí.

1. **Gestiona Vacunación.**

Figura 8Proceso de registro de vacunación otorgado por Dr. Marcelino Banegas

Director general de la veterinaria Piraí.

1. **Gestiona Desparasitación.**

Figura 9 Proceso de registro de Desparasitación otorgado por Dr. Marcelino Banegas Director general de la veterinaria Piraí.

**CAPÍTULO 3–Envió de mensajes de texto y correos electrónicos**

# Envió de Notificaciones

* 1. **Vía Correo Electrónico**

### **JavaMail.**

El API JavaMail proporciona un marco independiente de la plataforma y el protocolo independiente de la construcción de aplicaciones de correo y mensajería. El API JavaMail está disponible como un paquete opcional para su uso con la plataforma Java SE y también se incluye en la plataforma Java EE, **(Oracle, 2013)**.

**Rubén Gómez,(2008)** define a JavaMail como una API opcional a la versión estándar de Java (J2SDK) que proporciona funcionalidades de correo electrónico, a través del paquete javax.mail permite realizar desde tareas básicas como enviar, leer y componer mensajes, hasta tareas más sofisticadas como manejar gran variedad de formatos de mensajes y datos, y definir protocolos de acceso y transporte. También afirman que aunque a primera vista pueda parecer que su utilidad se orienta a construir clientes de correo-e de tipo Outlook, ThunderBird, etc., su aplicación se puede generalizar a cualquier programa Java que necesite enviar y/o recibir mensajes, como por ejemplo, aplicaciones de intranets, páginas JSP, etc.

* + - 1. **Envío de correo electrónico.**

Según **Rubén Gómez, (2008),** para enviar un correo electrónico se deben seguir los siguientes pasos:

* Crear una sesión a la que le pasan las propiedades de la conexión, entre las que se ha debido incluir el servidor SMTP que se quiere usar.
* Crear un correo electrónico y rellenar sus campos:
  + Asunto.
  + Texto del mensaje.
  + Emisor.
  + Destinatario.
* Enviar el correo electrónico.

Los pasos anteriores se describirán de forma más extensa a continuación:

1. **Establecer la conexión.**

Primero se debe indicar el servidor SMTP que se va a usar para enviar el correo. Esto se hace añadiendo una nueva propiedad a las propiedades del sistema, que indicará que se quiere usar un servidor SMTP e identificará su correspondiente valor, es decir su dirección.

String smtpHost = "smtp.auna.com";

Propertiesprops = System.getProperties();

props.put("mail.smtp.host",smtpHost);

Las principales propiedades de una conexión a e-mail son las siguientes:

* **Mail debug.-**Hace que el programa se ejecute en modo depurador, es decir, muestra por pantalla todas las operaciones que realiza y los posibles errores. Por defecto no está activa. P.ej. props.put (“mail.debug”,”true”)
* **Mail from.-** Añade la dirección del remitente en el mensaje. P.ej. props.put (“mail.from”,”Ignacio@gmail.com”)
* **Mail.*protocolo.*host.-** Direccion del protocolo para el servidor para el protocolo especificado, es decir, el que se va a usar para enviar o recibir. P.ej. props.put (“mail.smtp.host”,”smtp.ono.com”)
* **Mail.*protocolo.*port.-** Numero de puerto del servidor que hay que usar. P.ej. props.put (“mail.pop3.port”,”995”)
* **Mail.*protocolo.*auth.-** Indica que vamos a autenticarnos en el servidor. P.ej. props.put (“mail.smtp.auth”,”true”)
* **Mail.*protocolo.*user.-** Nombre de usuario con el qué conectarse al servidor. P.ej. props.put (“mail.pop3.user”,”pruebasmail2013”).

1. **Crear el mensaje.**

Luego de haber especificado el servidor SMTP y una vez creada la sesión de e-mail, ya se puede proceder a la creación del mensaje que se quiere enviar.

La creación del mensaje se especifica de la siguiente manera:

Message mensaje = new MimeMessage(sesion);

Seguido de esto se debe llenar los atributos de la clase como se muestra a continuación:

* **VoidsetFrom(Addressdirection)**Establece como dirección del remitente la que pasa como parámetro.
* **VoidsetSubject(String Asunto)**Establecer como asunto el String que se pasa como parámetro.
* **VoidsetContent(Objectobj, Stringtype)**Establece como contenido del mensaje del tipo que le indicamos. Puede ser “*text/plain*” si el mensaje se rellena con un Sting sencillo. “*text/html*” si se hace con el formato HTML, o “*multipart/alternative*”, “*multipart/related*”, “*multipart/mixed*”si el mensaje se rellena con un objeto Multipart.
* **VoidsetContent(Multipartmp)**Rellena el mensaje con el Multipart que se le pasa y cuyo tipo MIME habrá sido establecido al crearlo.
* **VoidsetText(String Texto)**Es equivalente a usar el método setContent insertando un String e indicando que se trata de un texto plano (“*text/plain*”)

1. **Enviar el mensaje.**

Ya creado el mensaje se procede a enviarlo de la siguiente manera:

Message mensaje = new MimeMessage(sesion);

mensaje.setSubject("HolaMundo");

mensaje.setFrom(new InternetAddress(from));

mensaje.addRecipient( Message.RecipientType.TO,

newInternetAddress(to));

mensaje.setText("Este es el cuerpo del mensaje");

Transport.send(mensaje);

* + - 1. **Recepción de correo electrónico.**

**Rubén Gómez, (2008)**. Afirma que para recuperar los mensajes del servidor de correo electrónico son los siguientes:

1. **Definición de las propiedades de la conexión.**

Para definir las propiedades de la conexión se procederá a obtener las propiedades del sistema definiendo lo siguiente:

String popHost = "pop.correo.yahoo.es";

Propertiesprops = System.getProperties();

Session sesion = Session.getDefaultInstance(props,null);

1. **Conexion a un almacen de mensajes.**

Conceptualmente un Store representa nuestro buzón de correo dentro del servidor, es decir, nuestra cuenta de correo. La clase Store permite acceder a una estructura de carpetas (clase Folder) y a los mensajes contenidos en cada una de ellas. Define además el protocolo de acceso que se usará en la conexión:

Store store = sesion.getStore("pop3");

store.connect(popHost,usuario,clave);

1. **Recuperación de mensajes.**

Una vez conectados a nuestro Store hay que abrir la carpeta INBOXque, es la única disponible en POP3. Para ello se debe crear un objeto Folder a partir de la clase Store y que haga referencia a esta carpeta. Luego utilizaremos el método open y especificamos el método de apertura de los mensajes:

* READ\_ONLY.
* READ\_WRITE.

De modo que el código de apertura seria de la siguiente manera:

Folder folder = store.getFolder("INBOX");

folder.open(Folder.READ\_ONLY);

1. **Procesamiento de los mensajes.**

Por último se debe procesar los mensajes de alguna forma; en este ejemplo los mensajes serán mostrados por pantalla, presentado por cada mensaje su asunto, remitente (en el caso de que haya), fecha y hora en que fue enviado (recordar que POP3 no nos permite usar el método getReceviedDate para ver la fecha y hora en que ha sido recibido) y su contenido:

for (inti=0; i<mensajes.length; i++) {

System.out.println(

"Mensaje " + i + ":\n" +

"\tAsunto: " + mensajes[i].getSubject() + "\n" +

"\tRemitente: " + mensajes[i].getFrom()[0] + "\n" +

"\tFecha de Envío: " + mensajes[i].getSentDate() + "\n" +

"\tContenido: " + mensajes[i].getContent() + "\n");

}

Los Principales métodos de recuperación de atributos de un mensaje son los siguientes:

* **Addres[] getFrom()**

Devuelve un array en el que hay tantas direcciones como remitentes, Null si el campo *from*noestá presente en el mensaje o un arrayvacio si el mensaje no contiene ninguna dirección en el *from.*

* **Date getSentDate()**

Devuelve la fecha en la que el mensaje se envió.

* **StringgetSubject()**

Devuelve el asunto del mensaje.

* **ObjectgetContent()**

Devuelve el contenido del mensaje. Si este es de tipo texto plano, devolverá un string, y si es de tipo Multipart devolverá un objeto de esta clase.

* **Javax.activation.DataHandler  
  getDataHandler()**

Devuelve un DataHandler con el que poder manejar el contenido del mensaje. Con un DataHandler es posible manejar contenidos complejos a travez de un canal de entrada.

