

2021

Alejandro Rodríguez Moreno

Estudiante del IES Torre del Rey

alejandrordriguezmoreno@iestorredelrey.es

# PROYECTO INTEGRADO

Final FP

---

Ciclo formativo de grado superior

**Desarrollo de aplicaciones multiplataforma**

# ÍNDICE

<b>INTRODUCCIÓN</b>	<b>4</b>
<b>ESTUDIO DE VIABILIDAD</b>	<b>5</b>
<b>DESCRIPCIÓN DEL SISTEMA</b>	<b>5</b>
<b>IDENTIFICACIÓN DE REQUISITOS DEL SISTEMA</b>	<b>5</b>
REQUISITOS DE INFORMACIÓN	5
REQUISITOS FUNCIONALES	7
<b>DESCRIPCIÓN DEL LA SOLUCIÓN</b>	<b>8</b>
<b>PLANIFICACIÓN DEL PROYECTO</b>	<b>8</b>
EQUIPO DE TRABAJO	8
PLANIFICACIÓN TEMPORAL	9
<b>ESTUDIO DEL COSTE DEL PROYECTO</b>	<b>10</b>
<b>ANÁLISIS DEL SISTEMA DE INFORMACIÓN</b>	<b>12</b>
<b>IDENTIFICACIÓN DEL ENTORNO TECNOLÓGICO</b>	<b>12</b>
<b>MODELO DE DATOS</b>	<b>12</b>
MODELO ENTIDAD-RELACIÓN	13
ESQUEMA DE LA BASE DE DATOS	13
DATOS DE PRUEBA	13
<b>IDENTIFICACIÓN DE LOS USUARIOS PARTICIPANTES Y FINALES</b>	<b>13</b>
<b>IDENTIFICACIÓN DE SUBSISTEMAS DE ANÁLISIS</b>	<b>14</b>
<b>ESTABLECIMIENTO DE REQUISITOS</b>	<b>14</b>
<b>DIAGRAMAS DE ANÁLISIS</b>	<b>16</b>
<b>DEFINICIÓN DE INTERFACES DE USUARIO</b>	<b>20</b>
ESPECIFICACIÓN DE PRINCIPIOS GENERALES DE INTERFAZ	20
ESPECIFICACIÓN DE FORMATOS INDIVIDUALES DE LA INTERFAZ DE PANTALLA	20
IDENTIFICACIÓN DE PERFILES DE USUARIO	39
ESPECIFICACIÓN DE FORMATOS DE IMPRESIÓN	40
ESPECIFICACIÓN DE LA NAVEGABILIDAD ENTRE PANTALLAS	40
<b>CONSTRUCCIÓN DEL SISTEMA</b>	<b>42</b>
<b>CONCLUSIONES</b>	<b>43</b>

<b>GLOSARIO DE TÉRMINOS</b>	<b>44</b>
-----------------------------	-----------

---

<b>BIBLIOGRAFÍA</b>	<b>45</b>
---------------------	-----------

---

# Introducción

---

Este proyecto consiste en el desarrollo de una aplicación destinada a una **clínica veterinaria** que consta de tres partes diferenciadas, aplicación de escritorio que será el servidor, aplicación cliente de escritorio y aplicación cliente para smartphone.

La aplicación sirve para la gestión y almacenamiento de datos de los clientes y pacientes, y facilitar los datos clínicos de sus mascotas a sus clientes mediante una aplicación a modo de cartilla sanitaria de su mascota que podría consultar de forma mucho más cómoda y rápida en cualquier momento, además de otro modo de pedir y gestionar sus citas.

Esto hará que las clínicas veterinarias tengan un aplicación donde pueden almacenar y consultar los datos clínicos de pacientes y los datos de sus clientes, pudiendo gestionar citas de forma cómoda. Además podrá generar facturas para sus clientes con un simple clicky e imprimir informes del historial de los pacientes, algo que facilitará mucho la labor del veterinario, de modo que podrá agilizar su trabajo, ganando tiempo.

Por otro lado hará que los clientes puedan concertar citas de una forma cómoda sin tener que personarse en la clínica para ello y sin necesidad de llamar por teléfono. Esto podría conllevar que el cliente opte por la comodidad y con ello atraer más clientes.

# Estudio de viabilidad

---

## DESCRIPCIÓN DEL SISTEMA

Cómo ya he comentado, el sistema está compuesto principalmente por tres partes diferenciadas, la parte servidor que será una aplicación de escritorio, y dos partes cliente, una destinada al veterinario o los veterinarios encargados de gestionar todos los datos de clientes, pacientes, citas, etc. Otra destinada a los clientes de la clínica veterinaria en cuestión, que podrán gestionar sus citas.

## IDENTIFICACIÓN DE REQUISITOS DEL SISTEMA

### Requisitos de información

Se trabajará con una base de datos relacional en la que debemos guardar datos de los veterinarios, clientes, pacientes, citas, historial clínico y tratamientos. Por lo que será necesario el uso de un gestor de base de datos como HeidiSQL para la construcción de la base de datos de forma que lo facilite.

La tabla **veterinario** constará de *siete* campos, estos campos van a ser:

1. **ID Veterinario** → Tipo de dato **entero** y autoincremental.
2. **DNI** → Tipo de dato **cadena**, longitud 9, campo *obligatorio*.
3. **Nombre** → Tipo de dato **cadena**, campo *obligatorio*.
4. **Apellidos** → Tipo de dato **cadena**, campo *obligatorio*.
5. **Fecha de nacimiento** → Tipo de dato **fecha**, campo *opcional*.
6. **Especialidad** → Tipo de dato **cadena**, campo *opcional*.
7. **Clave** → Tipo de dato **cadena**, campo *opcional*.

La tabla **cliente** constará de *diez* campos, estos campos van a ser:

1. **ID Cliente** → Tipo de dato **entero** y autoincremental.
2. **DNI** → Tipo de dato **cadena**, longitud 9, campo *obligatorio*.
3. **Nombre** → Tipo de dato **cadena**, campo *obligatorio*.
4. **Apellidos** → Tipo de dato **cadena**, campo *obligatorio*.
5. **Dirección** → Tipo de dato **cadena**, campo *obligatorio*.

6. **Población** → Tipo de dato **cadena**, campo *obligatorio*.
7. **Teléfono** → Tipo de dato **cadena**, campo *obligatorio*.
8. **Email** → Tipo de dato **cadena**, campo *opcional*.
9. **Clave** → Tipo de dato **cadena**, campo *obligatorio*.
10. **Baja** → Tipo de dato **fecha**, campo *opcional*.

La tabla **mascota** constará de *siete* campos, estos campos van a ser:

1. **ID Mascota** → Tipo de dato **entero** y autoincremental.
2. **Número de chip** → Tipo de dato **cadena**, campo *obligatorio*.
3. **Nombre** → Tipo de dato **cadena**, campo *obligatorio*.
4. **Año de nacimiento** → Tipo de dato **entero**, campo *obligatorio*.
5. **Especie** → Tipo de dato **cadena**, campo *opcional*.
6. **Peso** → Tipo de dato **decimal**, campo *opcional*.
7. **ID Cliente** → Tipo de dato **entero**, campo *obligatorio*.

La tabla **historial clínico** constará de *siete* campos, estos campos van a ser:

1. **ID Historial clínico** → Tipo de dato **entero** y autoincremental.
2. **Juicio clínico** → Tipo de dato **cadena**, campo *obligatorio*.
3. **Fecha** → Tipo de dato **cadena**, campo *obligatorio*.
4. **Pagado** → Tipo de dato **cadena**, campo *obligatorio*.
5. **ID Cliente** → Tipo de dato **entero**, campo *obligatorio*.
6. **ID Mascota** → Tipo de dato **entero**, campo *obligatorio*.
7. **ID Veterinario** → Tipo de dato **entero**, campo *obligatorio*.

La tabla **tratamiento** constará de *tres* campos, estos campos van a ser:

1. **ID Tratamiento** → Tipo de dato **entero** y autoincremental.
2. **Descripción** → Tipo de dato **cadena**, campo *obligatorio*.
3. **Precio** → Tipo de dato **decimal**, campo *obligatorio*.

La tabla **cita** constará de *cinco* campos, estos campos van a ser:

1. **ID Cita** → Tipo de dato **entero** y autoincremental.
2. **Motivo** → Tipo de dato **cadena**, campo *obligatorio*.
3. **Fecha** → Tipo de dato **fecha**, campo *obligatorio*.
4. **ID Cliente** → Tipo de dato **entero**, campo *obligatorio*.
5. **ID Mascota** → Tipo de dato **entero**, campo *obligatorio*.

Los campos descritos son los que componen cada una de las tablas de la base de datos.

