



ESCUELA SUPERIOR DE INGENIERÍA

Ingeniería Técnica en Informática de
Gestión

Aplicación de Internet Enriquecida para la
Gestión Informática de Clínicas
Veterinarias

Curso 2012-2013

Agustín Álvarez García

Cádiz, 5 de Abril de 2013



ESCUELA SUPERIOR DE INGENIERÍA

Ingeniería Técnica en Informática de
Gestión

Aplicación de Internet Enriquecida para la
Gestión Informática de Clínicas
Veterinarias

DEPARTAMENTO: Ingeniería Informática.

DIRECTOR DEL PROYECTO: Juan Boubeta Puig.

AUTOR DEL PROYECTO: Agustín Álvarez García.

Cádiz, 5 de Abril del 2013

Fdo.: Agustín Álvarez García

Índice general

1. Introducción	9
1.1. Objetivos	9
1.2. Alcance	10
1.3. Visión general	10
1.4. Glosario	11
1.4.1. Acrónimos	11
1.4.2. Definiciones	12
2. Estado del arte	13
2.1. GWT	13
2.2. Flex	14
2.3. Silverlight	14
2.4. ZK	14
3. Desarrollo del calendario	17
3.1. Fases	17
3.1.1. Elicitación de requisitos	17
3.1.2. Análisis	17
3.1.3. Diseño	17
3.1.4. Implementación	17
3.1.5. Verificación	18
3.1.6. Documentación	18
3.2. Diagrama de Gantt	18
4. Descripción general del proyecto	21
4.1. Perspectiva del producto	21
4.1.1. Entorno del producto	21
4.1.2. Interfaz de usuario	21
4.2. Funciones	21
4.3. Características del usuario	22
4.4. Restricciones generales	22
4.4.1. Control de versiones	22
4.4.2. Lenguajes de programación y tecnologías	23

4.4.3.	Herramientas	23
4.4.4.	Sistemas operativos y hardware	24
5.	Desarrollo del proyecto	25
5.1.	Modelo de ciclo de vida	25
5.2.	Requisitos	25
5.2.1.	Funcionales	25
5.2.2.	De información	29
5.2.3.	Reglas de negocio	44
5.2.4.	Interfaz	45
5.2.5.	No funcionales	46
5.3.	Análisis del sistema	47
5.3.1.	Casos de uso	47
5.3.2.	Modelo conceptual de datos del dominio	122
5.3.3.	Diagramas de secuencia	127
5.4.	Diseño del sistema	130
5.4.1.	Arquitectura	130
5.4.2.	Diseño conceptual de la base de datos	134
5.4.3.	Diseño lógico de la base de datos	160
5.5.	Implementación	167
5.5.1.	Inserción, actualización y eliminación de elementos	167
5.5.2.	Generación de gráficas	168
5.5.3.	Conexión con la base de datos	169
5.5.4.	Consulta a la base de datos	169
5.5.5.	Listado de elementos	170
5.5.6.	Diseño físico de la base de datos	171
5.6.	Pruebas y validación	181
5.6.1.	Plan de pruebas	181
5.6.2.	Especificación del diseño de pruebas	182
6.	Resumen	185
6.1.	Descripción general	185
6.2.	Características generales	185
6.3.	Objetivos	185
7.	Conclusiones y trabajo futuro	187
7.1.	Conclusiones	187
7.2.	Valoración personal	187
7.3.	Trabajo futuro	188
8.	Agradecimientos	189

A. Manual de instalación	191
A.1. Instalación en el lado del cliente	191
A.2. Instalación en el lado del servidor	191
A.2.1. JRE (Java Runtime Environment)	191
A.2.2. MySQL	192
A.2.3. Apache Tomcat 7.0	192
B. Manual de gestión de copias de seguridad	195
B.1. Copia de seguridad de la base de datos	195
B.2. Restauración de la copia de seguridad de la base de datos	198
B.3. Copia de seguridad de los ficheros	198
B.4. Restauración de la copia de seguridad de ficheros	200
C. Manual de usuario	201
C.1. Pantalla de acceso	201
C.2. Pantalla principal	201
C.3. Gestión de clientes	201
C.3.1. Nuevo cliente	202
C.3.2. Eliminar cliente	202
C.3.3. Filtrar clientes	202
C.3.4. Asignar mascota	206
C.3.5. Asignar cita	206
C.3.6. Ver perfil del cliente	208
C.4. Gestión de mascotas	209
C.4.1. Ver perfil de mascota	211
C.4.2. Eliminar cliente	216
C.5. Gestión de productos	216
C.5.1. Nueva familia de productos	216
C.5.2. Nuevo producto	218
C.5.3. Modificar producto	218
C.5.4. Filtrar productos	219
C.5.5. Eliminar producto	219
C.6. Gestión de servicios	219
C.6.1. Nueva familia de servicios	219
C.6.2. Nuevo servicio	219
C.6.3. Filtrar servicios	219
C.6.4. Modificar servicio	221
C.6.5. Eliminar servicio	221
C.7. Gestión de proveedores	221
C.7.1. Nuevo proveedor	221
C.7.2. Filtrar proveedores	221

Índice general

C.7.3. Modificar proveedor	221
C.7.4. Eliminar proveedor	223
C.8. Gestión de pedidos	223
C.8.1. Nuevo pedido	223
C.8.2. Modificar pedido	224
C.8.3. Filtrar pedidos	224
C.8.4. Eliminar pedido	224
C.9. Gestión de ventas	224
C.9.1. Nueva venta	224
C.9.2. Generar albarán	225
C.9.3. Modificar venta	225
C.9.4. Filtrar venta	227
C.9.5. Eliminar venta	227
C.10. Gestión de facturas	227
C.10.1. Nueva factura	227
C.10.2. Generar factura	229
C.10.3. Descargar factura	229
C.10.4. Modificar factura	229
C.10.5. Filtrar factura	229
C.10.6. Eliminar factura	230
C.11. Consultar estadísticas	230
C.11.1. Estadísticas Mascota	230
C.11.2. Estadísticas Ingresos/Gastos	230
C.12. Gestión de usuarios	232
C.12.1. Nuevo usuario	232
C.12.2. Modificar usuario	232
C.12.3. Eliminar usuario	232
C.12.4. Modificar contraseña	235
D. Documentos generados por el sistema	237
Bibliografía	241

Índice de figuras

3.1. Diagrama de Gantt	19
5.1. Diagrama de Casos de Uso 1	47
5.2. Diagrama de Casos de Uso 2	51
5.3. Diagrama Gestión de Clientes	52
5.4. Diagrama Gestión de Mascotas	57
5.5. Diagrama Gestión de Proveedores	64
5.6. Diagrama Gestión de Pedidos	75
5.7. Diagrama Gestión de Productos	76
5.8. Diagrama Gestión de Usuarios	82
5.9. Diagrama Gestión de Ventas	85
5.10. Diagrama Gestión de Citas	95
5.11. Diagrama Gestión de Facturas	106
5.12. Diagrama Gestión de Vacunas	109
5.13. Diagrama Gestión de Historiales	114
5.14. Diagrama de clases conceptuales I	123
5.15. Diagrama de clases conceptuales II	124
5.16. Diagrama de clases conceptuales III	125
5.17. Diagrama de clases conceptuales completo	126
5.18. Diagrama Secuencia UC-01 Alta Cliente	127
5.19. Diagrama Secuencia UC-02 Baja Cliente	127
5.20. Diagrama Secuencia UC-03 Filtrar Cliente	127
5.21. Diagrama Secuencia UC-04 Modificar Cliente	128
5.22. Diagrama Secuencia UC-05 Listar Cliente	128
5.23. Diagrama Secuencia UC-06 Ver Mascotas Cliente	128
5.24. Diagrama Secuencia UC-07 Ver Citas Cliente	129
5.25. Diagrama Secuencia UC-08 Ver Ventas Cliente	129
5.26. Diagrama Secuencia UC-21 Registrsrar Nuevo Pedido	130
5.27. Diagrama Secuencia UC-22 Cambiar Estado Pedido	130
5.28. Diagrama Secuencia UC-36 Registrar Nueva Venta	131
5.29. Diagrama Secuencia UC-41 Generar Albarán	131
5.30. Diagrama Secuencia UC-42 Descargar Albaran	131
5.31. Diagrama Secuencia UC-49 Crear Nueva Factura	132