* 1. **Mensajes de Texto**
     1. **Servicio SMS usando SMPP 3.4**

Según **Luz Marina Santos, (2012)**, el servicio SMS permite transferir un mensaje de texto corto entre dos estaciones móviles (MSs), o entre una estación externa fija (ESME) y una estación móvil a través de un centro de servicio SMSC. Al enviar mensajes desde dispositivos externos, tal como un servidor, se debe conocer el API del operador de telefonía móvil para interactuar directamente con el SMSC, o encontrar otra solución.

El servicio final ofrecido es una comunicación extremo-extremo entre las estaciones involucradas. Cuando se envía un mensaje para solicitar algún tipo de servicio (o realizar alguna votación, como en los concursos de TV), un extremo es una estación móvil y la otra es un servidor que atiende las peticiones (o anota los votos).

* + - 1. **SMPP V3.4**

Según el portal **Oracle (2013)**, la implementación del Services Gatekeeper SMPP depende de un módulo básico, el servicio SMPP, que proporciona servicios de conectividad SMPP para plug-ins. El SMPP API define las interfaces entre los plug-ins y el Servicio SMPP.

Con esta API, los desarrolladores pueden crear plataformas SMPP plug-ins sin necesidad de administrar las tareas de bajo nivel de conexión de Servicios Gatekeeper para aplicaciones y para SMSCs.

**Luz Marina Santos, (2012)** afirma que es un protocolo estándar para el intercambio de mensajes SMS entre equipos que gestionan los mensajes como pueden ser los SMSCs y un sistema de solicitud de SMS como puede ser un servidor WAP o cualquier Gateway de mensajería. Se utiliza normalmente para permitir a terceros enviar mensajes (tales como pueden ser los proveedores de contenido).

SMPP fue desarrollado por Aldiscon, una pequeña firma irlandesa comprada posteriormente por Logica. En 1999, SMPP pasó formalmente a manos del *'SMPP DevelopersForum'*, posteriormente rebautizado como el *SMS Forum*. El SMPP es el interfaz que permite que entidades de envío de SMS que se encuentran fuera de la red móvil (ESME) puedan interconectarse con los elementos internos como la SMSC.

SMPP define:

* El conjunto de operaciones para el intercambio de SMSs entre los ESME y el SMSC.
* Los datos que los ESME debe intercambiar con el SMSC durante la conexión.

El protocolo se basa en el intercambio, petición/respuesta, de pares de PDUs, estos se intercambian sobre la capa de transporte. El intercambio de datos puede realizarse de manera síncrona, esperando cada parte la respuesta/petición del otro para enviar la correspondiente petición/respuesta, o asíncrona, donde el envío y la recepción van a través de distintos hilos.

El intercambio de mensajes entre una ESME y un SMSC a través de SMPP puede ser categorizado dentro de tres distintos grupos de transacciones.

* Mensaje enviado de la ESME (Transmisor) para el SMSC.
* Mensaje enviado de la SMSC para la ESME (Receptor).
* Mensaje enviado de la ESME (Transceptor) para el SMSC y del SMSC para el ESME (Transceptor).

# FASE DE INICIO.

# CAPÍTULO 4 - Requisitos

1. **Captura de Requisitos**


5. 1. **Descripción.**

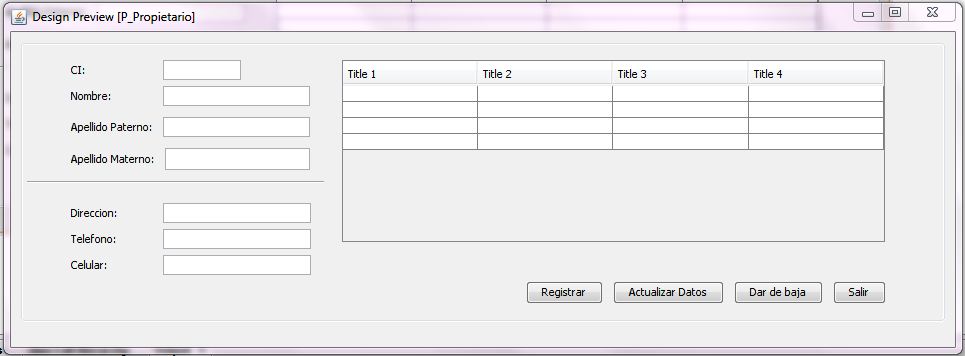
Durante este flujo de trabajo se identificaran las necesidades del cliente en términos de requisitos, vale decir, se definirá el alcance del proyecto en términos de requisitos funcionales del mismo, así como también se llevara a cabo la identificación y detalle de los requisitos no funcionales del proyecto.

* 1. **Identificar actores y casos de uso.**

**Caso de Uso 1 Gestionar Propietario.**

****

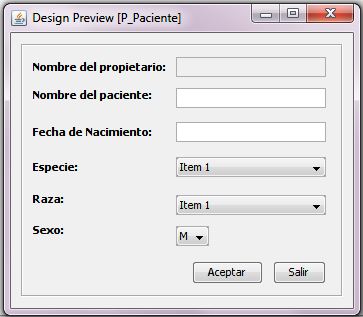
|  |  |
| --- | --- |
| **Nro.:** 1 | **CU:** Gestionar Propietario |
| **Propósito:** Realizar la gestión de todos los propietarios de mascotas que acuden a la clínica, que incluyen el registro o modificación de los propietarios de mascotas. | |
| **Descripción:** Proceso que se lleva a cavo en recepción de la clínica veterinaria, a cargo de la secretaria quien verifica si el es necesario el registro del propietario o la modificación de sus datos para mantener actualizado el sistema. | |
| **Actores:** Propietario | |
| **Iniciador:** Propietario | |
| **Proceso:**   * Si se trata de una modificación:   + Ingresar los datos a modificar   + Realizar actualizaciones pertinentes * Si se trate de un registro de nuevo propietario:   + Registrar los siguiente datos:     - CI     - Nombre     - Teléfono     - Celular     - Dirección   + Validar datos | |
| **Casos Especiales:** | |

****

**Caso de Uso 2 Gestionar Paciente**

****

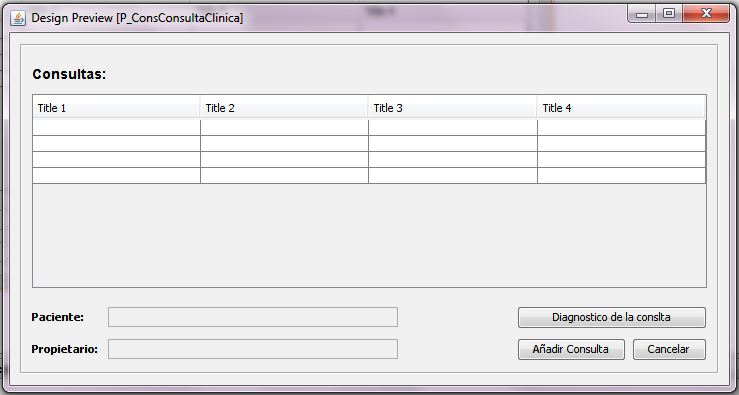
|  |  |
| --- | --- |
| **Nro.:** 2 | **CU:** Gestionar Paciente |
| **Propósito:** Realizar la gestión de todos los pacientes que acuden a la clínica, esto incluye las operaciones de registro y modificación de los pacientes de la clínica veterinaria. | |
| **Descripción:** Procesos que son llevados a cabo en la recepción de la clínica veterinaria, donde la secretaria se encarga ya sea de registrar un nuevo paciente o efectuar modificaciones de sus datos, como ser: especie, nombre, fecha de nacimiento, raza, color. | |
| **Actores:** Paciente, Propietario. | |
| **Iniciador:** Propietario | |
| **Proceso:**   * Si se desea registrar un nuevo paciente:   + Ingresar los siguientes datos del nuevo paciente: * Nombre * Fecha de nacimiento * Raza * Especie * Color del paciente * Sexo * Conducta * Alimentación   + Validar datos   + Asignar código * Si se desea modificar los datos de una mascota.   + Ingresar los datos a modificar   + Realizar actualizaciones pertinentes | |
| **Casos Especiales:** | |

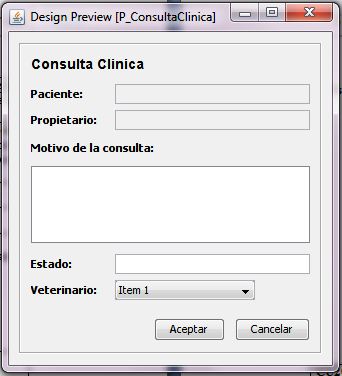
****

**Caso de uso 3 Gestionar Consultas Clínicas**

******

|  |  |
| --- | --- |
| **Nro.:** 3 | **CU:** Gestionar Consultas |
| **Propósito:** Realizar la gestión de las consultas que se realizan en la clínica. | |
| **Descripción:** Proceso que se lleva a cabo en la recepción de la clínica, a cargo de la recepcionista, quien verifica la disponibilidad de horarios y médicos veterinarios, para después asignarle al paciente una reserva de consulta. | |
| **Actores:** Propietario, Paciente, Veterinario | |
| **Iniciador:** Propietario | |
| **Proceso:**   * Seleccionar un propietario * Seleccionar un paciente * Seleccionar un veterinario * Registrar la fecha * Almacenar los datos seleccionados | |
| **Casos Especiales:** | |