El volumen de datos que se cree necesario para un proyecto destinado a la gestión de una clínica veterinaria como es el caso, será de gran tamaño para abarcar la posibilidad de que la compañía tenga muchas clínicas abiertas en distintos puntos del país o incluso del mundo, y pueda tener toda la información almacenada y centralizada en una única base de datos.

## Requisitos funcionales

En cuanto a funcionalidades se refiere, primero cabe decir que la aplicación servidor únicamente se encargará de procesar y gestionar todas las peticiones que le lleguen de las aplicaciones cliente.

Por otro lado tenemos la aplicación android donde tenemos como principal funcionalidad iniciar sesión como usuario que previamente habría sido registrado por el veterinario.

También podrá consultar su *ID Cliente* introduciendo su DNI. Además puede modificar la contraseña.

Una vez haya iniciado sesión aparecerá la lista de citas que tiene. Tocando en cualquiera de las citas podrá acceder al detalle de cada cita donde tendrá la opción de eliminar la cita o modificarla.

En el menú podrá elegir la opción de consertar una nueva cita que le mandará a una pantalla donde deberá introducir los datos de la cita.

Finalmente tenemos la opción de salir de salir de la aplicación sin cerrar sesión y también tenemos la opción de cerrar sesión sin salir de la aplicación.

Para la aplicación de escritorio podemos iniciar sesión, consultar el ID de veterinario y modificar la contraseña. Una vez accede tendrá un menú o barra superior donde encontrará las opciones de alta cliente, consulta cliente, alta mascota, consulta mascota, consulta cita. En cada una de las pantallas de consultas tendrá a su vez opciones de eliminar o baja, modificar y generar informes, además de consultar y dar de alta en historial de una mascota o paciente.

## DESCRIPCIÓN DEL LA SOLUCIÓN

Para realizar o desarrollar las aplicaciones se ha necesitado diferentes librerías, para la aplicación servidor he utilizado las librerías de Hibernate, MySqlConnection, librerías de JasperReport y Gson para el uso de Json.

Para la aplicación cliente de escritorio se ha necesitado la librería Jcalendar para utilizar el calendario y poder guardar las fechas, la librería de JavaHelp, JhSearch e iTextPdf.

Para la app móvil también se ha usado la libería Gson para el uso de Json.

Para tener el código más ordenado y de forma que sea más fácil de entender y de localizar ciertas partes del código he estructurado todo en funciones de forma que hay bloques de códigos que realizan diferentes funcionalidades o tareas, esto también es de ayuda para localizar más rápido un error y no perderse en la inmesidad de las líneas de código que hay. Además está todo comentado de forma que rápidamente de un vistazo se entienda que hace esa parte del código.

Esto se ha hecho en las tres aplicaciones, servidor, cliente escritorio y cliente android.

También están por un lado las clases en un paquete, en otro paquete está la clase principal.

En el caso del servidor está igual pero además tiene una clase Operaciones donde se encuentran funciones para la base de datos.

## PLANIFICACIÓN DEL PROYECTO

### Equipo de trabajo

La realización del proyecto en esta ocasión no se hará en equipo si no de forma individual. Una sola persona llevará a cabo todo el proceso de documentación, planificación, realización y desarrollo de las aplicaciones del sistema descrito.

Los equipos o herramientas utilizadas para llevar acabo el desarrollo de este proyecto es un ordenador de sobremesa cuyas características son **16 GB RAM, 1 TB HDD, 256 GB SSD M.2**, y procesador **intel core i5**, este servirá para el uso de las tecnologías de trabajo más pesado para el ordenador. También se ha usado un portátil con **4 GB RAM, 500 GB HDD** y un porcesador **intel core i3**. Este último se ha usado para tareas menos pesadas que consumen menos recursos. Por último, también se han usado **máquinas**



**virtuales de smartphone** en android studio, para pruebas más simples y un **smartphone** para tareas más complejas, como las conexiones a la base de datos.

El software utilizado para el ordenador de sobremesa y para el portátil ha sido **Windows 10**, para las máquinas virtuales y el smartphone se ha utilizado **Android**.

También se han realizado particiones para la ejecución de las aplicaciones de desarrollo más pesadas (como por ejemplo **android studio**) para que se ejecuten en el disco de procesamiento más rápido.

## Planificación temporal

A continuación expongo un **SCRUM** con varios ciclos temporales de realización de las diferentes partes del proyecto:

1. **Sprint** *(8 de Marzo – 14 de Marzo)*
  - Modelo de datos.
  - Creación de la base de datos.
2. **Sprint** *(15 de Marzo – 21 de Marzo)*
  - Diseño interfaz (App Escritorio Cliente).
  - Crear la interfaz gráfica (App Escritorio Cliente).
  - Crear todas las clases y conexión a la base de datos con Hibernate (App Escritorio Cliente).
3. **Sprint** *(22 de Marzo – 28 Marzo)*
  - Registro, login, modificar contraseña y consultar ID (App Escritorio Cliente).
  - Login, modificar contraseña y consultar ID (App Android Cliente).
4. **Sprint** *(29 de Marzo – 4 de Abril)*
  - Crear las clases (App Android Cliente).
  - Crear lista de citas (App Android Cliente).
  - Crear menú (App Android Cliente).
  - Guardar datos de inicio de sesión en archivo preferences para que el cliente no tenga que iniciar sesión cada vez que se abre la aplicación (App Android Cliente).
5. **Sprint** *(5 de Abril – 11 de Abril)*
  - Crear pantalla de concertar cita y su funcionalidad (App Android Cliente).
  - Crear la pantalla de detalle de la cita (App Android Cliente).
  - Añadir funcionalidades a la pantalla detalle para eliminar y modificar una cita (App Android Cliente).

**6. Sprint** (12 de Abril – 18 de Abril)

- Añadir funcionalidad al ítem **cerrar sesión** y al ítem **salir** de la aplicación (App Android Cliente).
- Alta de clientes y alta de mascotas (App Escritorio Cliente).

**7. Sprint** (19 de Abril – 25 de Abril)

- Consulta (filtros) de clientes y consulta (filtros) de mascotas (App Escritorio Cliente).
- Añadir historial clínico y consulta historial clínico (App Escritorio Cliente).

**8. Sprint** (26 de Abril – 9 de Mayo)

- Consulta citas (App Escritorio Cliente).
- Baja clientes (App Escritorio Cliente).
- Eliminar mascota (App Escritorio Cliente).
- Modificar cliente (App Escritorio Cliente).
- Modificar mascota (App Escritorio Cliente).
- Funcionalidad de los diferentes botones de cancelar (App Escritorio Cliente).

**9. Sprint** (10 de Mayo – 16 de Mayo)

- Crear detalle de citas (App Escritorio Cliente).
- Modificar citas (App Escritorio Cliente).
- Anular citas (App Escritorio Cliente).

**10. Sprint** (17 de Mayo – 23 de Mayo)

- Generar informe del historial de una mascota (App Escritorio Cliente).
- Generar factura (App Escritorio Cliente).
- Crear la página de ayuda, información de como usar la aplicación (App Escritorio Cliente).

**11. Sprint** (24 de Mayo – 30 de Mayo)

- Probar las aplicaciones y solucionar posibles fallos o detalles por terminar (App Escritorio Cliente).

*Ver diagrama de planificación al final del documento.*

## ESTUDIO DEL COSTE DEL PROYECTO

Para conocer el coste aproximado del proyecto completo, he realizado un cálculo de las horas aproximadas que me llevará terminar completamente el proyecto. De forma general me llevará unas **12 semanas** (casi 3 meses) aproximadamente.

Pero esto es relativo, ya que el precio será en función de las horas que aunque se prolongue a 12 semanas no serán tantas horas como parece sino que debido a la falta de tiempo se prolongará en el tiempo.

Por lo que teniendo en cuenta que sería unos **25 € la hora**, y un cálculo de que el proyecto se llevará unas **88 horas en total**, el coste del proyecto sería aproximadamente de **2200 €**.

# Análisis del sistema de información

---

## IDENTIFICACIÓN DEL ENTORNO TECNOLÓGICO

Como comenté en uno de los apartados anteriores el entorno tecnológico constará de un ordenador de sobremesa, un portátil, un smartphone y máquinas virtuales.

Por otro lado las tecnologías que se van a utilizar son:

Para la aplicación servidor se usará el lenguaje de programación **Java**, con **Swing** para el desarrollo de la interfaz con ayuda del IDE **NetBeans**, además para trabajar con la base de datos se utilizará la herramienta de mapeo objeto-relacional **Hibernate**.