Índice de figuras

5.32. Arquitectura ZK Bind	133
5.33. Arquitectura MVVM	134
5.34. Diagrama Entidad-Relación	137
B.1. Seleccionar Base de Datos	196
B.2. Seleccionar Exportar	196
B.3. Pantalla Exportar	197
B.4. Pantalla Importar	199
C.1. Pantalla de acceso	202
C.2. Pantalla principal	203
C.3. Cliente nuevo	204
C.4. Cliente eliminar	204
C.5. Cliente filtrado	205
C.6. Cliente botón derecho	206
C.7. Cliente asignar mascota	207
C.8. Cliente asignar cita	208
C.9. Cliente perfil	210
C.10. Cliente botón cancelar	211
C.11. Mascota botón derecho	211
C.12. Mascota perfil	213
C.13. Mascota historial	214
C.14. Mascota historial imagen	215
C.15. Nueva familia de productos	217
C.16. Producto modificar	218
C.17. Nueva familia de servicios	220
C.18. Proveedor modificar	222
C.19. Venta nueva	226
C.20. Factura línea	228
C.21. Factura mensaje alerta	229
C.22. Estadísticas de Mascota	231
C.23. Estadísticas de Ingresos/Gastos	233
C.24. Usuario nuevo	234
C.25. Usuario submenú	235
C.26. Usuario modificar contraseña	236
D.1. Ejemplo de albarán	238
D.2. Ejemplo de factura	239

Índice de tablas

4.1. Sistema Operativo y Hardware	24
5.1. FRQ-001 Gestión de clientes	25
5.2. FRQ-002 Gestión de mascotas	26
5.3. FRQ-003 Gestión de proveedores	26
5.4. FRQ-004 Gestión de productos	26
5.5. FRQ-005 Gestión de ventas	26
5.6. FRQ-006 Gestión de historiales	26
5.7. FRQ-007 Gestión de citas	27
5.8. FRQ-008 Gestión de facturas	27
5.9. FRQ-009 Gestión de vacunas	27
5.10. FRQ-010 Gestión de usuarios	28
5.11. FRQ-011 Creación de albaranes	28
5.12. FRQ-012 Modificación contraseña de acceso	28
5.13. FRQ-013 Creación automática de recordatorios	28
5.14. FRQ-014 Creación automática de código de producto	28
5.15. FRQ-015 Creación automática de código servicio	29
5.16. IRQ-01 Cita	29
5.17. IRQ-02 Cliente	29
5.18. IRQ-03 Especie	31
5.19. IRQ-03 Tabla especie	31
5.20. IRQ-04 Factura	31
5.21. IRQ-05 Ficheros	32
5.22. IRQ-06 Historial	32
5.23. IRQ-07 IVA	33
5.24. IRQ-08 Mascota	33
5.25. IRQ-10 Mascota-Vacuna	34
5.26. IRQ-11 Pedido	34
5.27. IRQ-12 Pedido Línea	35
5.28. IRQ-13 Peso	35
5.29. IRQ-14 Población	36
5.30. IRQ-15 Producto	36
5.31. IRQ-16 Producto Familia	37

Índice de tablas

5.32. IRQ-17 Proveedor	37
5.33. IRQ-18 Provincia	39
5.34. IRQ-19 Raza	39
5.35. IRQ-20 Servicio	40
5.36. IRQ-21 Servicio Familia	40
5.37. IRQ-21 Usuario	41
5.38. IRQ-22 Vacuna	41
5.39. IRQ-23 Venta	43
5.40. IRQ-24 Venta Línea	43
5.41. RNRQ-01 Contraseñas seguras y cifradas	44
5.42. IFRQ-01 Consistencia en la apariencia	45
5.43. IFRQ-02 Interfaz amigable e intuitiva	45
5.44. NFRQ-01 Fiabilidad	46
5.45. NFRQ-02 Multiplataforma	46
5.46. NFRQ-03 Rendimiento	46
5.47. UC-01 Alta Cliente	47
5.48. UC-02 Baja Cliente	48
5.49. UC-03 Filtrar Cliente	49
5.50. UC-04 Modificar Cliente	53
5.51. UC-05 Listar Clientes	54
5.52. UC-06 Ver Mascotas del Cliente	54
5.53. UC-07 Ver Citas del Cliente	55
5.54. UC-08 Ver Ventas del Cliente	55
5.55. UC-09 Alta Mascota	57
5.56. UC-10 Baja Mascota	58
5.57. UC-11 Filtrar Mascota	59
5.58. UC-12 Modificar Mascota	60
5.59. UC-13 Listar Mascotas	61
5.60. UC-14 Consultar Pesos	61
5.61. UC-15 Registrar Nuevo Peso	62
5.62. UC-16 Alta Proveedor	64
5.63. UC-17 Baja Proveedor	65
5.64. UC-18 Filtrar Proveedor	66
5.65. UC-19 Modificar Proveedor	68
5.66. UC-20 Listar Proveedores	69
5.67. UC-21 Registrar Nuevo Pedido	69
5.68. UC-22 Cambiar Estado Pagado	71
5.69. UC-23 Modificar Pedido	71
5.70. UC-24 Filtrar Pedido	72
5.71. UC-25 Listar Pedidos	73

5.72. UC-26 Eliminar Pedido	73
5.73. UC-27 Alta Nuevo Producto	76
5.74. UC-28 Baja Producto	78
5.75. UC-29 Modificar Producto	78
5.76. UC-30 Filtrar Producto	79
5.77. UC-31 Listar Productos	80
5.78. UC-32 Alta Nuevo Usuario	80
5.79. UC-33 Baja Usuario	83
5.80. UC-34 Modificar Datos Usuario	83
5.81. UC-35 Listar Usuarios	84
5.82. UC-36 Registrar Nueva Venta	86
5.83. UC-37 Modificar Venta	88
5.84. UC-38 Filtrar Venta	89
5.85. UC-39 Listar Ventas	90
5.86. UC-40 Eliminar Venta	90
5.87. UC-41 Generar Albarán	92
5.88. UC-42 Descargar Albarán	92
5.89. UC-43 Alta Nueva Cita	93
5.90. UC-44 Cancelar Cita	96
5.91. UC-45 Modificar Cita	96
5.92. UC-46 Filtrar Cita	97
5.93. UC-47 Listar Citas	98
5.94. UC-48 Cambiar Estado Cita	98
5.95. UC-49 Crear Nueva Factura	99
5.96. UC-50 Modificar Factura	102
5.97. UC-38 Filtrar Factura	103
5.98. UC-52 Listar Facturas	104
5.99. UC-53 Eliminar Factura	104
5.100UC-54 Generar Factura	107
5.101UC-55 Descargar Factura	107
5.102UC-56 Alta Vacuna	109
5.103UC-57 Modificar Vacuna	110
5.104UC-58 Listar Vacunas	111
5.105UC-59 Registrar Nueva Vacuna Mascota	111
5.106UC-60 Listar Vacunas de la Mascota	112
5.107UC-61 Registrar Nuevo Historial	114
5.108UC-62 Borrar Historial	116
5.109UC-63 Modificar Historial	116
5.110UC-64 Listar Historiales	118
5.111UC-65 Adjuntar Fichero al Historial	118

Índice de tablas

5.112UC-66 Listar Ficheros del Historial	120
5.113UC-67 Descargar Fichero del Historial	120
5.114UC-68 Consulta de Estadísticas	122
5.115Tabla de Entidades	135
5.116Entidad Cita	138
5.118Entidad Cliente	139
5.120Entidad Especie	140
5.122Entidad Factura	141
5.124Entidad Ficheros	142
5.126Entidad Historial	143
5.128Entidad IVA	144
5.130Entidad Mascota	145
5.133Entidad Pedido Línea	146
5.134Entidad Peso	147
5.136Entidad Población	148
5.138Entidad Producto	149
5.140Entidad Producto Familia	150
5.142Entidad Proveedor	151
5.144Entidad Provincia	152
5.146Entidad Raza	153
5.148Entidad Servicio	154
5.150Entidad Servicio Familia	155
5.152Entidad Usuario	156
5.154Entidad Vacuna	157
5.156Entidad Venta	158
5.158Entidad Venta Línea	159
5.160Tabla Cita Lógica	160
5.161Tabla Cliente Lógica	160
5.162Tabla Especie Lógica	161
5.163Tabla Factura Lógica	161
5.164Tabla Ficheros Lógica	161
5.165Tabla Historial Lógica	161
5.166Tabla IVA Lógica	162
5.167Tabla Mascota Lógica	162
5.168Tabla Mascota-Vacuna Lógica	162
5.169Tabla Pedido Lógica	162
5.170Tabla Pedido Línea Lógica	163
5.171Tabla Peso Lógica	163
5.172Tabla Población Lógica	163
5.173Tabla Producto Lógica	163