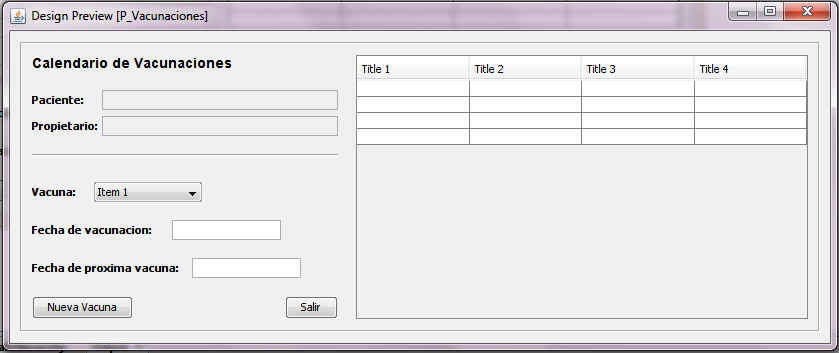
****

****

**Caso de Uso 4 Gestionar Vacunación**

****

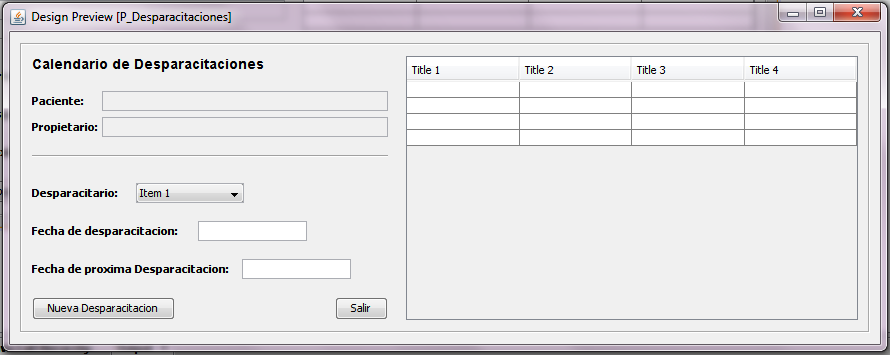
|  |  |
| --- | --- |
| **Nro.:** 4 | **CU:** Gestionar Vacunación |
| **Propósito:** Realizar la gestión de toda la vacunación que reciben los pacientes de la clínica. | |
| **Descripción:** Proceso que es llevado a cabo por el médico veterinario en su consultorio, quien registrara las vacunas suministradas a los pacientes, la fecha de la misma y los fármacos implicados. | |
| **Actores:** Veterinario, Paciente, Propietario | |
| **Iniciador:** Propietario | |
| **Proceso:**   * Consultar Consulta * Ingresar los siguientes datos:   + Nombre de la vacuna   + Fecha   + Fecha de la próxima vacuna   + Nombre de la próxima vacuna * Validar los datos * Almacenar los datos en el perfil del paciente | |
| **Casos Especiales:** | |

****

**Caso de Uso 5 Gestionar Desparasitación**

****

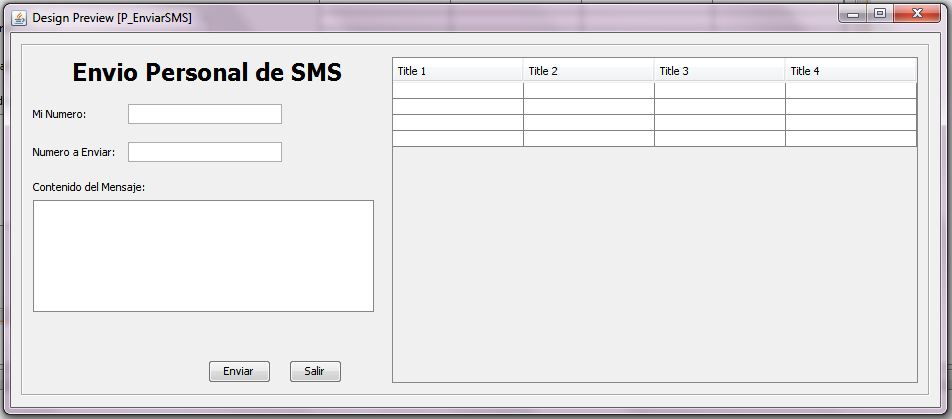
|  |  |
| --- | --- |
| **Nro:** 5 | **CU:** Gestionar Desparasitación |
| **Propósito:** Realizar la gestión de todas las desparasitaciones que se realizan en la clínica. | |
| **Descripción:** Gestionar todas las desparasitaciones que se realizan en la clínica a los pacientes que la requieren, y cuando la requieren. Viendo los medicamentos adecuados para cada paciente. Se registraran la fecha en que se realizo, producto utilizado, fecha de la próxima desparasitación. | |
| **Actores:** Paciente, Médico Veterinario | |
| **Iniciador:** Médico Veterinario | |
| **Proceso:**   * Consultar veterinario * Consultar Propietario * Consultar paciente * Si no está registrado * Registrar al nuevo paciente * Consultar producto veterinario * Registrar la Fecha, Producto, Paciente de la desparasitación. | |
| **Casos Especiales:** | |

****

**Caso de uso 6 Envío personal de SMS.**

****

|  |  |
| --- | --- |
| **Nro:** 6 | **CU:** Envío personal de sms. |
| **Propósito:** Enviar mensajes de texto a un propietario de paciente determinado en la clínica veterinaria. | |
| **Descripción:** Caso de uso que permitirá él envió de las notificaciones o recordatorios mediante el envío de mensajes de texto del celular del propietario. | |
| **Actores:** Propietario. | |
| **Iniciador:** Propietario | |
| **Proceso:**   * Ingresar datos de propietario * Ingresar el cuerpo del mensaje. * Enviar mensaje de texto. | |
| **Casos Especiales:** | |

****

**Caso de uso 7 Enviar notificaciones por correo electrónico.**

****

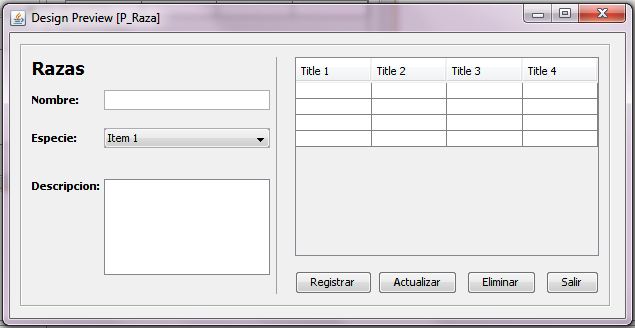
|  |  |
| --- | --- |
| **Nro:** 7 | **CU:** Enviar notificaciones por correo electrónico. |
| **Propósito:** Enviar notificaciones al correo electrónico del propietario del paciente de la clínica veterinaria. | |
| **Descripción:** Caso de uso que permitirá el envió de las notificaciones al propietario del paciente de la clínica veterinaria | |
| **Actores:** Propietario. | |
| **Iniciador:** Propietario. | |
| **Proceso:**   * Consultar datos del paciente. * Seleccionar propietario de paciente. * Ingresar el asunto y cuerpo del correo electrónico. * Enviar correo electrónico. | |
| **Casos Especiales:** | |

****

**Caso de uso 8 Gestionar Raza**

****

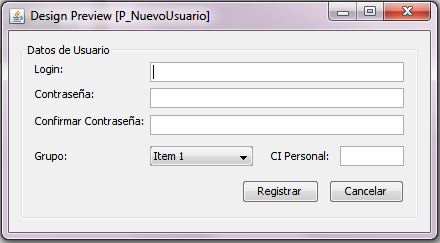
|  |  |
| --- | --- |
| **Nro:** 8 | **CU:** Gestionar Raza |
| **Propósito:** Gestionar todas las razas registradas en el sistema y con las cuales se trabaja en la clínica. | |
| **Descripción:** Gestionar todas las razas cada una perteneciente a una especie respectiva, con las que se trabaja en la clínica. Para mantener un control de que pacientes y cuantos corresponden a cada raza. Para esto se deberá registrar el nombre de la raza, y una descripción de la misma, además de la especie a la que corresponde. | |
| **Actores:** Secretaria | |
| **Iniciador:** Secretaria | |
| **Proceso:**   * Indicar el nombre de la raza y una descripción de la misma. * Si no está registrado * Se asignara un código a la nueva raza * Se guardaran los nuevos datos * Si la raza ya está registrada * Se informa de la existencia de la raza * No se guardan los datos ingresados | |
| **Casos Especiales:** | |