Para las aplicación cliente de escritorio (parte dedicada a la empresa) se usará el lenguaje de programación **Java**, con **Swing** para el desarrollo de la interfaz con ayuda del IDE **NetBeans**, también se usarán informes con **iReport** y tendrá un apartado *ayuda* para los usuarios que utilicen la aplicación a fin de que les facilite su uso, para ello se usará **JavaHelp**.

En cuanto a la aplicación cliente que está destinada a que la utilicen los clientes de la clínica veterinaria se usará **Android Studio**, donde se usará principalmente el lenguaje de programación **Java** para su desarrollo.

En cuanto a la base de datos tendrá seis tablas, éstas son *cliente*, *mascota*, *cita*, *historial clínico*, *tratamiento* y *veterinario*. Se usará una base de datos relacional con uso del lenguaje **SQL**, para ello se hará uso de MySQL y para facilitar la creación de la base de datos se hará uso del gestor de base de datos **HeidiSQL**.

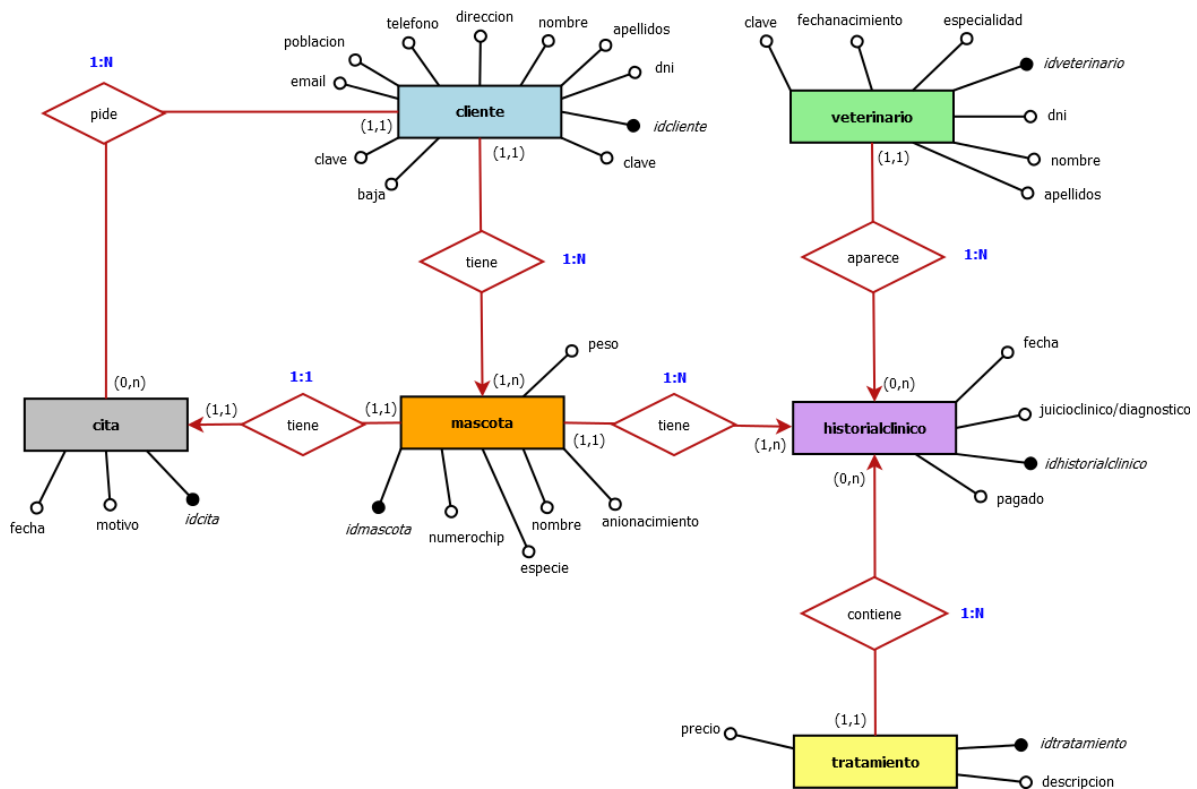
Aunque parezca un poco redundante pero a fin de dar una visión rápida de las tecnologías que se van a utilizar proporciono aquí un breve resumen a modo de lista:

También se utilizará la *herramienta CASE* **Dia** para el modelo de datos y entidad relación, casos de usos, etc

## MODELO DE DATOS

En uno de los apartados del *estudio de viabilidad* se describieron las tablas y campos de los datos para la base de datos.

## Modelo Entidad-Relación



## Esquema de la base de datos

Junto con este documento se adjuntará la estructura o esquema de la base de datos con las tablas y campos → **clinicaveterinariaESQUEMA.sql**

## Datos de prueba

Junto con este documento se adjuntará la estructura o esquema de la base de datos con las tablas y campos → **clinicaveterinariaDATOS.sql**

## IDENTIFICACIÓN DE LOS USUARIOS PARTICIPANTES Y FINALES

La aplicación cliente de escritorio tendrá un usuario que normalmente será el trabajador o veterinario de la clínica veterinaria. En principio cualquier veterinario que se encuentre en la base de datos tendrá acceso y podrá realizar las gestiones de altas de clientes, pacientes, consultas, citas, etc. También podrá consultar su ID y modificar su contraseña.

Por otro lado en la aplicación android tenemos como usuario al cliente de la clínica veterinaria. Cada cliente que haya sido dado de alta en la clínica veterinaria en cuestión tendrá asignada una contraseña que podrá modificar. Y al igual que el veterinario también podrá consultar su ID. También podrá gestionar sus citas, concertar citas, anularlas y modificarlas.

Ambas aplicaciones interactúan con otra que es el servidor, que se encarga de gestionar las peticiones, además el servidor interactúa con la base de datos.

## IDENTIFICACIÓN DE SUBSISTEMAS DE ANÁLISIS

Hay principalmente tres aplicaciones, pero podríamos decir **cuatro** subsistemas, los cuales serían, aplicación servidor, la base de datos, aplicación cliente de escritorio y aplicación android.

El subsistema servidor es el encargado de recibir y gestionar peticiones de las aplicaciones clientes y gestionarlás junto con la base de datos con la que interactúa.

El subsistema cliente escritorio es el que se encarga de gestionar altas de mascotas, clientes, historial, consultas de los mismos, gestión de citas, generar facturas, etc. Envía peticiones al servidor.

El subsistema cliente android se encarga de gestionar las citas, pudiendo dar de alta citas, eliminar y modificar las mismas. Envía peticiones al servidor encargado de la relación con la base de datos.

Finalmente tenemos el subsistema que es la base de datos, este subsistema es donde se almacenarán los datos con los que trabajarán los demás subsistemas directa o indirectamente.

Ver diagrama de **interacción** en el apartado **diagrama de análisis**.

## ESTABLECIMIENTO DE REQUISITOS

La **funcionalidad del servidor** es interactuar con las aplicaciones cliente y con la base de datos.

Después tenemos por un lado la **aplicación cliente android** donde encontramos funcionalidades como iniciar sesión, modificar contraseña y consultar ID en la pantalla principal. Después al acceder aparece la lista de citas si la hubiera, podrá tocar cada una de ellas y acceder al detalle de las mismas pudiendo anular la cita o modificarla, en cuyo caso te llevará a la pantalla de crear una cita pero esta vez para modificarla. También hay un menú clásico donde tenemos la opción de ir a la **lista de citas**, **concertar una nueva cita** que llevará a la pantalla de nueva cita, en el menú también está la opción de **cerrar sesión**, lo que llevará a la pantalla principal, y la opción **salir** que cerrará la aplicación manteniendo la sesión abierta, lo que quiere decir que al volver a abrir la aplicación no habrá que meter los datos de inicio de sesión de nuevo.