5.174	Tabla Producto-Familia Lógica	164
5.175	Tabla Proveedor Lógica	164
5.176	Tabla Provincia Lógica	164
5.177	Tabla Raza Lógica	164
5.178	Tabla Servicio Lógica	165
5.179	Tabla Servicio-Familia Lógica	165
5.180	Tabla Usuario Lógica	165
5.181	Tabla Vacuna Lógica	165
5.182	Tabla Venta Lógica	166
5.183	Tabla Venta-Línea Lógica	166

1. Introducción

En la actualidad, el Centro Veterinario «El Rey de la Casa», situado en la localidad sevillana de Montellano, realiza toda su gestión de forma manual. Por tanto, esto conlleva una ralentización en los procesos de gestión así como la aparición no deseada de información duplicada e incoherente.

Este PFC (Proyecto Fin de Carrera) surge a petición de dicha clínica veterinaria que desea implantar un sistema informático en su empresa que le permita automatizar toda la información de su empresa, así como la gestión de sus actividades de una forma más rigurosa y fiable. Por otro lado, el veterinario de la clínica necesita realizar continuas salidas para dar servicio a grandes animales o atender los servicios de urgencias y, por ello, sería deseable consultar y/o registrar el historial clínico del animal a tratar de forma remota. Además de esto, el sistema debe permitir un acceso multiusuario, ya que existen varios empleados en la clínica.

Así pues, en este PFC se desarrolla una Aplicación de Internet Enriquecida o *Rich Internet Application* (RIA) que permite a los veterinarios y sus empleados gestionar toda la información de sus clínicas veterinarias de una forma amigable e intuitiva, como si de una aplicación de escritorio se tratase, y con las características propias de las aplicaciones web, tales como el uso de un navegador web para la interacción con la aplicación sin necesidad de instalar *software* adicional en la máquina cliente y la disponibilidad de información a nivel mundial (acceso *online*) siempre y cuando se posea una conexión a Internet. Para ello, tras un estudio de las opciones disponibles, el desarrollo del PFC se ha realizado con el *framework* ZK [14], basado en JavaScript, el cual reúne todas las características deseadas por el cliente.

Los procesos de gestión que recoge este PFC son: la gestión de clientes, mascotas, la gestión de proveedores, productos, gestión de pedidos, ventas, citas, vacunas, gestión de historiales de las mascotas, facturas, y la gestión de usuarios.

Cabe mencionar que existe en el mercado aplicaciones para la gestión de centros veterinarios, pero o son demasiado complejas y caras, o no reúnen todos los requisitos deseados por nuestro cliente.

1.1. Objetivos

El objetivo principal de este PFC es el desarrollo de un *software* RIA para la gestión interna de la Clínica Veterinaria “El Rey de la Casa”.

1. Introducción

Para ello se llevará a cabo los siguientes sub-objetivos:

1. Gestionar los datos de los clientes, mascotas, productos, proveedores, pedidos, ventas y citas, así como administrar y generar de manera dinámica facturas, informes y listados.
2. Dotar de accesibilidad a la aplicación web las 24 horas del día, desde cualquier dispositivo a través de Internet y sin necesidad de instalación de ningún componente, basta con un navegador web.
3. Desarrollar una interfaz de usuario amigable e intuitiva junto con una buena documentación que ayude a utilizarla.
4. Gestionar y consultar estadísticas y gráficas generadas dinámicamente a partir de los datos almacenados en la base de datos.

1.2. Alcance

El PFC que se presenta está orientado a la utilización por parte de usuarios con conocimientos básicos en informática, con lo que se ha tenido especial cuidado en la usabilidad, que en ocasiones propuesta por el propio cliente, adaptándose a la forma de trabajar que realizan. Debido a la propia esencia del *framework* ZK, éste nos proporciona un entorno de desarrollo para la interfaz de una forma rápida, intuitiva y robusta, con la utilización de su propio lenguaje ZUML, similar a XML. El carácter de RIA le aporta una facilidad de uso extra para el usuario final, ya que no se tiene que preocupar por instalaciones, configuraciones, etc; basta con un simple navegador web. Además al ser *online*, el acceso es total, es decir, se puede acceder las 24 horas del día, independientemente del lugar desde el que se accede, basta con una conexión a Internet.

1.3. Visión general

En primer lugar, se analiza el estado del arte de los diferentes *frameworks* RIA existentes en el mercado en la actualidad. A continuación detallamos el calendario seguido para llevar a cabo este PFC con un diagrama de Gantt.

A lo largo del resto de la memoria se detalla las fases de análisis, diseño, implementación y pruebas que se ha seguido al realizar el proyecto. Los manuales de instalación y de usuario se incluyen tras un resumen de los aspectos más destacables del proyecto y las conclusiones.

1.4. Glosario

1.4.1. Acrónimos

AJAX Asynchronous JavaScript and XML

API Application Programming Interface

CE Community Edition

CIF Código de Identificación Fiscal

CPU Central Processing Unit

CSS Cascading Style Sheets

EE Enterprise Edition

EL Expression Language

FNBC Forma Normal Boyce-Codd

GPL GNU General Public License

GWT Google Web Toolkit

HTML HiperText Markup Language

IDE Integrated Development Environment

IVA Impuesto al Valor Agregado

JPEG Joint Photographic Experts Group

JRE Java Runtime Environment

LGPL GNU Lesser General Public License

MVC Model-View-Controller

MVVM Model-View-ViewModel

NIF Número de Identificación Fiscal

PDF Portable Document Format

PE Professional Edition

1. Introducción

PFC Proyecto Fin de Carrera

PNG Portable Network Graphics

POJO Plain Old Java Object

RAM Random Access Memory

RIA Rich Internet Application

SGBD Sistema de Gestión de Base de Datos

SMS Short Message Service

SVG Scalable Vector Graphics

UCA Universidad de Cádiz

URL Uniform Resource Locator

WAR Web application ARchive

XHTML eXtensible HyperText Markup Language

XML eXtensible Markup Language

ZUML ZK User Markup Language

1.4.2. Definiciones

Framework En español, marco de trabajo, es una estructura conceptual y tecnológica de soporte definido, con un conjunto de conceptos, prácticas y criterios estandarizados.

Open Source En el desarrollo del *software*, es una filosofía, una metodología pragmática que promueve la redistribución libre y acceso al diseño de un producto final y a sus detalles de implementación.

IDE Entorno de desarrollo integrado. Programa informático compuesto por un conjunto de herramientas y ayudas para la programación.

Add-on En español, complemento o añadido, es una aplicación que se relaciona con otra para aportarle una función nueva y generalmente muy específica. Esta aplicación adicional es ejecutada por la aplicación principal.

2. Estado del arte

En la actualidad existen distintos *frameworks* RIA: GWT [8], Silverlight [10], Flex [5] y ZK [14], entre otros. Para la elección del *framework* más apropiado para desarrollar este proyecto hemos tenido en cuenta distintas variables. En primer lugar, que fuera gratuito, ya que la empresa no puede asumir costes de licencias. Además, tiene que ser un *framework* maduro, suficientemente probado y que disponga de una gran comunidad *software* que haga posible su duración a lo largo del tiempo, proporcionando una buena documentación. Finalmente, debe disponer de utilidades potentes para la creación de formularios, tablas de datos y gráficos, generación de informes y listados para imprimir, debe permitir crear interfaces que sean usables y amigables, y debe proporcionar persistencia de datos así como desarrollo con programación orientada a objetos.