****

**Caso de uso 9 Gestionar usuarios**

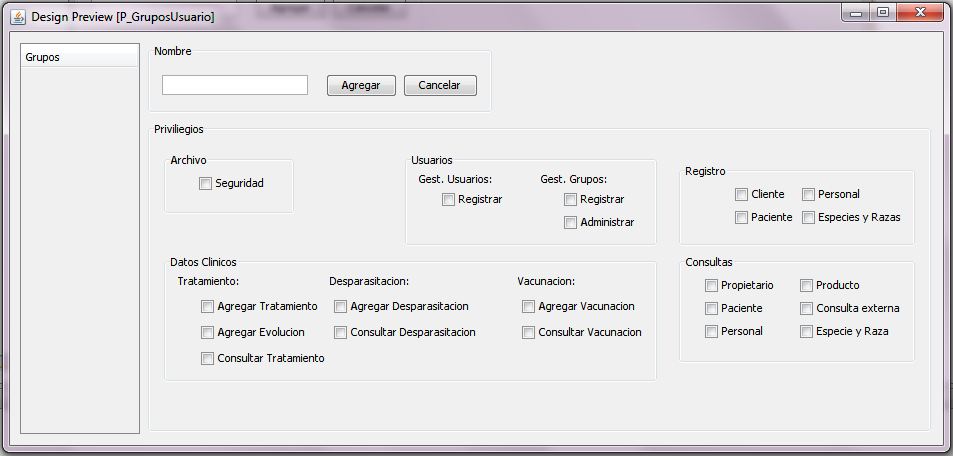


|  |  |
| --- | --- |
| **Nro:** 9 | **CU:** Gestionar Usuarios |
| **Propósito:** Realizar la gestión de los usuarios que tendrán acceso al sistema. | |
| **Descripción:** Gestionar los datos como el nombre de usuario, clave y privilegios de todos los usuarios que podrán acceder al sistema. | |
| **Actores:** Administrador | |
| **Iniciador:** Administrador | |
| **Proceso:**  Identificar Usuario como por ejemplo administrador  Ingresar nombre, clave, privilegios que tendrá el nuevo usuario  Guardar los datos ingresados | |
| **Casos Especiales:** | |

****

**Caso de uso 10 Gestionar Grupos de Usuario**

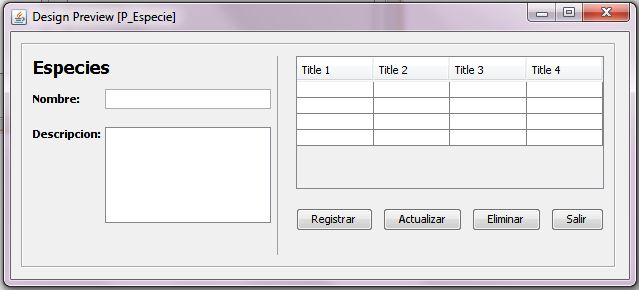
|  |  |
| --- | --- |
| **Nro:** 10 | **CU:** Gestionar Grupos de Usuario |
| **Propósito:** Crear grupos de usuario y definir los privilegios que tendrá. | |
| **Descripción:** Crear grupos a los que pertenecerán los usuarios del sistema y de esta manera restringir el acceso al software para cada grupo de usuario. | |
| **Actores:** Administrador. | |
| **Iniciador:** Administrador. | |
| **Proceso:**  Ingresar el nombre del nuevo grupo.  Indicar los privilegios del nuevo grupo.  Crear el nuevo grupo. | |
| **Casos Especiales:** | |

****

**Caso de Uso 11 Gestionar Especie**

****

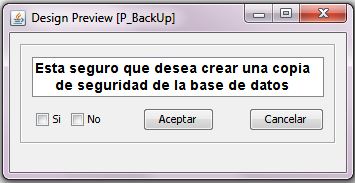
|  |  |
| --- | --- |
| **Nro:**11 | **CU:** Gestionar Especie |
| **Propósito:** Realizar las gestiones de todas las especies de animales que la clínica atenderá. | |
| **Descripción:** Gestionar las especies que la clínica manejara para tener registrado los paciente que pertenecen a cada especie. Registrando el nombre de la especie, y una descripción de la misma. A cada especie registrada le pertenecerán diferentes razas. | |
| **Actores:** Administrador | |
| **Iniciador:** Administrador | |
| **Proceso:**   * Indicar el nombre de la especie y una descripción dela misma. * Si no está registrado * Se asignara un código a la nueva especie * Se guardaran los nuevos datos * Si la especie ya está registrada * Se informa de la existencia de la especie * No se guardan los datos ingresados | |
| **Casos Especiales:** | |

****

**Caso de Uso 12 Elaborar un Backup del sistema**



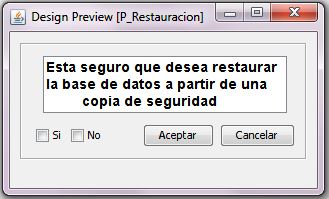
|  |  |
| --- | --- |
| **Nro:12** | **CU:** Elaborar un Backup del sistema |
| **Propósito:** Asegurar la información del Sistema, como ser los datos de los clientes y sus mascotas, además de los usuarios y los productos que se encuentran entre otros. | |
| **Descripción:**El administrador dispondrá de la posibilidad de efectuar una copia de seguridad de los datos que contiene el sistema en el momento que vea conveniente, de la misma manera podrá programarlo para realizar una copia periódica de la información con la finalidad de asegurar la información de la clínica veterinaria. | |
| **Actores:** Administrador, Tiempo | |
| **Iniciador:** Administrador | |
| **Proceso:**   * El administrador entra al Sistema con sus Datos correspondientes * El administrador ingresa a la opción de Backup * El sistema le pide la dirección donde desea guardar la información * El administrador selecciona la dirección de guardado * El sistema Guarda dicha información en la dirección indicada por el administrador * El sistema muestra una notificación donde indica que se finalizo dicho proceso * El Sistema Vuelve a el Formulario principal | |
| **Casos Especiales:** | |

****

**Caso de Uso 13 Realizar restauración del sistema**

****

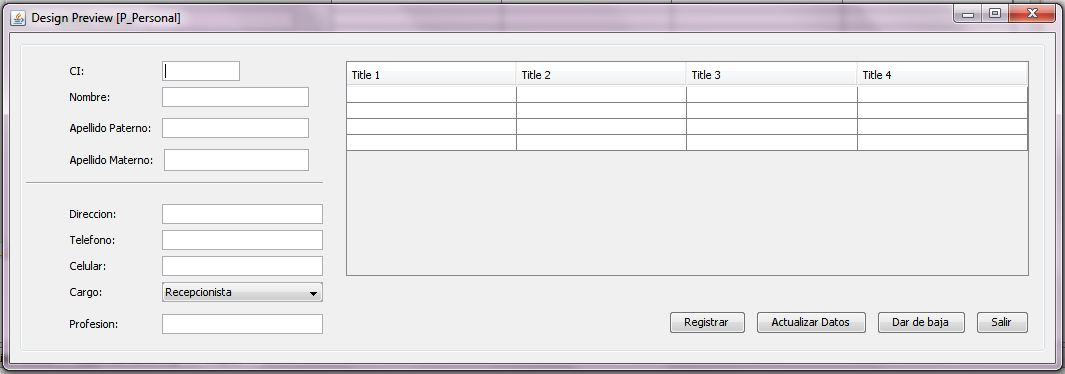
|  |  |
| --- | --- |
| **Nro.:13** | **CU:** Realizar restauración del sistema |
| **Propósito:** Recuperar información del sistema luego de algún acontecimiento que cree alguna inconsistencia de datos o perdida de información. | |
| **Descripción:** En caso de que el sistema sufra alguna contingencia, el administrador dispondrá de la posibilidad de efectuar una restauración del sistema desde una de las copias de seguridad que realizo anteriormente. | |
| **Actores:** Administrador | |
| **Iniciador:** Administrador | |
| **Proceso:**   * El administrador entra al Sistema con sus Datos correspondientes * El administrador ingresa a la opción de Restauración del Sistema * El sistema le pide la dirección de donde se encuentra el archivo de Backup * El administrador selecciona la dirección donde el archivo de backup se encuentre guardado * El administrador selecciona el botón para iniciar la restauración del Sistema * El sistema muestra una notificación para asegurar si el administrador desea continuar o abortar. Dicho proceso, indicando que la información actual será reemplazada y no habrá vuelta atrás. * Si el usuario desea continuar con la restauración del sistema * El sistema inicia con la restauración del sistema * Al terminar muestra una notificación al Administrador indicando que el proceso a terminado * El sistema vuelve a el formulario principal | |
| **Casos Especiales:** | |