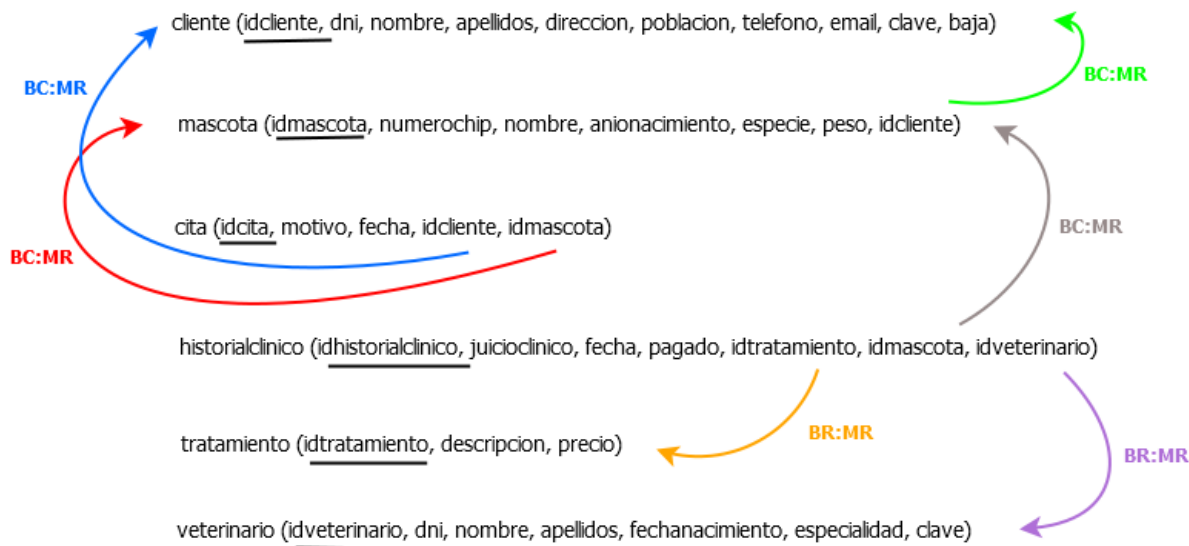
Por último las **funcionalidades de la aplicación cliente de escritorio** son:

- Iniciar sesión
- Modificar contraseña
- Consultar ID
- Dar de alta a clientes
- Dar de altas a pacientes/mascotas
- Consultar clientes con posibilidad de filtrar
- Consultar mascotas con posibilidad de filtrar
- Eliminar mascota
- Dar de baja a un cliente
- Modificar mascota
- Modificar cliente
- Añadir líneas al historial de las mascotas
- Consultar historial de las mascotas
- Generar informe del historial
- Anular citas
- Modificar citas
- Generar facturas

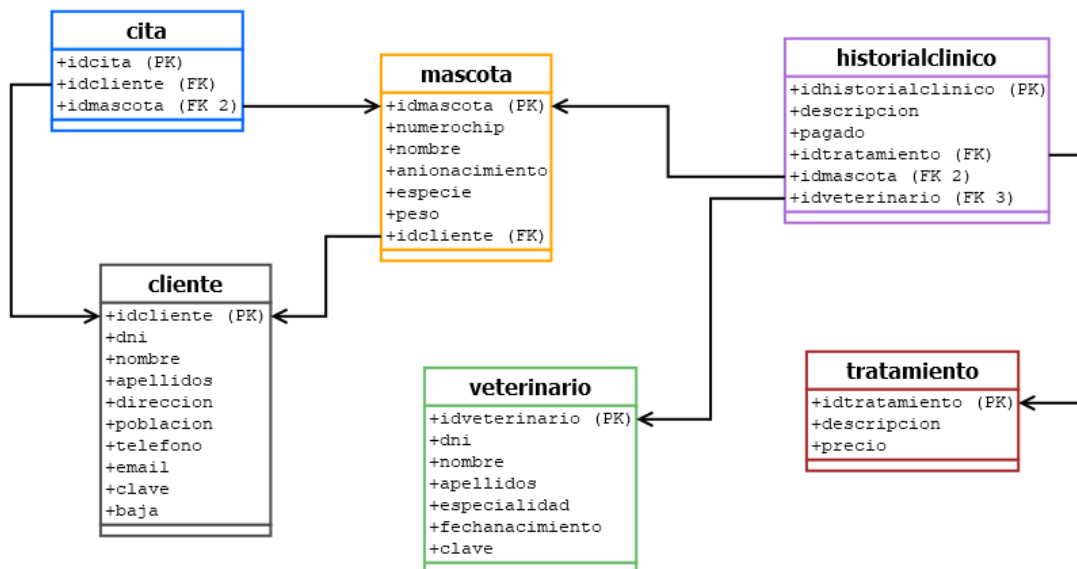
Ver diagrama de **casos de usos** en el apartado **diagrama de análisis**.