2.1. GWT

GWT (Google Web Toolkit) es un conjunto de herramientas de desarrollo para la construcción de aplicaciones complejas optimizadas para navegadores. Su objetivo es permitir el desarrollo de aplicaciones web de alto rendimiento sin que el desarrollador tenga que conocer las características singulares de cada navegador. Su interfaz de usuario es bastante sencilla y aburrida, si queremos mejorarla debemos utilizar widgets adicionales como GWT-EXT o SmartGWT.

GWT consta de cuatro componentes:

GWT Java-to-JavaScript Compiler Este componente es el encargado de traducir el código desarrollado en Java [20] a lenguaje JavaScript.

Hosted Web Browser Este componente nos permite ejecutar la aplicación Java sin traducirla a JavaScript, en modo host usando la máquina virtual Java.

JRE Emulation Library Contiene las bibliotecas más importantes de las clases de Java.

GWT Web UI Class Library Este componente contiene un conjunto de elementos para la interfaz de usuario.

2. Estado del arte

2.2. Flex

Flex [5] es un *framework Open Source* de Adobe basado en Flash que nos permite crear programas RIA. Además podemos crear aplicaciones para iOS, Android™ y dispositivos con Blackberry® TabletOS. Para el desarrollo de las interfaces de usuario, tenemos que hacer uso de un lenguaje XML propio de Flex, como es el MXML y para la programación lógica utiliza ActionScript3, aunque para el desarrollo de la lógica dura, como puede ser el acceso a la base de datos, generación de PDFs, se puede realizar en Java™, PHP o .NET.

Al final se genera un fichero SWF (FlashPlayer), aunque también se puede generar para una aplicación de escritorio con Adobe Air.

El entorno de desarrollo más utilizado para este RIA es FlexBuilder, el cual es de pago; aunque también existe un plugin para Eclipse.

2.3. Silverlight

Microsoft Silverlight [10] es una herramienta para la creación de RIA, desarrollada por Microsoft Corporation. Como ocurre con Flex, el desarrollo de la interfaz se realiza con un lenguaje XML propio, como es XAML y el desarrollo de la parte lógica se realiza con el lenguaje .NET. El entorno de desarrollo es Visual Studio.

Al final del desarrollo, para usar la aplicación se necesita instalar un componente adicional en el navegador.

2.4. ZK

Finalmente tenemos ZK, el *framework* elegido para el desarrollo de nuestro programa RIA, propiedad de la compañía Potix Corporation. ZK nace con la premisa de potenciar la productividad de los desarrolladores, en la que se realiza una programación orientada a objetos gracias al uso del lenguaje Java y traduciendo este código Java a JavaScript, dando como resultado una aplicación AJAX (Asynchronous JavaScript and XML) sin la necesidad de conocer JavaScript. El desarrollo de la interfaz de usuario se realiza con el lenguaje ZUML (ZK User Markup Language), basado en el lenguaje XUL con alrededor de 123 componentes y otros 83 componentes XHTML (eXtensible HyperText Markup Language), estos componentes son una representación POJO (Plain Old Java Object) de todos los componentes XHTML y una conjunto adicional de todos los componentes del propio ZK, que con su interfaz cuidada, amigable, independiente del sistema operativo y totalmente personalizable con el uso de CSS, hace que sea una de sus bazas fuertes. Para el desarrollo de la capa lógica hacemos uso de Java™, aunque también soporta desarrollar en Groovy, Rhino, Ruby y Python. Podemos embeber código Java dentro de ZUML,

así como manipular datos dentro de un archivo ZUML con el uso de *EL* (Expression Language).

Aunque para el desarrollo de este PFC hemos utilizado la versión LGPL (GNU Lesser General Public License), ZK CE; existen otras versiones de pago: ZK PE y ZK EE, con componentes y extensiones adicionales, como es capacidad de integrar ZK Spring y ZK JSP Tags, o bien componentes muy útiles como añadir un calendario con ZK Calendar.

Como entornos de desarrollo disponibles tenemos como referencia y recomendado Eclipse instalando el *add-on* ZK Studio, también podemos usar NetBeans utilizando un *add-on* REM [12], que es el que hemos utilizado en este PFC .

La página web de ZK es la siguiente: <http://www.zkoss.org>

2. Estado del arte

3. Desarrollo del calendario

En esta sección se describirá la planificación de las fases de desarrollo y para una mejor visualización se adjunta un diagrama de Gantt. (Véase Figura 3.1).

3.1. Fases

3.1.1. Elicitación de requisitos

Antes de reunirnos con el cliente, recibimos un documento con los requisitos y funcionalidades deseadas. Una vez estudiado y analizado el documento, nos reunimos con el cliente para que nos explicara de primera mano qué era lo que necesitaba y así resolver las dudas que pudieramos tener, en esta reunión asistió el veterinario y la gestora de la clínica, así nos pudieron dar la información de todas los requisitos que debiera llevar la aplicación. (Duración 16 días).

3.1.2. Análisis

Tras determinar los requisitos de la fase anterior, procedemos al análisis de estos. Extrayendo los casos de uso, definiendo actores implicados, desarrollo del modelo conceptual de datos y diagramas de secuencia del sistema. (Duración 21 días).

3.1.3. Diseño

Durante la fase de diseño, se estudió como se iba a estructurar el sistema, además se tuvo en cuenta la reutilización de componentes. (Duración 30 días).

3.1.4. Implementación

En esta fase se realizó la codificación del sistema para que realizara el comportamiento deseado, además de la creación de la base de datos y las tablas necesarias. (Duración 132 días).

3. Desarrollo del calendario

3.1.5. Verificación

La fase de verificación se ha ido realizando desde la fase de implementación, probando y verificando que el resultado era el deseado. Una vez finalizada la fase de implementación se entregó la aplicación a personas ajenas al desarrollo para que la probaran ya que el programador siempre tiende a realizar pruebas con resultado satisfactorio. (Duración 175 días).

3.1.6. Documentación

La documentación se ha ido redactando según se iba avanzando en cada fase del desarrollo, aunque cabe indicar que cuando se finalizó las etapas anteriores se aumentó el tiempo dedicado a su redacción. (Duración 247 días).

3.2. Diagrama de Gantt

A continuación, se puede observar en la Figura 3.1 el diagrama de Gantt, como resultado de la planificación de las fases de desarrollo del *software*.

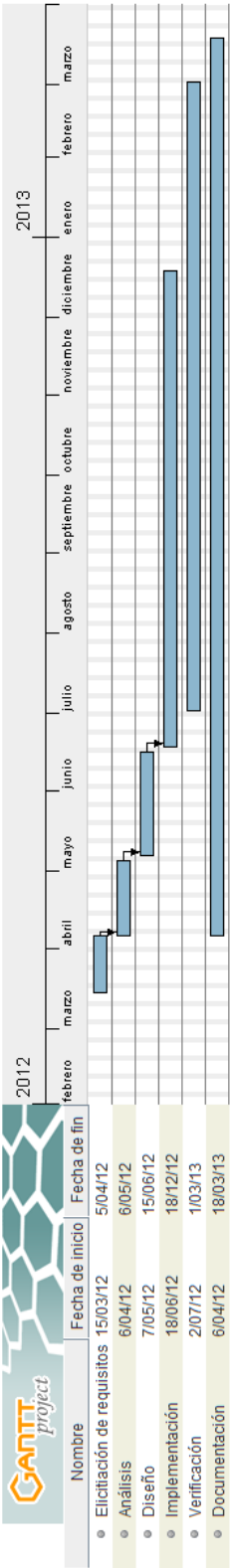


Figura 3.1.: Diagrama de Gantt

3. Desarrollo del calendario

4. Descripción general del proyecto

En este capítulo, especificaremos los aspectos que debe reunir el desarrollo de este PFC.

4.1. Perspectiva del producto

El producto final es una programa RIA. La aplicación es totalmente independiente y dispone de todas las herramientas necesarias para la gestión de la clínica veterinaria «El Rey de la Casa», al ser estos procesos de gestión generalizados, son por tanto extensibles a cualquier clínica veterinaria con características similares.

4.1.1. Entorno del producto

El *software* de Gestión Informática de Clínicas Veterinarias es un *software* RIA, que como propio nombre indica, está orientada a su uso online, y cómo único requisito se establece la necesidad de disponer de un navegador web moderno. Se recomienda las últimas versiones de los navegadores Google Chrome y Mozilla Firefox.