****

**Caso de Uso 14 Gestionar Personal**

****

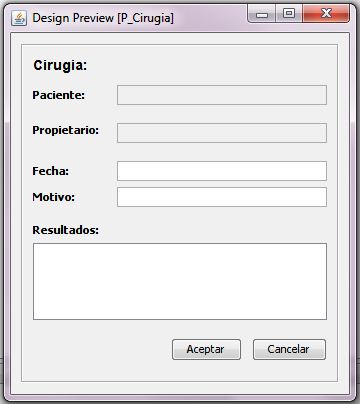
|  |  |
| --- | --- |
| **Nro.:** 14 | **CU:** Gestionar Personal |
| **Propósito:** Realizar la gestión del personal que trabajan en la clínica. | |
| **Descripción:** Proceso que se lleva a cabo en oficinas de administración, donde el administrador se encarga de realizar las operaciones de registro de nuevo personal de trabajo, así como también la modificación de los datos. | |
| **Actores:**  Personal | |
| **Iniciador:** Personal | |
| **Proceso:**   * Si se trata de una modificación:   + Ingresar los datos a modificar   + Almacenar actualizaciones pertinentes * Si se trata de registro de un nuevo personal   + Ingresar los siguientes datos:     - CI     - Registro     - Dirección     - Teléfono.   + Asignar código al personal | |
| **Casos Especiales:** | |

****

**Caso de Uso 15 Gestionar Cirugía**

****

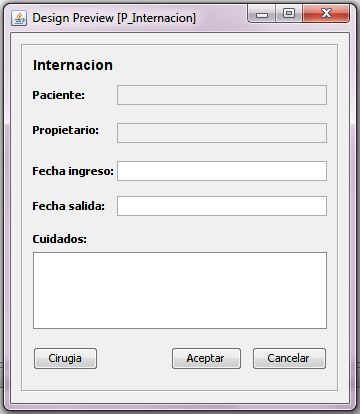
|  |  |
| --- | --- |
| **Nro.:** 15 | **CU:** Gestionar Cirugía |
| **Propósito:** Realizar la gestión de todas las cirugías que se hacen en la clínica. | |
| **Descripción:** Gestionar todas las cirugías que se hacen o se requieran practicar a los pacientes de la clínica. Registrando tipo de cirugía, porcentaje de riesgo, fecha, observaciones. | |
| **Actores:** Médico Veterinario, Paciente | |
| **Iniciador:** Médico Veterinario | |
| **Proceso:**   * Consultar Medico Veterinario * Consultar Paciente * Registrar estado pre Cirugía del paciente * Registrar resultados, observaciones, tiempo, fecha y tipo de la cirugía | |
| **Casos Especiales:** | |

****

**Caso de uso 16 Gestionar Internación**

****

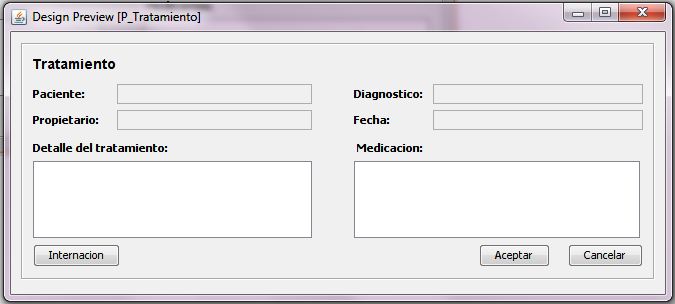
|  |  |
| --- | --- |
| **Nro.:** 16 | **CU:** Gestionar Internación |
| **Propósito:** Realizar la gestión de las internaciones que se realicen en la clínica. | |
| **Descripción:** Se registraran datos de la internación que un veterinario ordena a un paciente, ingresando la fecha de la internación, síntomas del paciente, tiempo aproximado, y el motivo de la internación. | |
| **Actores:** Medico Veterinario, Paciente | |
| **Iniciador:** Medico Veterinario | |
| **Proceso:**  Consultar Propietario  Si no está registrado  Se registra al nuevo propietario y a la mascota  Si esta registrado  Indicar datos del Paciente  Consultar paciente  Si está registrado  Registrar fecha, tiempo aproximado, motivo de la internación, síntomas.  Guardar datos | |
| **Casos Especiales:** | |

****

**Caso de uso 17 Gestionar Tratamiento**

****

|  |  |
| --- | --- |
| **Nro.:** 17 | **CU:** Gestionar Tratamiento |
| **Propósito:** Realizar la gestión de todos los tratamientos que se realicen a todas las mascotas en la clínica. | |
| **Descripción:** Se gestionara el tratamiento prescrito por un veterinario a un paciente. Registrando los medicamentos, dosis, tiempo del tratamiento y cuidados, que corresponde al tratamiento prescrito. | |
| **Actores:** Paciente, Médico Veterinario | |
| **Iniciador:** Medico Veterinario | |
| **Proceso:**  Agregar nuevo tratamiento  Ingresar la fecha, medicamento, dosis, tiempo de duración y cuidados.  Guardar los datos ingresados. | |
| **Casos Especiales:** | |



**Caso de uso 18 Procesar Consulta por Correo electrónico.**

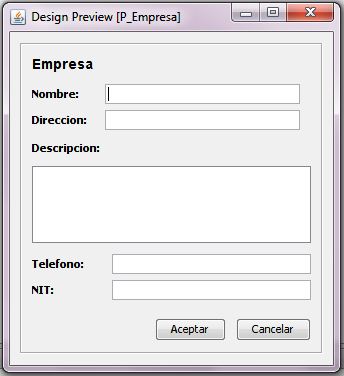
****

|  |  |
| --- | --- |
| **Nro.:** 18 | **CU:** Procesar Consulta por Correo electrónico. |
| **Propósito:** Procesar las consultas que se realicen a través de correo electrónico a la base de datos. | |
| **Descripción:** Caso de uso que permitirá a los propietarios realizar consultas sobre enfermedades y tratamientos del paciente mediante el envío de correos electrónicos. | |
| **Actores:** Propietario, Administrador. | |
| **Iniciador:** Administrador. | |
| **Proceso:**   * Recibir mensaje de correo electrónico. * Cargar datos solicitados. * Ingresar el asunto y cuerpo del correo electrónico. * Enviar correo electrónico. | |
| **Casos Especiales:** | |

**Caso de uso 19 Gestionar Empresa**

****

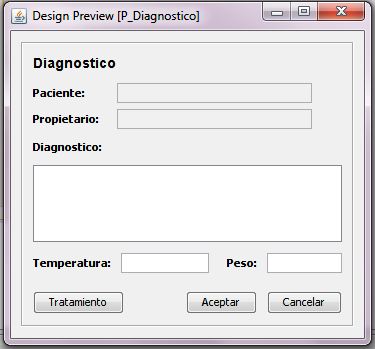
|  |  |
| --- | --- |
| **Nro.:** 19 | **CU:** Gestionar Empresa |
| **Propósito:** Ingresar al sistema la información de la clínica veterinaria con el propósito de personalizar el sistema. | |
| **Descripción:** El administrador se encarga de introducir al sistema los datos identificativos de la clínica veterinaria con el fin de personalizar el sistema para un manejo más adecuado de este. | |
| **Actores:** Administrador | |
| **Iniciador:** Administrador | |
| **Proceso:** Al iniciar a utilizar el sistema:   * El administrador ingresa al sistema. * Accede a la sección de administración del sistema. * Ingresa los datos de la clínica veterinaria, tales como:   + El nombre de la clínica veterinaria   + Los horarios de atención de la clínica   + El personal del que dispone la clínica veterinaria. * El administrador guarda los cambios. | |
| **Casos Especiales:** | |