## DIAGRAMAS DE ANÁLISIS

### GRAFO RELACIONAL

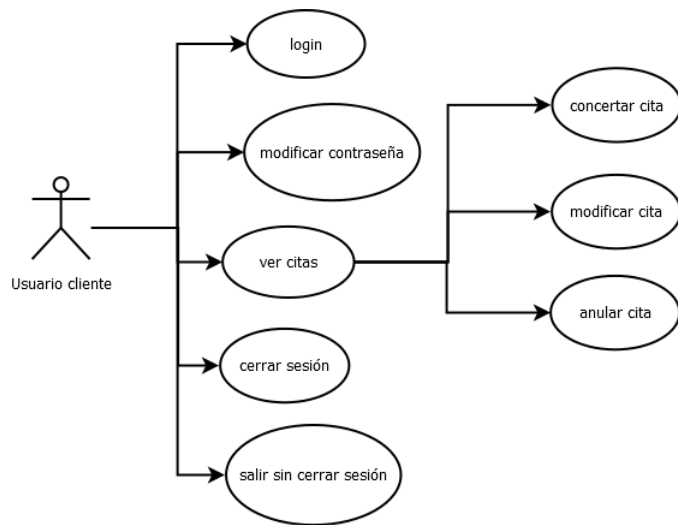


### MODELO DE DATOS

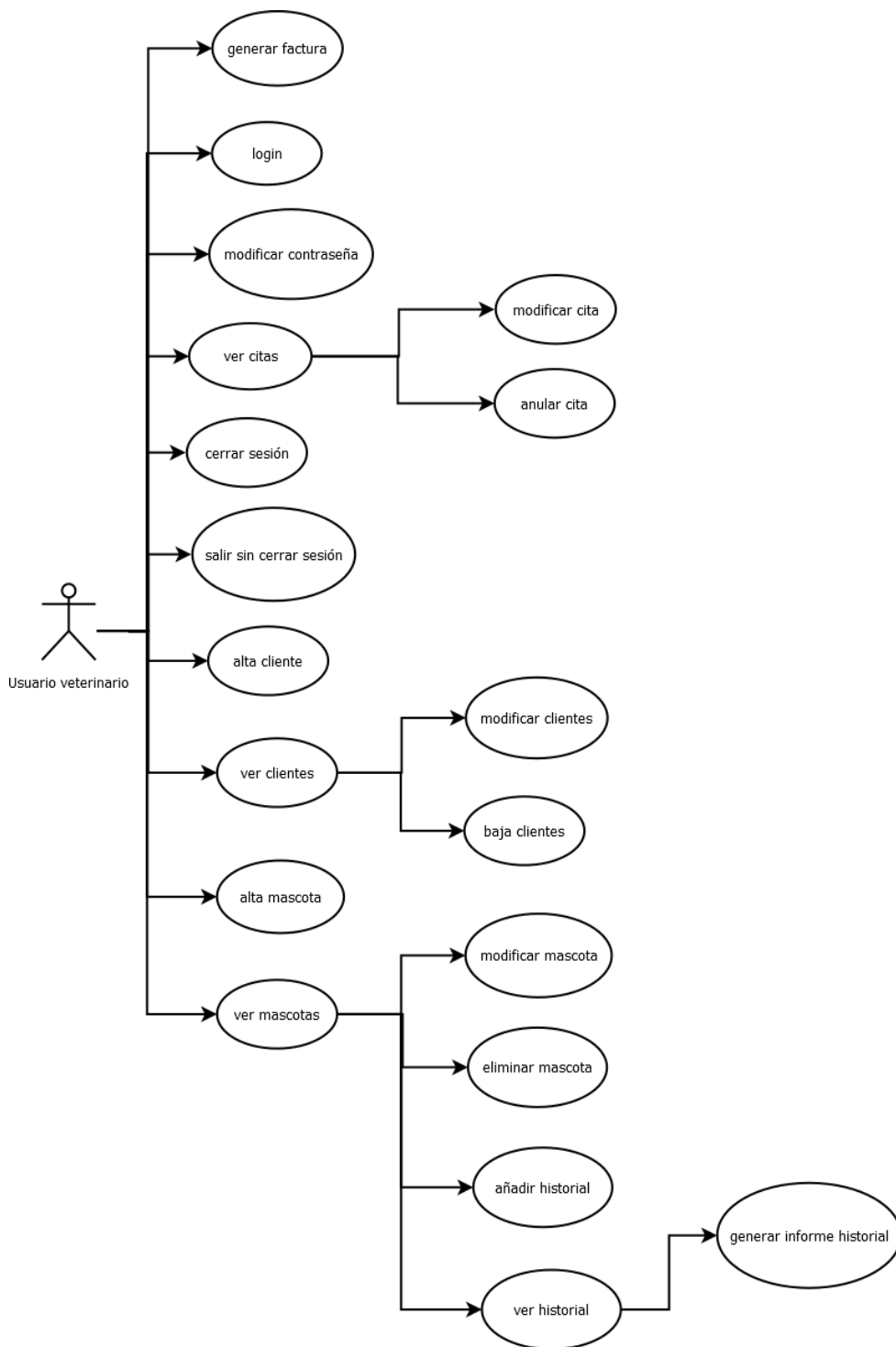




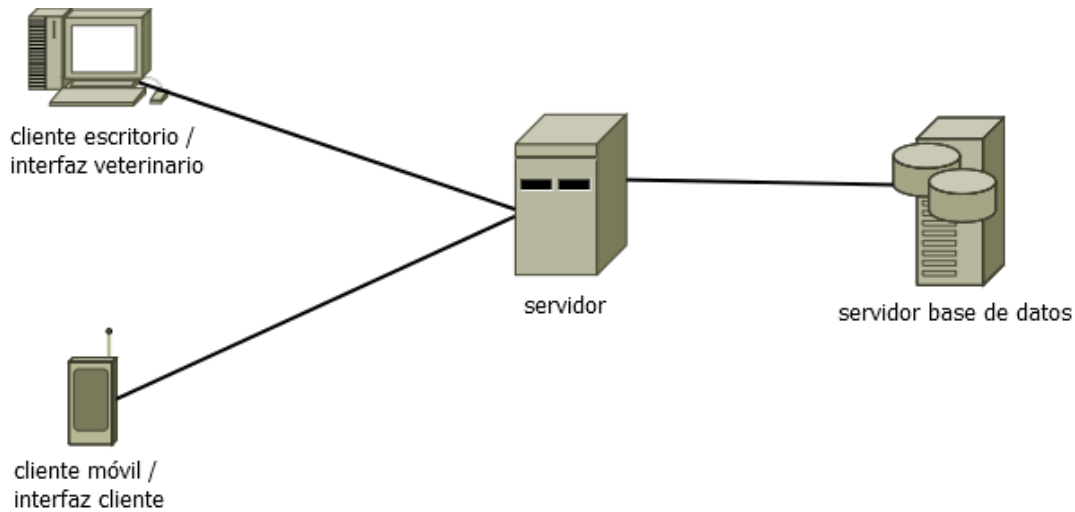
## CASOS DE USOS CLIENTE



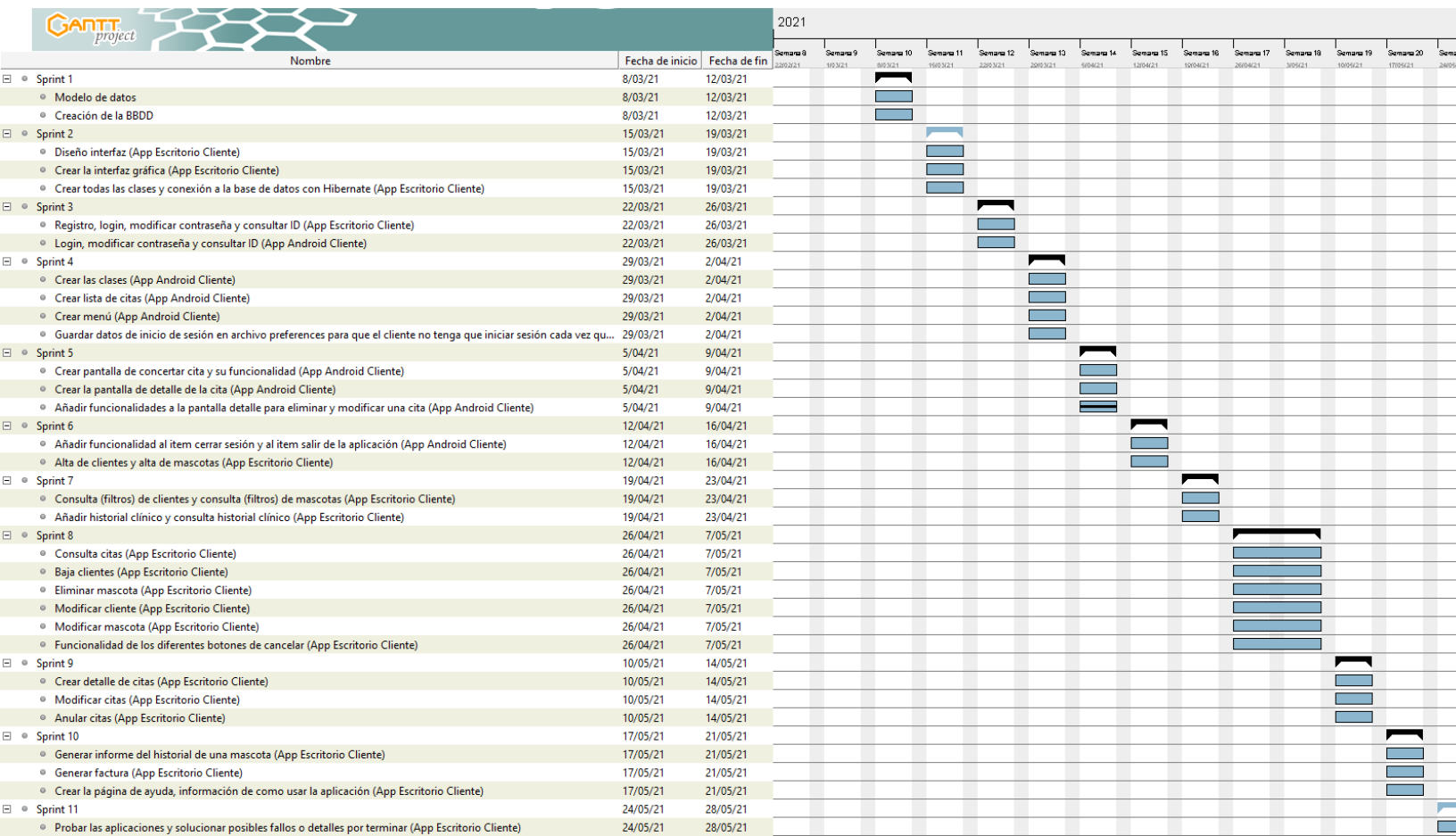
## CASOS DE USOS VETERINARIO



## INTERACCIÓN SUBSISTEMAS



## PLANIFICACIÓN (diagrama de gantt)



## DEFINICIÓN DE INTERFACES DE USUARIO

### Especificación de principios generales de interfaz

En cuanto a la interfaz de usuario en la aplicación de escritorio se optado principalmente por una única ventana donde van cambiando la pantalla (JPanel) en función de la opción elegida en el menú superior o dentro de las opciones que ofrece cada pantalla. Esta ventana no se puede redimensionar, mantiene un tamaño fijo y al iniciar la aplicación aparece centrada en el centro de la pantalla del monitor.

En muchas de las acciones como salir de la aplicación, eliminar algún registro o dar de alta, entre otros, aparecerá un mensaje informativo o de confirmación. Este mensaje aparecerá en forma de alerta, como una pequeña ventana de diálogo.

El menú superior aparecerá solamente cuando estemos en alguna de las pantallas, no en todas, por ejemplo, en la pantalla principal de inicio de sesión no estará el menú superior.

En cuanto a la distribución de los elementos se ha optado por mantenerlos centrados o alineado a la derecha en la mayoría de los casos, haciendo que se recolquen cuando alguno de los elementos se oculta o se hace visible.

Se dispone de una serie de elementos tales como botones, campos de texto, combobox, calendarios, radio botones, etc.

También incluye la posibilidad de generar informes de historial y de facturas, además de contar con un apartado de ayuda para el usuario que utilice la aplicación.

En la aplicación Android se ha optado por campos de texto, algún botón e imágenes con eventos click, también por calendario, seekbar para la elección de la hora y spinner. Además en la lista se ha jugado con los colores para saber las citas que están por llegar o que ya han pasado, lo que le proporciona un aspecto visual más amigable y es entender de un simple vistazo.

También se ha utilizado menú clásico para la navegación entre pantallas (activities).

Aparecerán mensajes de confirmación, por ejemplo, al salir de la aplicación o al eliminar una cita. También saldrán mensajes informativos a realizar cualquier acción (Toast).

### Especificación de formatos individuales de la interfaz de pantalla

## APLICACIÓN ESCRITORIO

La aplicación de escritorio consta de una única ventana (JFrame) y de varios paneles, cada panel es una pantalla distinta de la aplicación. Esto es así para que no tenga que abrirse una ventana por cada apartado de la aplicación, algo que resultaría muy molesto, de esta forma solo tendremos una ventana lo que resultará más cómodo. La ventana de la aplicación no es redimensionable si no que tiene un tamaño fijo.