4.1.2. Interfaz de usuario

Una de las características a destacar por parte del *framework* ZK, es la interfaz amigable de la que dispone. Hace que el desarrollador se pueda centrar en las cosas importantes durante el proceso de desarrollo del *software*. Además se ha tenido en cuenta las sugerencias aportadas por el cliente en las distintas entrevistas que se ha tenido con ellos durante el desarrollo del mismo, así que los procesos de gestión son conocidos y aprobados por ellos.

4.2. Funciones

A continuación, se describen las funcionalidades de las que dispone nuestra aplicación.

4. Descripción general del proyecto

- Dotar de accesibilidad a la aplicación web las 24 horas del día, desde cualquier dispositivo a través de Internet y sin necesidad de instalación de ningún componente, basta con un navegador web.
- Gestionar los datos de los clientes y las mascotas que éstos posean.
- Gestionar el historial de las mascotas, dando la posibilidad de adjuntar ficheros, ya sean fotografías o cualquier otro tipo de documento, como PDF, Word, Excel; de interés para el historial de la mascota.
- Dotar del control de pesos y las vacunas que se administran a los animales, así como generar automáticamente recordatorios.
- Gestionar los productos y su stock, gestionar proveedores, pedidos, ventas realizadas y citas solicitadas.
- Gestionar las facturas y dotar la capacidad de generación de documentos en PDF para su almacenamiento o impresión posterior. Así como ocurre con las facturas, también se puede generar documentos PDF dinámicamente de los albaranes.
- Desarrollar una interfaz de usuario amigable e intuitiva junto con una buena documentación que ayude a utilizarla.
- Gestionar y consultar estadísticas y gráficas generadas dinámicamente a partir de los datos almacenados en la base de datos.

4.3. Características del usuario

Nuestra aplicación de gestión va dirigida a los empleados de la clínica veterinaria. Los conocimientos informáticos que se requiere por parte del empleado, son conocimientos básicos, el hecho de disponer de una interfaz amigable e intuitiva hace que el usuario se sienta cómodo usándola. Además se aporta un manual de usuario, para que el usuario si tuviera alguna duda, pueda solventar sin dificultad.

4.4. Restricciones generales

4.4.1. Control de versiones

Puesto que el desarrollo de la aplicación se ha realizado de forma incremental, ha sido necesario la utilización de un sistema de control de versiones. Con un sistema

de control de versiones, nos aseguramos de la pérdida de información accidental y así como codificaciones erróneas, pudiéndose revertir a un estado estable del mismo.

En este caso, hemos optado por utilizar Mercurial [19], de fácil configuración e integración con el IDE NetBeans, multiplataforma y con licencia GPL V2.

Todas estas razones hacen que nos hayamos decantado por su utilización y no otros SVNs. Existe a disposición de los usuarios una completa documentación, con un libro gratuito en <http://hgbook.red-bean.com/>.

4.4.2. Lenguajes de programación y tecnologías

Para llevar a cabo la realización del PFC, se ha utilizado distintos lenguajes de programación y varias tecnologías:

- ZK *framework* [14], es la tecnología principal utilizada.
- Java[™] [20] es el lenguaje de programación que utiliza ZK.
- ZUML, como lenguaje para la creación de la capa Vista de nuestra aplicación.
- MySQL [4] como sistema de gestión de base de datos.
- Apache Tomcat 7.0 [17].
- Servidor HTTP Apache.
- Generación de gráficas con ZHighCharts [13] como componente de ZK, diseñado para incluir la librería de Highcharts JS dentro de la aplicación.
- iText[®] [3] es la tecnología elegida para la generación de documentos PDF.
- UML [16] para el diseño de los distintos diagramas.

4.4.3. Herramientas

En esta sección se comenta las herramientas utilizadas para el desarrollo del proyecto:

- NetBeans [11]: es un entorno de desarrollo integrado (IDE) libre y de código abierto. Dispone de multitud de herramientas y utilidades que ayudan al desarrollador a realizar su propósito.
- REM [12]: es un módulo de NetBeans para el desarrollo de aplicaciones RIA con el *framework* ZK.

4. Descripción general del proyecto

- Dia [6]: *software* para el diseño de diagramas de distintos tipos, en este proyecto es utilizado para el diseño de los diagramas UML.
- Navegador Chrome: es un navegador web desarrollado por Google con el motor de renderizado WebKit.
- GIMP (GNU Image Manipulation Program) [7]: es un programa de manipulación de imágenes, utilizado en este PFC para la edición de iconos.
- L^AT_EX [9]: procesador de documentos que combina la potencia y flexibilidad de T_EX/L^AT_EX.
- Gantt Project: *software open source* para la creación de diagramas de Gantt.
- phpMyAdmin: herramienta para la gestión de la base de datos de una forma gráfica.

4.4.4. Sistemas operativos y hardware

A continuación se describirán las características en materia de sistema operativo y hardware utilizado a la hora de realizar el presente PFC. (véase la Tabla 4.4.4).

Aunque el desarrollo se haya realizado con las características detallada en la tabla 4.1, nuestra aplicación no requiere de tales características, bastaría con 1GB de RAM y una CPU de doble núcleo. En lo que se refiere al Sistema Operativo, no se requiere el utilizado durante el desarrollo, el único requisito es que se pueda instalar JRE 1.6, el servidor web Apache Tomcat 7.0, el servidor HTTP Apache necesaria para la herramienta phpMyAdmin, y MySQL como sistema de gestión de base de datos.

Una vez tengamos instalado este *software*, con un navegador web moderno podremos acceder a nuestra aplicación de gestión de clínicas veterinarias.

Tabla 4.1.: Sistema Operativo y Hardware

Sistema Operativo	Microsoft Windows 7 Home Premium 64 bits
Modelo	Dell XPS 15 L502X
Monitor externo	LG M227WDP
Procesador	Intel Core i7 2630QM 2.0GHz
Memoria RAM	6GB DDR3-1333
Tarjeta Gráfica	GeForce GT540M 2GB GDDR4

5. Desarrollo del proyecto

En este capítulo procederemos a la descripción de las distintas etapas que se llevan a cabo durante el desarrollo del *software*.

5.1. Modelo de ciclo de vida

El ciclo de vida es el conjunto de etapas por las que pasa el sistema que se está desarrollando desde que nace la idea inicial hasta que el *software* es retirado o reemplazado por otro. Un modelo de ciclo de vida de *software* es una vista de las actividades que ocurren durante el desarrollo de *software*.

Para el presente PFC se ha utilizado un modelo de desarrollo secuencial adaptado a la ingeniería del *software* orientada a objetos, que fomenta la reutilización de código. Los sistemas orientados a objetos son más fáciles de adaptar y escalar.

5.2. Requisitos

En esta sección se describirán los distintos requisitos: requisitos funcionales, requisitos de información, requisitos de reglas de negocio, requisitos de interfaz y requisitos no funcionales.

5.2.1. Funcionales

Tabla 5.1.: FRQ-001 Gestión de clientes

Descripción	El sistema deberá gestionar los datos de los clientes, dotando la posibilidad de dar de alta, bajas, modificaciones y la consulta de sus datos.
Frecuencia	Alta
Comentarios	

5. Desarrollo del proyecto

Tabla 5.2.: FRQ-002 Gestión de mascotas

Descripción	El sistema deberá gestionar los datos de las mascotas, dotando la posibilidad de dar de alta, bajas, modificaciones y la consulta de sus datos.
Estabilidad	Alta
Comentarios	

Tabla 5.3.: FRQ-003 Gestión de proveedores

Descripción	El sistema deberá gestionar los datos de los proveedores, dotando la posibilidad de dar de alta, bajas, modificaciones y la consulta de sus datos.
Frecuencia	Alta
Comentarios	

Tabla 5.4.: FRQ-004 Gestión de productos

Descripción	El sistema deberá gestionar los datos de los productos, dotando la posibilidad de dar de alta, bajas, modificaciones y la consulta de sus datos.
Frecuencia	Alta
Comentarios	

Tabla 5.5.: FRQ-005 Gestión de ventas

Descripción	El sistema deberá gestionar los datos de los ventas, dotando la posibilidad de añadir nueva venta, eliminaciones, modificaciones y consultas.
Frecuencia	Alta
Comentarios	