**Caso de Uso 20 Gestionar Diagnostico**

****

|  |  |
| --- | --- |
| **Nro.:** 20 | **CU:** Gestionar Diagnóstico |
| **Propósito:** Realizar las gestión de todos los diagnósticos que se hacen en la clínica. | |
| **Descripción:** Gestionar todos los diagnósticos y consultas que se realizan en la clínica a los pacientes registrados. Registrando los síntomas y resultados como diagnostico después de el chequeo realizado por el veterinario. | |
| **Actores:** Paciente, Médico Veterinario | |
| **Iniciador:** Medico Veterinario | |
| **Proceso:**   * Indicar la fecha, temperatura del paciente, estado del paciente, observaciones, la descripción del diagnostico * Si no existen análisis * Agregar análisis * Guardar análisis * Agregar nuevo diagnostico * Guardar los datos ingresados * Registrar el diagnóstico | |
| **Casos Especiales:** | |

****

**Caso de Uso 21: Envío automático de sms.**

****

|  |  |
| --- | --- |
| **Nro.:**21 | **CU:** Envío automático de sms. |
| **Propósito:** Enviar mensajes de texto de forma automática a los propietarios de pacientes de la clínica veterinaria. | |
| **Descripción:** Caso de uso que permitirá él envió automático de notificaciones o recordatorios al celular del propietario | |
| **Actores:** Propietario. | |
| **Iniciador:** Propietario | |
| **Proceso:**   * Consultar datos del paciente. * Si hay notificación: Seleccionar propietario. * Si hay notificación: Enviar mensaje de texto. | |
| **Casos Especiales:** | |

**Caso de Uso 22: Procesar SMS.**

****

|  |  |
| --- | --- |
| **Nro.:**21 | **CU:** Envío automático de sms. |
| **Propósito:** Procesar los sms que llegan como respuesta a un aviso de cita para una vacunación. | |
| **Descripción:** Caso de uso que permitirá el procesamiento de sms que llegan como respuesta por parte de los clientes que son notificados para vacunaciones que se deben realizar en una fecha determinada. | |
| **Actores:** Propietario. | |
| **Iniciador:** Propietario | |
| **Proceso:**   * Recibir un sms de clientes con una de las dos opciones:   + Confirmar   + Cancelar. * Verificar datos de la cita. * Si la cita está pendiente   + Si Confirmar: Confirmar la cita   + Si Cancelar: Cancelar la cita. | |
| **Casos Especiales:** | |

* 1. **Priorización de casos de uso**

|  |  |  |  |
| --- | --- | --- | --- |
| **N°** | **Casos de Uso** | **Actores** | **Prioridad** |
| 1 | CU1. Gestionar Propietario | Propietario | Esencial |
| 2 | CU2. Gestionar Paciente | Paciente | Esencial |
| 3 | CU3. Gestionar Consultas Clínicas | Paciente, Veterinario, Propietario | Medio |
| 4 | CU4. Gestionar Vacunación | Paciente, Veterinario, Propietario | Medio |
| 5 | CU5. Gestionar Desparasitación | Paciente, Veterinario | Medio |
| 6 | CU6. Envío personal de sms. | Propietario | Baja |
| 7 | CU7: Enviar notificaciones por correo electrónico. | Propietario | Baja |
| 8 | CU8: Gestión Raza | Secretaria | Esencial |
| 9 | CU9. Gestionar Usuarios | Administrador | Esencial |
| 10 | CU10. Gestionar Grupos de Usuario | Administrador | Esencial |
| 11 | CU11: Gestionar Especies | Administrador | Esencial |
| 12 | CU12: Elaborar un Backup del sistema | Administrador, Tiempo | Esencial |
| 13 | CU13: Realizar restauración del sistema | Administrador | Esencial |
| 14 | CU14: Gestionar Personal | Administrador | Esencial |
| 15 | CU15: Gestionar Cirugía | Paciente, Veterinario | Medio |
| 16 | CU116: Gestionar Internación | Paciente, Veterinario | Medio |
| 17 | CU17: Gestionar Tratamiento | Paciente, Veterinario | Medio |
| 18 | CU18: Procesar Consulta por Correo electrónico | Administrador, Propietario | Baja |
| 19 | CU19: Gestionar Empresa | Administrador | Esencial |
| 20 | CU20: Gestionar Diagnostico | Paciente, Veterinario | Medio |
| 21 | CU21: Envío Automático de SMS | Propietario | Baja |

* 1. **Diagrama general de casos de uso.**

****



**FASE DE ELABORACION**

**CAPÍTULO 5 - Análisis**



## Descripción:

Durante este flujo de trabajo se realizara el análisis de los diferentes casos de uso identificados en el capítulo anterior. Para ello se realizaran los diagramas de comunicación para cada uno de los casos de uso identificados.

* 1. **Análisis Paquetes.**
     1. **Identificación de Paquetes.**
* **P1: Registro.**
* **P2: Datos Clínicos.**

****

* **P3: Usuarios.**

****

* **P4: Seguridad.**

****

* **P5: Servicios Adicionales.**

****

* 1. **Análisis de la arquitectura.**
     1. **Vista de paquetes.**
* **Paquete Registro.**

****

* **Paquete de Datos Clínicos.**

****

* **Paquete Usuarios.**

****

* **Paquete Seguridad.**

****

* **Paquete Servicios adicionales.**

****

* 1. **Análisis de Casos de Uso.**

### **Diagramas de Comunicación.**

**Caso 1: Gestionar Propietario**

****

**Caso 2: Gestionar Paciente.**

**Caso 3: Gestionar Consultas Cínicas.**

****

**Caso 5: Gestionar Personal.**

****

**Caso 8: Envío personal de sms.**

**Caso 9: Procesar Consulta por Correo electrónico**

****

**Caso 11: Gestionar Tratamiento.**

**Caso 12: Enviar notificaciones por correo electrónico.**

****

**Caso 13: Gestionar Empresa.**

****

**Caso 14: Gestionar Raza.**

****

**Caso 15: Gestionar Usuarios.**

****

**Caso 17: Gestionar Diagnostico.**



**Caso 19: Gestionar Especie.**

****

**Caso 20: Elaborar un Backup del Sistema.**

****

**Caso 21: Realizar restauración del Sistema.**

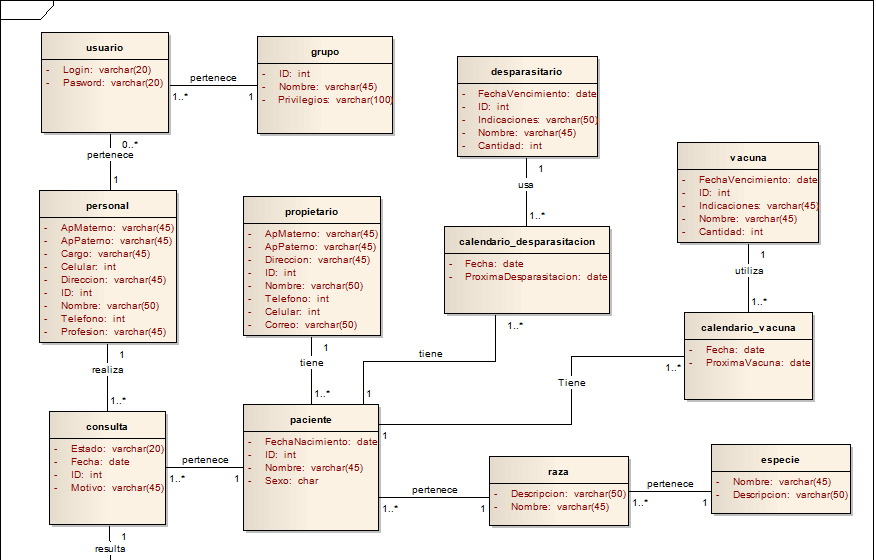
**Caso 22: Procesar SMS**

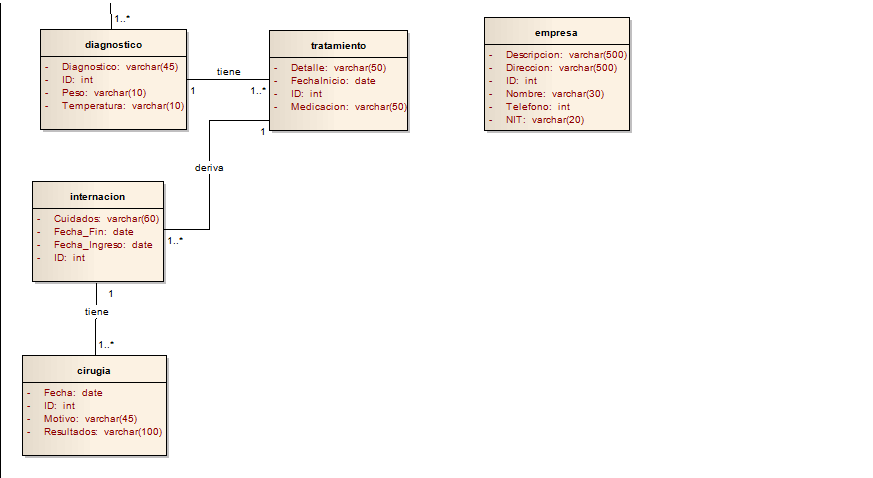
# CAPÍTULO 6 - Diseño

1. **Descripción**

Durante este capítulo se realizara el diseño del Software a fin de determinar la forma de implementar los requisitos descritos.