## PANTALLA DE INICIO DE SESIÓN



Clínica veterinaria

# INICIO DE SESIÓN

ID:

Contraseña:

**ACCEDER** **MODIFICAR CONTRASEÑA**

[Consulta tu ID](#)

Pantalla 1

## PANTALLA DE ALTA CLIENTE

The screenshot shows a web application window titled "Clínica veterinaria". The navigation bar includes "Gestión de clientes", "Gestión de pacientes", "Gestión de citas", "Cerrar Sesión", and "Ayuda". The main content area is titled "ALTA CLIENTE" and contains a form with the following fields:

- DNI\*:
- Nombre\*:
- Apellidos\*:
- Teléfono\*:
- Dirección\*:
- Población\*:
- Email (opcional):

At the bottom of the form are two buttons: "GUARDAR Y DAR DE ALTA MASCOTA" and "CANCELAR".

Pantalla 2

## PANTALLA DE CONSULTA CLIENTE

Clínica veterinaria

Gestión de clientes Gestión de pacientes Gestión de citas Cerrar Sesión Ayuda

## CONSULTA CLIENTE

☐ Nombre
 ☐ Apellidos
 ☐ Población
 ☐ Datos de baja
 ☒ Listado completo

**FILTRAR**

**BAJA**
**MODIFICAR**
**GENERAR FACTURA**
**IMPRIMIR LISTA**

Fecha de baja:

ID	DNI	Nombre	Apellidos
1001	9514429818	Matthaeus	Lott
1002	8488672055	Lauralee	Hold
1003	6939907645	Gretta	Peirone
1004	8964707141	Dorry	Golton
1005	6039439459	Moishe	Partner
1006	9743875441	Harlan	Piggins
1007	7062394025	Gisella	O'Towey
1008	0215430026	Kamillah	Shafier
1009	3452500683	Tabbi	Gavan
1010	2539947581	Catherine	Phethean
1011	4671875934	Kalila	Bosson
1012	5043827726	Samara	Alway
1013	9195264949	Saree	Aimable
1014	4538843648	Reverend	Abrahamson

Pantalla 3



## PANTALLA DE GENERAR FACTURA

Clínica veterinaria

## FACTURAR

Empresa:

Dirección:

Ciudad:

Teléfono:

Elige el día a facturar:

Pantalla 4

## PANTALLA DE ALTA MASCOTA

The screenshot shows a web application window titled 'Clínica veterinaria'. The navigation bar includes links for 'Gestión de clientes', 'Gestión de pacientes', 'Gestión de citas', 'Cerrar Sesión', and 'Ayuda'. The main content area is titled 'ALTA MASCOTA' and contains a form with the following fields:

- Número de chip\*:
- Nombre\*:
- Año de nacimiento\*:
- Especie (opcional):
- Peso (opcional):
- DNI Cliente\*:

At the bottom of the form are two buttons: 'GUARDAR' and 'CANCELAR'.

Pantalla 5

## PANTALLA DE CONSULTA MASCOTA

Clínica veterinaria

Gestión de clientes Gestión de pacientes Gestión de citas Cerrar Sesión Ayuda

## CONSULTA MASCOTA

☐ Lista completa ☒ Por clientes

148011081-7

FILTRAR CONSULTA HISTORIAL AÑADIR A HISTORIAL BAJA

ID	Numero de chip	Nombre
9	738F6	Virgo

Pantalla 6

## PANTALLA DE ALTA HISTORIAL

Clínica veterinaria

ID MASCOTA: 9

Juicio clínico:

cualquier cosa...

Tratamiento: 1.escayolar pata ▼

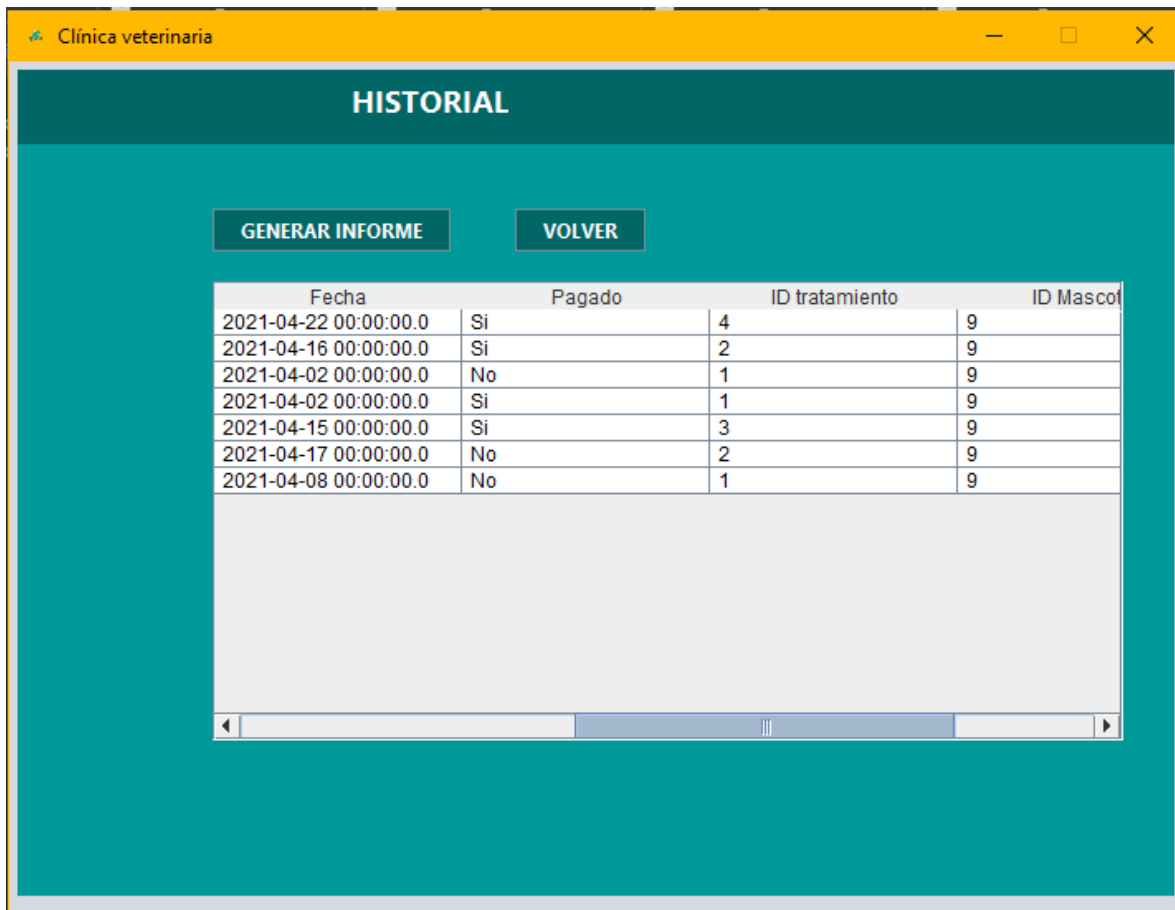
DNI veterinario: 1.García Sánchez, Juan (47867522G) ▼