Tabla 5.6.: FRQ-006 Gestión de historiales

5.2. Requisitos

Descripción	El sistema deberá gestionar los datos de los historiales, dotando la posibilidad de añadir nuevo historial y consultas de sus datos.
Frecuencia	Alta
Comentarios	

Tabla 5.7.: FRQ-007 Gestión de citas

Descripción	El sistema deberá gestionar la información sobre las citas, dotando la posibilidad de crear nuevas citas, modificaciones y la consulta de sus datos.
Frecuencia	Alta
Comentarios	

Tabla 5.8.: FRQ-008 Gestión de facturas

Descripción	El sistema deberá gestionar las facturas, dotando la posibilidad de crear nuevas facturas, modificaciones y su consulta.
Frecuencia	Alta
Comentarios	

Tabla 5.9.: FRQ-009 Gestión de vacunas

Descripción	El sistema deberá gestionar la información sobre las vacunas, dotando la posibilidad de asignar vacunas y la consulta de estos.
Frecuencia	Alta
Comentarios	

5. Desarrollo del proyecto

Tabla 5.10.: FRQ-010 Gestión de usuarios

Descripción	El sistema deberá gestionar los datos de los usuarios, dotando la posibilidad de dar de alta, bajas, modificaciones y la consulta de sus datos.
Frecuencia	Alta
Comentarios	

Tabla 5.11.: FRQ-011 Creación de albaranes

Descripción	El sistema deberá crear nuevos albaranes, dotando la posibilidad de generar documentos PDF para su almacenaje y posterior descarga.
Frecuencia	Alta
Comentarios	

Tabla 5.12.: FRQ-012 Modificación contraseña de acceso

Descripción	El sistema deberá permitir la posibilidad al usuario de modificar su propia contraseña.
Frecuencia	Baja
Comentarios	

Tabla 5.13.: FRQ-013 Creación automática de recordatorios

Descripción	El sistema deberá crear automáticamente nuevos recordatorios en el sistema cuando se registre una vacuna a una mascota.
Frecuencia	Baja
Comentarios	

Tabla 5.14.: FRQ-014 Creación automática de código de producto

Descripción	El sistema deberá crear automáticamente el código del producto cuando se de da alta uno nuevo.
Frecuencia	Baja

Comentarios	
-------------	--

Tabla 5.15.: FRQ-015 Creación automática de código servicio

Descripción	El sistema deberá crear automáticamente el código del servicio cuando se de da alta uno nuevo.
Frecuencia	Alta
Comentarios	

5.2.2. De información

En esta sección describiremos las distintas entidades de nuestro sistema, relaciones y atributos.

Tabla 5.16.: IRQ-01 Cita

Requisitos asociados	
Descripción	El sistema deberá almacenar la información de las citas asignadas a un cliente.
Datos específicos	<ul style="list-style-type: none"> ■ Fecha y hora. ■ Estado de la cita. ■ Tipo de cita. ■ Mascota a la que pertenece la cita. ■ Cliente al que pertenece la cita. ■ Empleado que creó la cita.
Frecuencia	Media
Comentarios	

Tabla 5.17.: IRQ-02 Cliente

Requisitos asociados	
Descripción	El sistema deberá almacenar los datos de los clientes.

5. Desarrollo del proyecto

Datos específicos	<ul style="list-style-type: none">■ NIF/CIF■ Nombre y apellidos.■ Dirección, Ciudad y Provincia.■ Teléfono y teléfono2.■ Email.■ Fecha de alta.
Frecuencia	Media
Comentarios	

Tabla 5.18.: IRQ-03 Especie

Requisitos asociados	IRQ-Raza
Descripción	El sistema deberá almacenar las especies.
Datos específicos	<ul style="list-style-type: none"> ■ Nombre de la especie
Frecuencia	Media
Comentarios	

Tabla 5.19.: IRQ-03 Tabla especie

Tabla 5.20.: IRQ-04 Factura

Requisitos asociados	
Descripción	El sistema deberá almacenar las facturas creadas a partir de las ventas a los clientes.
Datos específicos	<ul style="list-style-type: none"> ■ Número. ■ Cliente al que pertenece. ■ Empleado que creó la factura. ■ Fecha. ■ Ruta donde se encuentra almacenada.
Frecuencia	Media
Comentarios	

5. Desarrollo del proyecto

Tabla 5.21.: IRQ-05 Ficheros

Requisitos asociados	
Descripción	El sistema deberá almacenar la información referente a los ficheros y documentos.
Datos específicos	<ul style="list-style-type: none">■ Historial al que pertenece■ Ruta donde se encuentra almacenada.■ Fecha de subida.
Frecuencia	Media
Comentarios	

Tabla 5.22.: IRQ-06 Historial

Requisitos asociados	
Descripción	El sistema deberá almacenar la información referente a los Historiales asociados a una Mascota.
Datos específicos	<ul style="list-style-type: none">■ Mascota a la que pertenece.■ Veterinario que creó el historial.■ Fecha de creación.■ Tipo de visita.■ Peso de la mascota.■ Anamnesis, diagnóstico y tratamiento
Frecuencia	Alta
Comentarios	

Tabla 5.23.: IRQ-07 IVA

Requisitos asociados	
Descripción	El sistema deberá almacenar los tipos de IVA existentes.
Datos específicos	<ul style="list-style-type: none"> ■ Valor. ■ Nombre y descripción breve.
Frecuencia	Baja
Comentarios	

Tabla 5.24.: IRQ-08 Mascota

Requisitos asociados	
Descripción	El sistema deberá almacenar los datos de las Mascotas.
Datos específicos	<ul style="list-style-type: none"> ■ Número de chip. ■ Nombre. ■ Sexo. ■ Fecha de Nacimiento y Defunción. ■ Altura ■ Tipo de Pelo (Corto, Duro, Largo, Medio, Medio-largo, Propio, Rizado) ■ Raza a la que pertenece. ■ Fecha de alta en el sistema. ■ Fecha de baja en el sistema. ■ Cliente al que pertenece.
Frecuencia	Alta
Comentarios	

5. Desarrollo del proyecto

Tabla 5.25.: IRQ-10 Mascota-Vacuna

Requisitos asociados	
Descripción	El sistema deberá almacenar las relaciones de las Mascotas con las Vacunas que se le ha administrado.
Datos específicos	<ul style="list-style-type: none">■ Mascota al que pertenece.■ Vacuna que se le ha administrado.■ Fecha.■ Veterinario que ha administrado la vacuna.
Frecuencia	Alta
Comentarios	

Tabla 5.26.: IRQ-11 Pedido

Requisitos asociados	
Descripción	El sistema deberá almacenar los datos de los Pedidos que se realizan.
Datos específicos	<ul style="list-style-type: none">■ Fecha del pedido, fecha de entrega estimada, fecha del pago.■ Estado del pago.■ Proveedor al que se ha realizado el Pedido.■ Empleado que ha hecho el Pedido.
Frecuencia	Media
Comentarios	

Tabla 5.27.: IRQ-12 Pedido Línea

Requisitos asociados	
Descripción	El sistema deberá almacenar cada una de las líneas de las que se compone un Pedido.
Datos específicos	<ul style="list-style-type: none"> ■ Pedido al que pertenece. ■ Producto que se ha solicitado. ■ Cantidad de producto solicitado. ■ Precio del Producto.
Frecuencia	Media
Comentarios	

Tabla 5.28.: IRQ-13 Peso

Requisitos asociados	
Descripción	El sistema deberá almacenar un historial de pesos de la mascota.
Datos específicos	<ul style="list-style-type: none"> ■ Mascota a la que pertenece. ■ Valor. ■ Fecha de medición.
Frecuencia	Media
Comentarios	

5. Desarrollo del proyecto

Tabla 5.29.: IRQ-14 Población

Requisitos asociados	
Descripción	El sistema deberá almacenar las poblaciones del estado español.
Datos específicos	<ul style="list-style-type: none">■ Nombre.■ Provincia a la que pertenece.
Frecuencia	Baja
Comentarios	