## Diseño conceptual de la Base de Datos.





## Diseño Lógico de la Base de Datos.

### **Mapeo.**

**grupo**

|  |  |  |
| --- | --- | --- |
| ID | Nombre | Privilegios |

**usuario**

|  |  |  |  |
| --- | --- | --- | --- |
| Login | Pasword | **Grupo** | **Personal\_ID** |

**personal**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ID | Nombre | ApPaterno | ApMaterno | Telefono | Celular | Direccion | Cargo | Profesion |

**consulta**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID | Fecha | Motivo | Estado | **Personal\_ID** | **Paciente\_ID** |

**paciente**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ID | Nombre | FechaNacimiento | **Propietario\_ID** | **Raza** | **Especie** | Sexo |

**propietario**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| ID | Nombre | ApPaterno | ApMaterno | Telefono | Direccion | Celular | Correo |

**raza**

|  |  |  |
| --- | --- | --- |
| Nombre | Discripcion | **Especie** |

**especie**

|  |  |
| --- | --- |
| Nombre | Descripcion |

**calendario\_vacuna**

|  |  |  |  |
| --- | --- | --- | --- |
| **Vacuna\_ID** | **Paciente\_ID** | Fecha | ProximaVacuna |

**vacuna**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Nombre | FechaVencimiento | Indicaciones | Cantidad |

**Calendario\_desparasitacion**

|  |  |  |  |
| --- | --- | --- | --- |
| **Desparasitario\_ID** | **Paciente\_ID** | Fecha | ProximaDesparasitacion |

**desparasitario**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Nombre | FechaVencimiento | Indicaciones | Cantidad |

**diagnostico**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Diagnostico | Temperatura | Peso | **Consulta\_ID** |

**tratamiento**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Detalle | Medicacion | FechaInicio | **Diagnostico\_ID** |

**internacion**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | **Tratamiento\_ID** | Fecha\_Ingreso | Fecha\_Fin | Cuidados |

**cirugia**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | **Internacion\_ID** | Fecha | Motivo | Resultados |

**empresa**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID | Nombre | Descripcion | Direccion | Telefono | NIT |

* 1. **Diseño Físico de la Base de Datos.**
     1. **Tablas de Volumen.**

**Calendario\_desparacitacion**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Atributos** | **Tipos de Datos** | **Amplitud** | **Llave** | **Nulo** | **Descripción** |
| Desparacitario\_ID | Int | 11 | PK, FK | NO | Lave foránea a la tabla desparasitario |
| Paciente\_ID | Int | 11 | PK, FK | NO | Lave foránea a la tabla Paciente |
| Fecha | date |  | PK | NO | Llave Primaria de la tabla |
| ProximaDesparasitacion | date |  | - | SI | Fecha de la próxima desparasitación. |

**Calendario\_vacuna**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Atributos** | **Tipos de Datos** | **Amplitud** | **Llave** | **Nulo** | **Descripción** |
| Vacuna\_ID | Int | 11 | PK, FK | NO | Lave foránea a la tabla vacuna |
| Paciente\_ID | Int | 11 | PK, FK | NO | Lave foránea a la tabla Paciente |
| Fecha | Date | - | PK | NO | Llave Primaria de la tabla |
| ProximaVacuna | Date | - | - | SI | Fecha de la próxima vacuna. |

**cirugia**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Atributos** | **Tipos de Datos** | **Amplitud** | **Llave** | **Nulo** | **Descripción** |
| ID | Int | 11 | PK | NO | Lave primaria de la tabla autoincrementable |
| Internacion\_ID | Int | 11 | FK | NO | Lave foránea a la tabla internacion |
| Fecha | Date | - | - | NO | Fecha de la cirugia |
| Motivo | Varchar | 100 | - | NO | Motivo de la cirugía |
| Resultados | Varchar | 100 | - | NO | Resultados de la cirugía |

**consulta**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Atributos** | **Tipos de Datos** | **Amplitud** | **Llave** | **Nulo** | **Descripción** |
| ID | Int | 11 | PK | NO | Lave primaria de la tabla autoincrementable |
| Fecha | Date | - | - | NO | Fecha de la consulta |
| Motivo | Varchar | 100 | - | NO | Motivo de la consulta |
| Estado | Varchar | 50 | - | NO | Resultados de la consulta |
| Personal\_ID | Int | 11 | FK | NO | Llave foránea a la tabla personal |
| Paciente\_ID | Int | 11 | FK | NO | Llave foránea a la tabla paciente |

**desparasitario**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Atributos** | **Tipos de Datos** | **Amplitud** | **Llave** | **Nulo** | **Descripción** |
| ID | Int | 11 | PK | NO | Lave primaria de la tabla autoincrementable |
| Nombre | Varchar | 30 | - | NO | Nombre del producto desparasitario |
| FechaVencimento | Date | - | - | NO | Fecha de vencimiento del producto |
| Indicaciones | Varchar | 100 | - | SI | Indicaciones de cuando es necesario elproducto |
| Cantidad | Int | 11 | - | NO | Cantidad de producto disponible |

**diagnostico**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Atributos** | **Tipos de Datos** | **Amplitud** | **Llave** | **Nulo** | **Descripción** |
| ID | Int | 11 | PK | NO | Lave primaria de la tabla autoincrementable |
| Diagnostico | Varchar | 50 | - | NO | Diagnostico de la consulta |
| Temperatura | varchar | 10 | - | NO | Temperatura del paciente al momento de la consulta |
| Peso | Varchar | 10 | - | NO | Peso del paciente al momento de la consulta |
| Consulta\_ID | Int | 11 | FK | NO | Llave foránea a la tabla consulta |

**Empresa**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Atributos** | **Tipos de Datos** | **Amplitud** | **Llave** | **Nulo** | **Descripción** |
| ID | Int | 11 | PK | NO | Lave primaria de la tabla autoincrementable |
| Nombre | Varchar | 30 | - | NO | Nombre de la empresa |
| Descripcion | varchar | 500 | - | SI | Descripción de la empresa |
| Direccion | Varchar | 150 | - | NO | Direccion de la empresa |
| Telefono | Int | 11 | - | SI | Telefono de la empresa |
| NIT | Varchar | 20 | - | SI | NIT de la empresa |

**especie**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Atributos** | **Tipos de Datos** | **Amplitud** | **Llave** | **Nulo** | **Descripción** |
| Nombre | Varchar | 30 | PK | NO | Lave primaria de la tabla y nombre de la especie |
| Descripcion | varchar | 50 | - | SI | Descripción de la especie |

**Grupo**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Atributos** | **Tipos de Datos** | **Amplitud** | **Llave** | **Nulo** | **Descripción** |
| ID | int | 11 | PK | NO | Llave primaria de la tabla |
| Nombre | Varchar | 45 | - | NO | Nombre del grupo. |
| Privilegios | Varchar | 100 | - | NO | Cadena de caracteres que representa los privilegios sobre las acciones que se pueden realizar en el software. |

**Internacion**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Atributos** | **Tipos de Datos** | **Amplitud** | **Llave** | **Nulo** | **Descripción** |
| ID | Int | 11 | PK | NO | Lave primaria de la tabla autoincrementable |
| Tratamiento\_ID | Int | 11 | FK | NO | Llave foránea a la tabla tratamiento |
| Fecha\_Ingreso | Date | - | - | NO | Fecha de internación |
| Fecha\_Fin | Date | - | - | NO | Fecha de dada de alta del paciente |
| Cuidados | varchar | 100 | - | NO | Cuidados sobre la mascota una vez dada de alta |

**Paciente**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Atributos** | **Tipos de Datos** | **Amplitud** | **Llave** | **Nulo** | **Descripción** |
| ID | Int | 11 | PK | NO | Lave primaria de la tabla autoincrementable |
| Nombre | Varchar | 45 | - | NO | Nombre del paciente |
| FechaNacimiento | Date | - | - | NO | Fecha de nacimiento del paciente |
| Propietario\_ID | int | 11 | FK | NO | Llave foránea a la tabla propietario |
| Raza | Varchar | 30 | FK | NO | Llave foránea a la tabla Raza |
| Especie | Varchar | 30 | FK | NO | Llave foránea a la tabla especie |
| Sexo | Char | 1 | - | NO | Genero del paciente |