Pagado: ☐ SI ☒ NO

GUARDAR CANCELAR

Pantalla 7

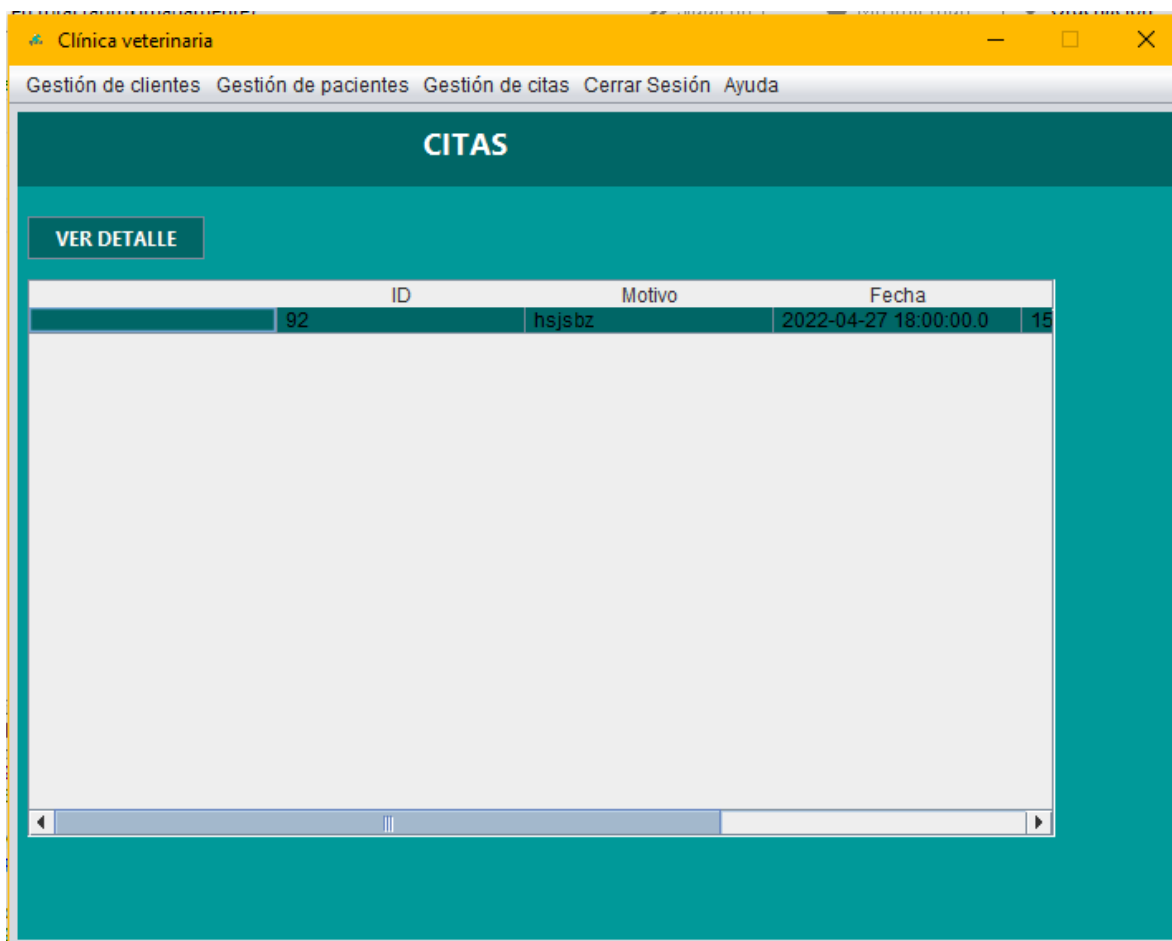
## PANTALLA DE CONSULTA HISTORIAL



Fecha	Pagado	ID tratamiento	ID Mascota
2021-04-22 00:00:00.0	Si	4	9
2021-04-16 00:00:00.0	Si	2	9
2021-04-02 00:00:00.0	No	1	9
2021-04-02 00:00:00.0	Si	1	9
2021-04-15 00:00:00.0	Si	3	9
2021-04-17 00:00:00.0	No	2	9
2021-04-08 00:00:00.0	No	1	9

Pantalla 8

## PANTALLA DE CONSULTA CITA



Pantalla 9

## PANTALLA DE DETALLE CITA



Clínica veterinaria

## DETALLE CITA

**ID:** 92

**Motivo:** hsjsbz

**Cliente:** 15

**Mascota:** 9

**Fecha:** 2022-04-27 18:00:00.0

**ANULAR CITA** **MODIFICAR FECHA** **CANCELAR**

Pantalla 10

## APLICACIÓN ANDROID

La aplicación android consta de varias actividades, de entre las cuáles está la MainActivity que es la pantalla de inicio, la de inicio de sesión y la de modificar la contraseña.

La ListaCitaActivity que es la pantalla del listado de citas, la DetalleActivity que es la pantalla del detalle de la cita, y la NuevaActivity que es la pantalla de concertar una nueva cita.

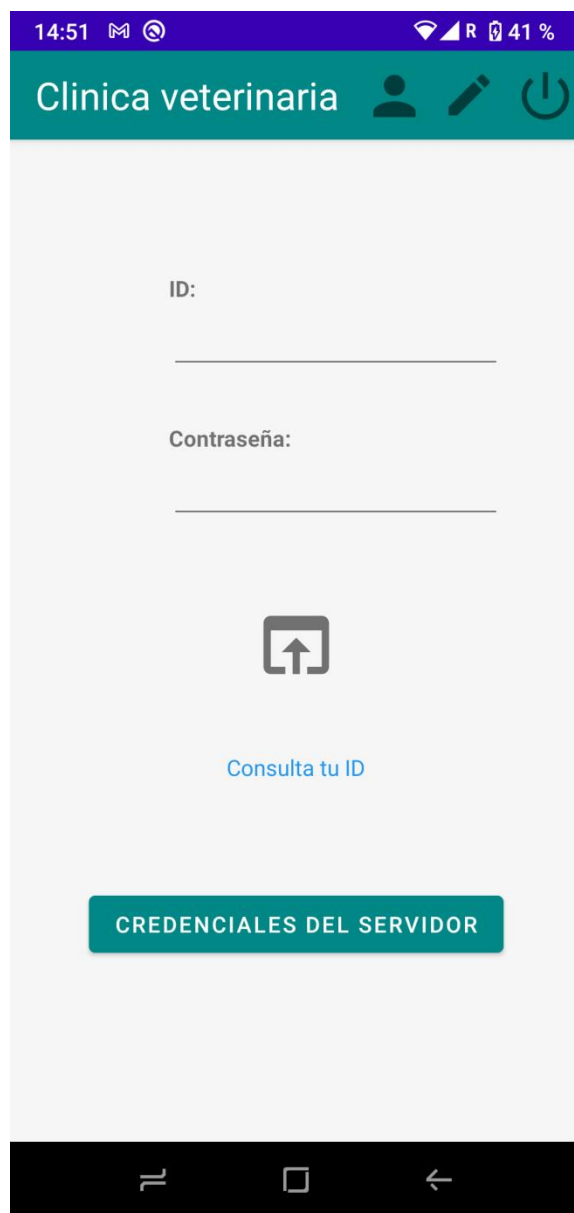
### PANTALLA DE INICIO



Pantalla 11



## PANTALLA DE INICIO DE SESIÓN



The screenshot shows a mobile application interface for 'Clinica veterinaria'. At the top, there is a status bar with the time 14:51, signal strength, and battery level at 41%. Below the status bar is a teal header with the text 'Clinica veterinaria' and three icons: a person, a pencil, and a power button. The main area is light gray and contains two input fields labeled 'ID:' and 'Contraseña:'. Below these fields is a square icon with an upward arrow. Underneath the icon is a blue link that says 'Consulta tu ID'. At the bottom of the main area is a teal button with the text 'CREDENCIALES DEL SERVIDOR'. The bottom of the screen shows a black navigation bar with three icons: a home button, a square, and a back arrow.

Pantalla 12

## PANTALLA DE CAMBIO CONTRASEÑA

14:51 M 41 %

Clinica veterinaria

ID:

Contraseña:

Nueva contraseña:

Consulta tu ID

CREDENCIALES DEL SERVIDOR

Pantalla 13

## PANTALLA DE LISTADO DE CITAS

14:51 M Q

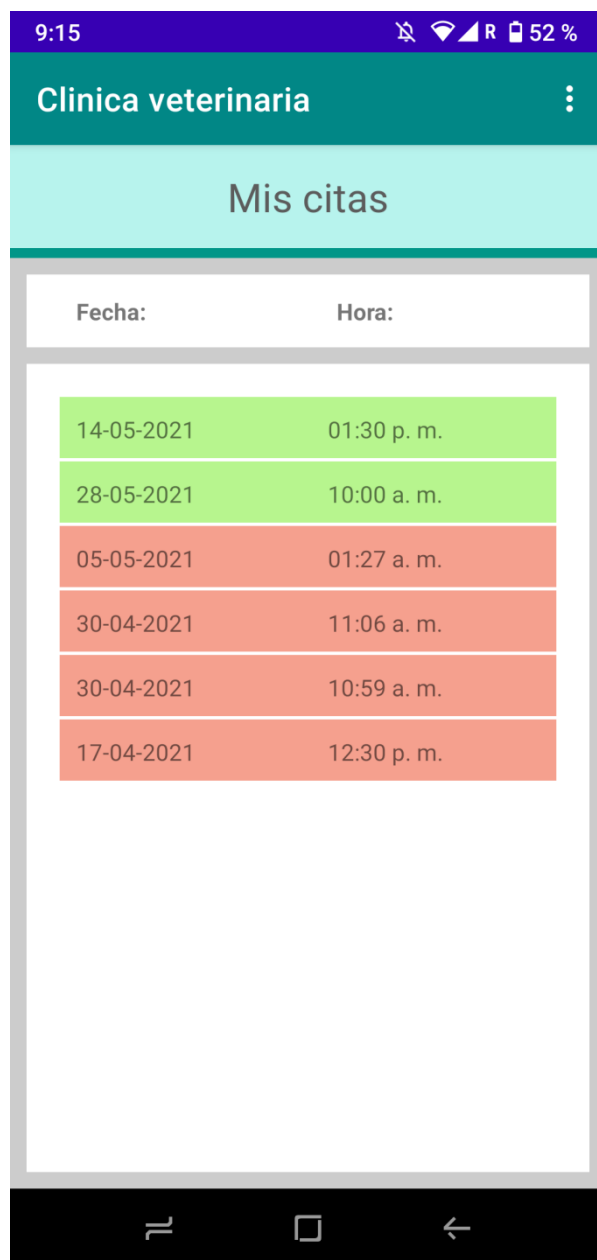
Clínica veterinaria

Introduce la IP del servidor

ACEPTAR

Pantalla 14

## PANTALLA DE LISTADO DE CITAS



Fecha:	Hora:
14-05-2021	01:30 p. m.
28-05-2021	10:00 a. m.
05-05-2021	01:27 a. m.
30-04-2021	11:06 a. m.
30-04-2021	10:59 a. m.
17-04-2021	12:30 p. m.