Tabla 5.30.: IRQ-15 Producto

Requisitos asociados	
Descripción	El sistema deberá almacenar la información correspondiente a los productos.
Datos específicos	<ul style="list-style-type: none">■ Código.■ Nombre.■ Familia a la que pertenece el producto.■ Proveedor que ha suministrado el producto.■ Precio de venta al público.■ Precio de compra.■ IVA que lleva asociado.■ Stock del que se dispone.■ Ruta de una imagen del producto.■ Fecha de alta en el sistema.
Frecuencia	Media
Comentarios	

Tabla 5.31.: IRQ-16 Producto Familia

Requisitos asociados	
Descripción	El sistema deberá almacenar la información correspondiente a la Familia de Productos que se vendan en la clínica.
Datos específicos	<ul style="list-style-type: none"> ■ Nombre. ■ Tipo tratamiento.
Frecuencia	Media
Comentarios	

Tabla 5.32.: IRQ-17 Proveedor

Requisitos asociados	
Descripción	El sistema deberá almacenar la información correspondiente a los proveedores.
Datos específicos	<ul style="list-style-type: none"> ■ NIF. ■ Nombre. ■ Direcció, población y provincia en la que se encuentra. ■ Teléfono y teléfono2. ■ Email. ■ Fecha de alta en el sistema. ■ Fecha de baja en el sistema. ■ Activo. ■ Nombre y apellidos de la persona de contacto.
Frecuencia	Media

5. *Desarrollo del proyecto*

Comentarios	
-------------	--

Tabla 5.33.: IRQ-18 Provincia

Requisitos asociados	
Descripción	El sistema deberá almacenar las provincias del Estado español.
Datos específicos	<ul style="list-style-type: none"> ■ Nombre de la provincia
Frecuencia	Media
Comentarios	

Tabla 5.34.: IRQ-19 Raza

Requisitos asociados	
Descripción	El sistema deberá almacenar las razas de las mascotas.
Datos específicos	<ul style="list-style-type: none"> ■ Nombre de la raza. ■ Especie a la que pertenece.
Frecuencia	Media
Comentarios	

5. Desarrollo del proyecto

Tabla 5.35.: IRQ-20 Servicio

Requisitos asociados	
Descripción	El sistema deberá almacenar la información correspondiente a los servicios ofertados por la clínica veterinaria.
Datos específicos	<ul style="list-style-type: none">■ Código.■ Nombre.■ Breve descripción del servicio.■ Precio.■ Tipo de IVA asociado.■ Familia a la que pertenece el Servicio.
Frecuencia	Media
Comentarios	

Tabla 5.36.: IRQ-21 Servicio Familia

Requisitos asociados	
Descripción	El sistema deberá almacenar la información correspondiente a los servicios ofertados por la clínica veterinaria.
Datos específicos	<ul style="list-style-type: none">■ Código.■ Nombre.■ Breve descripción del servicio.■ Precio.■ Tipo de IVA asociado.■ Familia a la que pertenece el Servicio.

Frecuencia	Media
Comentarios	

Tabla 5.37.: IRQ-21 Usuario

Requisitos asociados	
Descripción	El sistema deberá almacenar la información correspondiente a los usuarios.
Datos específicos	<ul style="list-style-type: none"> ■ Nombre de usuario. ■ Nombre y apellidos. ■ Contraseña. ■ Rol. ■ NIF. ■ Domicilio. ■ Teléfono de contacto. ■ Email. ■ Número de la Seguridad Social. ■ Fecha de alta en el sistema.
Frecuencia	Media
Comentarios	

Tabla 5.38.: IRQ-22 Vacuna

Requisitos asociados	
Descripción	El sistema deberá almacenar los datos de las vacunas.

5. Desarrollo del proyecto

Datos específicos	<ul style="list-style-type: none">■ Nombre.■ Breve descripción de la vacuna.■ Especie a la que va dirigida la vacuna.■ Número de días hasta el recordatorio de la vacuna.■ Fecha de registro en el sistema.
Frecuencia	Alta
Comentarios	

Tabla 5.39.: IRQ-23 Venta

Requisitos asociados	
Descripción	El sistema deberá almacenar los datos de las ventas realizadas.
Datos específicos	<ul style="list-style-type: none"> ■ Fecha de la venta. ■ Cliente asociado a la Venta. ■ Empleado que realiza la Venta. ■ Veterinario asociado a la Venta. ■ Albarán. ■ Factura a la que corresponde le Venta. ■ Estado Facturado.
Frecuencia	Alta
Comentarios	

Tabla 5.40.: IRQ-24 Venta Línea

Requisitos asociados	
Descripción	El sistema deberá almacenar los datos de cada una de las líneas de las que se compone una venta.

5. Desarrollo del proyecto

Datos específicos	<ul style="list-style-type: none">■ Venta a la que corresponde la línea.■ Item asociado a la línea de venta.■ Tipo de item: Producto o Servicio.■ Cantidad.■ Fecha de venta.■ Precio de venta.■ IVA asociado.
Frecuencia	Media
Comentarios	

5.2.3. Reglas de negocio

Tabla 5.41.: RNRQ-01 Contraseñas seguras y cifradas

Requisitos asociados	
Descripción	El sistema deberá almacenar las contraseñas de los usuarios del sistema, deben permanecer ocultas en todo momento.
Datos específicos	Las contraseñas deben contener al menos 8 caracteres, además de contener letras mayúsculas, minúsculas y números.
Frecuencia	Alta
Comentarios	

5.2.4. Interfaz

Tabla 5.42.: IFRQ-01 Consistencia en la apariencia

Requisitos asociados	
Descripción	El sistema deberá mostrar la aplicación con independencia del sistema con el que se acceda.
Datos específicos	
Frecuencia	Alta
Comentarios	

Tabla 5.43.: IFRQ-02 Interfaz amigable e intuitiva

Requisitos asociados	
Descripción	El sistema deberá mostrar la aplicación con una interfaz amigable e intuitiva, destacando los procesos sugeridos por el usuario.
Datos específicos	
Frecuencia	Alta
Comentarios	

5. Desarrollo del proyecto

5.2.5. No funcionales

Tabla 5.44.: NFRQ-01 Fiabilidad

Requisitos asociados	
Descripción	El sistema deberá calcular correctamente el cálculo de las ventas, facturas y albaranes.
Datos específicos	
Frecuencia	Alta
Comentarios	

Tabla 5.45.: NFRQ-02 Multiplataforma

Requisitos asociados	
Descripción	El sistema deberá poder instalarse en cualquier sistema operativo en el que se pueda instalar el <i>software</i> necesario.
Datos específicos	
Frecuencia	Alta
Comentarios	

Tabla 5.46.: NFRQ-03 Rendimiento

Requisitos asociados	
Descripción	El sistema deberá disponer de una Venta Rápida. Carga asíncrona de listados.
Datos específicos	El menú existe un apartado para realizar una ventá rápida.
Frecuencia	Alta
Comentarios	Sugerencia aportada por el usuario del sistema. Una Venta Rápida es un acceso directo en el menú principal, en el que se crea una venta para los casos en los que el cliente no está dado de alta en el sistema o bien no es relevante guardar esta información sobre un cliente.

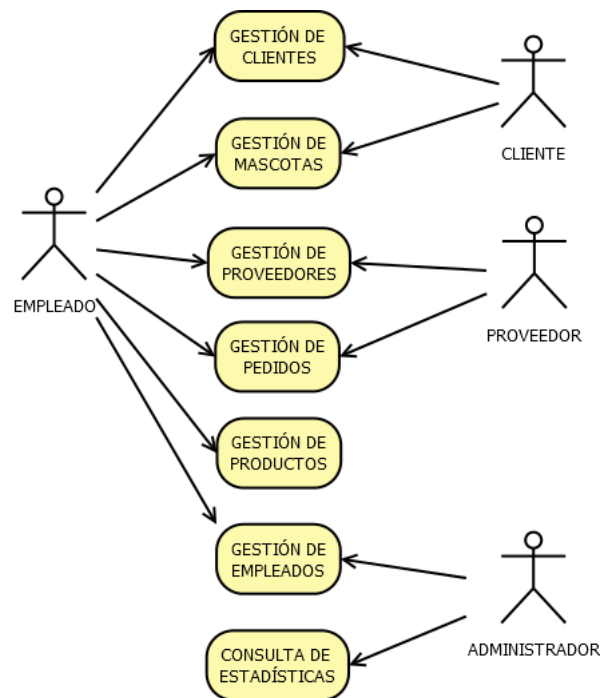


Figura 5.1.: Diagrama de Casos de Uso 1

5.3. Análisis del sistema

En sección procederemos a especificar el análisis del sistema, detallando cada uno de los Casos de Uso.