**Personal**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Atributos** | **Tipos de Datos** | **Amplitud** | **Llave** | **Nulo** | **Descripción** |
| ID | Int | 11 | PK | NO | Lave primaria de la tabla o CI del empleado |
| Nombre | Varchar | 30 | - | NO | Nombre del empleado |
| ApPaterno | Varchar | 20 | - | NO | Apellido paterno del empleado |
| ApMaterno | Varchar | 20 | - | NO | Apellido materno del empleado |
| Telefono | Int | 11 | - | SI | Teléfono del empleado |
| Celular | int | 11 | - | SI | Celular del empleado |
| Direccion | Varchar | 50 | - | NO | Dirección del empleado |
| Cargo | Varchar | 50 | - | NO | cargo del empleado |
| Profesion | varchar | 50 | - | NO | Profesión del empleado |

**Propietario**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Atributos** | **Tipos de Datos** | **Amplitud** | **Llave** | **Nulo** | **Descripción** |
| ID | Int | 11 | PK | NO | Lave primaria de la tabla o CI del propietario |
| Nombre | Varchar | 30 | - | NO | Nombre del propietario |
| ApPaterno | Varchar | 20 | - | NO | Apellido paterno del propietario |
| ApMaterno | Varchar | 20 | - | NO | Apellido materno del propietario |
| Telefono | Int | 11 | - | SI | Teléfono del propietario |
| Direccion | Varchar | 50 | - | SI | Dirección del propietario |
| Celular | int | 11 | - | SI | Celular del propietario |
| Correo | Varchar | 50 | - | SI | Correo electrónico del propietario |

**especie**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Atributos** | **Tipos de Datos** | **Amplitud** | **Llave** | **Nulo** | **Descripción** |
| Nombre | Varchar | 30 | PK | NO | Lave primaria de la tabla y nombre de la raza |
| Descripcion | varchar | 50 | - | SI | Descripción de la raza |
| Especie | Varchar | 30 | FK | NO | Especie a la que pertenece la raza |

**tratamiento**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Atributos** | **Tipos de Datos** | **Amplitud** | **Llave** | **Nulo** | **Descripción** |
| ID | Int | 11 | PK | NO | Lave primaria de la tabla autoincrementable |
| Detalle | Varchar | 50 | - | NO | Diagnostico de la consulta |
| Medicacion | varchar | 50 | - | NO | Medicamentos que debe tomar el paciente |
| FechaInicio | date | - | - | NO | Fecha de inicio del tratamiento |
| Diagnostico\_ID | Int | 11 | FK | NO | Llave foránea a la tabla diagnostico |

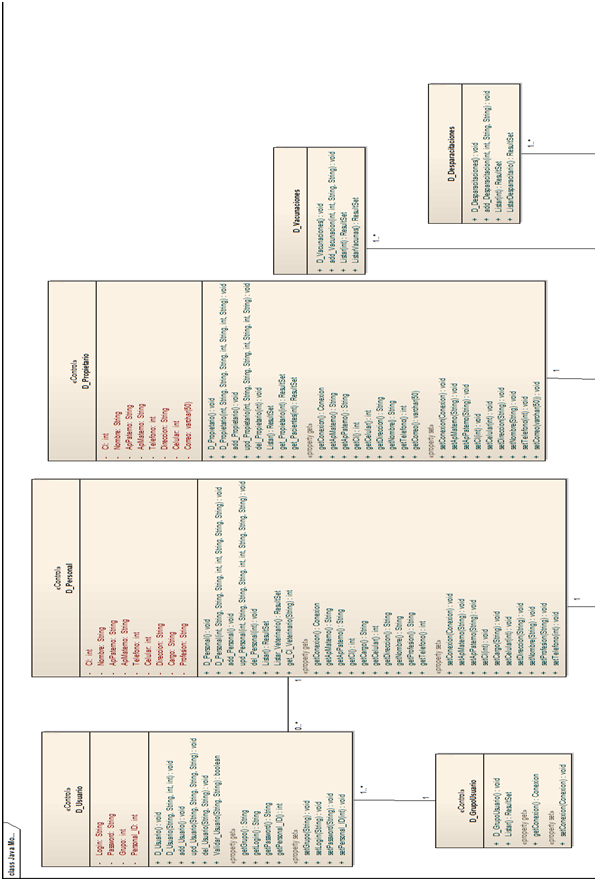
**usuario**

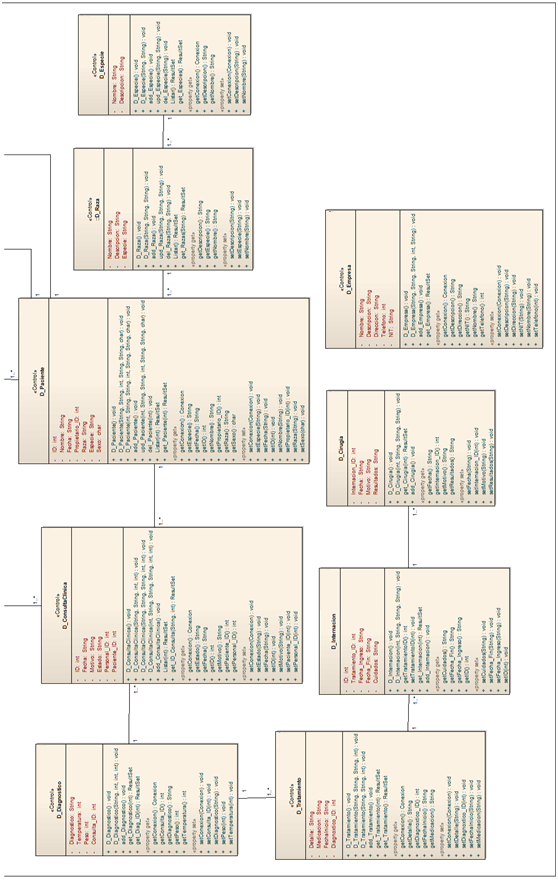
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Atributos** | **Tipos de Datos** | **Amplitud** | **Llave** | **Nulo** | **Descripción** |
| Login | Varchar | 20 | PK | NO | Nombre de la cuenta del usuario |
| Password | Varchar | 20 | - | NO | Contraseña del usuario |
| Grupo | int | 11 | FK | NO | Llave foránea a la tabla Grupo |
| Personal\_ID | Int | 11 | FK | NO | Llave foránea a la tabla personal |

**vacuna**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Atributos** | **Tipos de Datos** | **Amplitud** | **Llave** | **Nulo** | **Descripción** |
| ID | Int | 11 | PK | NO | Lave primaria de la tabla autoincrementable |
| Nombre | Varchar | 30 | - | NO | Nombre de la vacuna |
| FechaVencimento | Date | - | - | NO | Fecha de vencimiento de la vacuna |
| Indicaciones | Varchar | 100 | - | SI | Indicaciones de cuando es necesaria la vacuna |
| Cantidad | Int | 11 | - | NO | Cantidad de vacunas disponible |

* 1. **Diagrama de Clases**

****

****

* 1. **Diagramas de Comunicación**

**CU8: Envío personal de sms.**

****

**CU9: Procesar Consulta por Correo electrónico**

****

**CU12: Enviar notificaciones por correo electrónico.**



**CU22: Procesar SMS**

****

1. **Bibliografía**

* **BOOCH** Grady, Rumbaugh, Jacobson. Lenguaje Unificad de Modelado. Impreso en España: Addison Weasley, 2000.
* **GOMEZ**, Rubén. JavaMail: E-mails desde tu proyecto Java. Impreso en España: Universidad Autónoma de Madrid, 2008.
* **JACOBSON** Ivar, Booch, Rumbaugh. El proceso unificado de desarrollo de software. Impreso en España: Addison Weasley, 2000.
* **ORACLE**[en línea]: JavaMail API. <http://www.oracle.com/technetwork/java/javamail/index.html>. [Consultado 25/09/2013].
* **ORACLE** [en línea]: Communications Services Gatekeeper Patch Release Notes Release 5.0.0.1. <http://docs.oracle.com/cd/E16625_01/doc.50/e24004/pat_smpp.htm>. [Consultado 26/09/2013].
* **SANTOS,** Luz Marina. ”Gesta SMS: Librería JAVA SMS usando SMPP 3.4”. Scientia et Technica. Universidad Tecnológica de Pereira, Colombia. 2012.