Pantalla 15

## PANTALLA DE CONCERTAR CITA

The screenshot shows a mobile application interface for booking a veterinary appointment. At the top, a status bar displays the time 9:23, signal strength, Wi-Fi, and battery level at 59%. Below this is a teal header with the text 'Clínica veterinaria' and a three-dot menu icon. The main form has two sections: 'Motivo:' with a text input field containing 'manchas en la piel', and 'Mascota:' with a dropdown menu showing 'Riki'. Below the form is a calendar for May 2021, with the date 27 selected. Under the calendar is a time slider set to 11:00. At the bottom is a teal button labeled 'CONCERTAR CITA'. The entire app is shown on a black background with a white Android navigation bar at the very bottom.

9:23 59 %

Clínica veterinaria

Motivo:  
manchas en la piel

Mascota:  
Riki

< Mayo de 2021 >

L	M	X	J	V	S	D
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30

Hora: 11:00

CONCERTAR CITA

Pantalla 16

## PANTALLA DE DETALLE DE CITA



Pantalla 17

## SERVIDOR

La pantalla del servidor consta simplemente de una pantalla con una barra superior que indica si el servidor está encendido o apagado y un botón para encender el servidor.

### PANTALLA DEL SERVIDOR



Pantalla 18

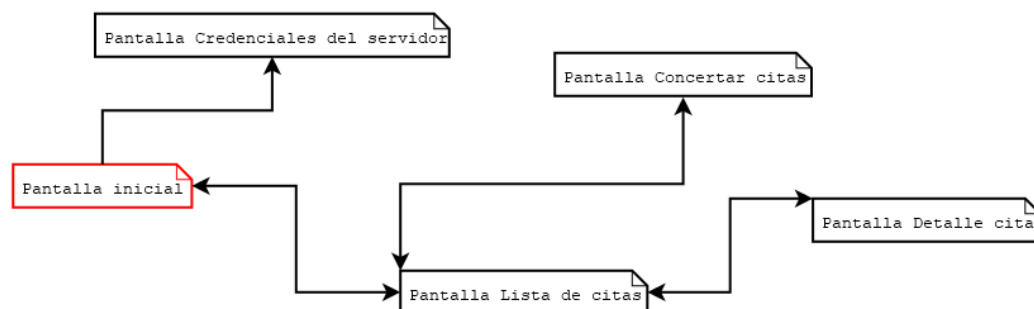
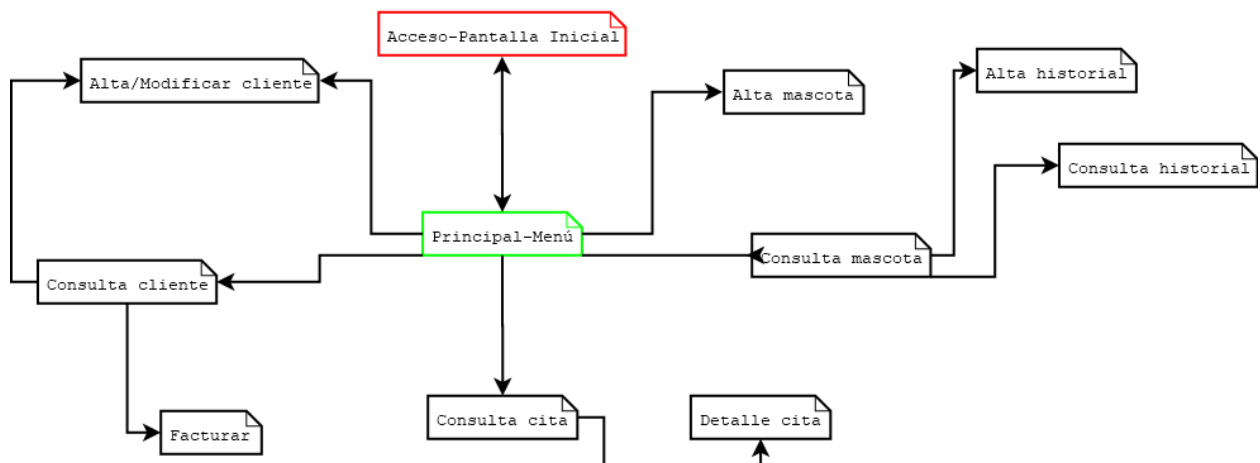
## Identificación de perfiles de usuario

Los dos perfiles de usuario existentes son el del veterinario y del cliente, pudiendo acceder el veterinario a cualquier pantalla de la aplicación de escritorio y el cliente a cualquier pantalla de la aplicación para móvil.

## Especificación de formatos de impresión

En la aplicación cliente de escritorio tenemos la opción de imprimir los datos del historial de una mascota y las facturas, esto se visualizará en formato pdf, esto se ha realizado con la herramienta iReport para Java, también se puede imprimir el listado de clientes filtrado por alguna de las opciones que tiene, este sin embargo también será en pdf pero usando otra herramienta Java llamada iTextPdf, esta aunque con un aspecto visual menos llamativo la he usado para aprovechar y aprender el funcionamiento de otras herramientas.

## Especificación de la navegabilidad entre pantallas







# Construcción del sistema

---

## Proyecto final FP:

- [https://github.com/alexproject8/Clinica\\_Veterinaria\\_Android](https://github.com/alexproject8/Clinica_Veterinaria_Android)
- [https://github.com/alexproject8/Clinica\\_Veterinaria\\_Servidor\\_Escritorio](https://github.com/alexproject8/Clinica_Veterinaria_Servidor_Escritorio)
- [https://github.com/alexproject8/Clinica\\_Veterinaria\\_Cliente\\_Escritorio](https://github.com/alexproject8/Clinica_Veterinaria_Cliente_Escritorio)

# Conclusiones

---

Para terminar cabe decir que durante toda la realización del proyecto y también de las prácticas he aprendido a realizar un proyecto con todos los pasos que conlleva y que hay que realizar hasta llegar al final del proyecto. Durante este período he aprendido a solucionar problemas autoaprendiendo de manera más fácil, he aprendido más en profundidad o con más detalle ciertas cosas que antes no me quedaban claras y que ahora puedo entender todo su funcionamiento gracias a la realización de un proyecto completo.

Si es verdad que me habría gustado añadir alguna funcionalidad extra más para que realmente fuera una aplicación completa, ya que entre las prácticas y el proyecto no disponía de tiempo suficiente para poder abarcarlo todo. De todas formas estoy contento porque se ha realizado todo lo que se pedía y aunque no le he podido dedicar mucho tiempo al aspecto visual me ha venido muy bien realizar un proyecto en el que he tenido que desarrollar no solo una aplicación de escritorio sino también una aplicación android con lo que me he enfrentado a problemas que el futuro podré resolver de manera más sencilla, además de que antes veía android algo más complejo, pero después de haber realizado el proyecto aunque no deja de ser complejo veo que puedo desenvolverme y entender mucho mejor ciertos conceptos.

Aunque en menor medida también me ha pasado lo mismo con la parte de escritorio.

Se podría decir que el proyecto me ha servido para mejorar como desarrollador y aprender a solucionar problemas por mi cuenta, además de para que junto con las prácticas hayan aumentado mis conocimientos.

# Glosario de términos

---

**JasperReport:** librería utilizada para la realización de informes en Java.

**iTextPdf:** librería para crear informes o algún tipo de documento como un pdf, docx, etc.

**Hibernate:** Herramienta de mapeo objeto-relacional (ORM) para Java que facilita una base de datos relacional y el modelo de objetos de una aplicación.

**SCRUM:** proceso de gestión que reduce la complejidad en el desarrollo de productos.

**JavaHelp:** expansión de Java que facilita la programación de las ventanas de ayudas.

**JSON:** formato de texto para el intercambio de datos.

# Bibliografía

---

<https://docs.oracle.com/javase/7/docs/api/index.html>

<https://developer.android.com/guide?hl=es>

Algunos foros.