5.3.1. Casos de uso

5.3.1.1 Subsistema de Gestión de Clientes

A continuación se describirán los detalles referente al subsistema Gestión de Clientes:

Tabla 5.47.: UC-01 Alta Cliente

Actores	
Requisitos asociados	
Descripción	Permite a un usuario dar de alta a un nuevo cliente en el sistema.

5. Desarrollo del proyecto

Precondiciones	El usuario tiene que estar logueado en el sistema.
Escenario principal	<ol style="list-style-type: none"> 1. El cliente acude a la clínica y aporta sus datos al usuario. 2. El usuario introduce los datos del cliente en el sistema: NIF/CIF, Nombre, Apellidos, Dirección, Población, Provincia, Teléfono, Teléfono2, Email. 3. El Sistema comprueba que no hay espacios en blanco en los campos obligatorios: NIF/CIF, Nombre, Apellidos, Dirección, Población, Provincia y Teléfono, y que no existe ningún Cliente con el mismo NIF/CIF. 4. El Sistema lo registra.
Postcondiciones	Se registra al cliente en el sistema con los datos aportados.
Escenario alternativo	<ol style="list-style-type: none"> 3a. El Sistema detecta que existen espacios en blanco en algún campo obligatorio. El Sistema muestra el error. 3b. El NIF/CIF introducido ya se encuentra registrado en el sistema. El Sistema muestra el error. <p>*. En cualquier momento el usuario puede cancelar el alta, pulsando en el boton «Cerrar» en la esquina superior derecha.</p>
Frecuencia	Media
Estabilidad	Alta
Comentarios	

Tabla 5.48.: UC-02 Baja Cliente

Actores	
Requisitos asociados	
Descripción	Permite a un usuario dar de baja a un cliente registrado en el sistema.
Precondiciones	El usuario tiene que estar logueado en el sistema.

Escenario principal	<ol style="list-style-type: none"> 1. El cliente desea darse de baja en el sistema. 2. El usuario hace un filtrado con los datos del cliente en el Sistema. 3. El Sistema muestra un listado con los datos de filtrado introducidos y el usuario selecciona el cliente. 4. El usuario selecciona en «Eliminar» y se abre un mensaje emergente preguntando la confirmación de eliminación. 5. El usuario confirma la eliminación y se elimina del Sistema.
Postcondiciones	El cliente es dado de baja en el Sistema.
Escenario alternativo	<ol style="list-style-type: none"> 3a. En el listado mostrado no se encuentra el cliente, se muestra un listado en blanco. Se vuelve al paso 2). 4a. El usuario no confirma la eliminación del cliente. No se realiza ningún cambio en el Sistema.
Frecuencia	Baja
Estabilidad	Alta
Comentarios	

Tabla 5.49.: UC-03 Filtrar Cliente

Actores	
Requisitos asociados	
Descripción	Permite a un usuario hacer un filtrado con los datos introducidos.
Precondiciones	El usuario tiene que estar logueado en el sistema.

5. Desarrollo del proyecto

Escenario principal	<ol style="list-style-type: none">1. El usuario introduce los datos en los campos que desea filtrar.2. El Sistema muestra un listado de los clientes que contienen los datos introducidos.
Postcondiciones	Se muestra un listado de los clientes.
Escenario alternativo	2a. Los datos introducidos no coinciden con ningún cliente y no se muestra ningún Cliente.
Frecuencia	Media
Estabilidad	Alta
Comentarios	

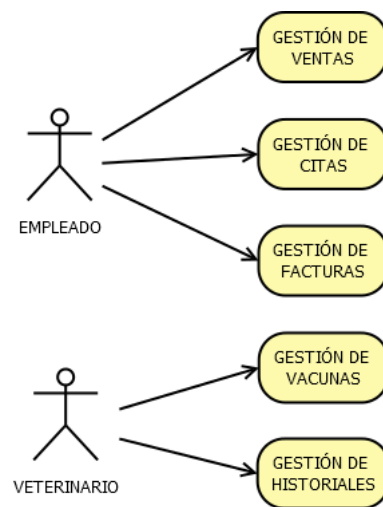


Figura 5.2.: Diagrama de Casos de Uso 2

5. Desarrollo del proyecto

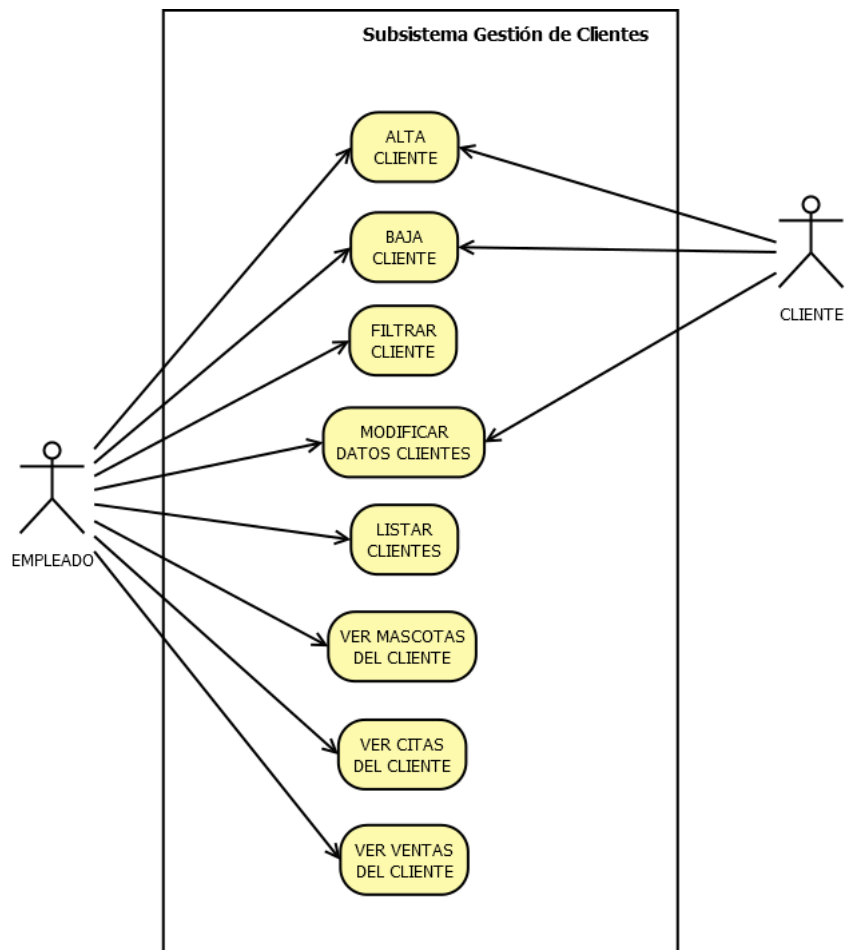


Figura 5.3.: Diagrama Gestión de Clientes

Tabla 5.50.: UC-04 Modificar Cliente

Actores	
Requisitos asociados	
Descripción	Permite a un usuario modificar los datos sobre un cliente.
Precondiciones	El usuario tiene que estar logueado en el sistema.
Escenario principal	<ol style="list-style-type: none"> 1. El usuario selecciona a un cliente específico de un listado que muestra el Sistema y hace doble click sobre él. 2. El Sistema muestra el perfil del Cliente. 3. El usuario pulsa en el botón «Modificar». 4. El Sistema habilita los campos para que se pueda modificar. 5. El usuario modifica los campos los campos necesarios. 6. El Sistema comprueba que los campos obligatorios no están vacíos. 7. El usuario pulsa en el botón «Guardar». 8. El Sistema deshabilita los campos y el botón «Guardar». A continuación guarda los datos modificados en el sistema.
Postcondiciones	Se modifica en el Sistema la información del Cliente con los datos especificados.
Escenario alternativo	<ol style="list-style-type: none"> 6a. El Sistema detecta que hay algún campo obligatorio vacío y muestra el Error. 8a. El Sistema detecta que existen campos vacíos en aquellos que son obligatorios, no se realiza la modificación.
Frecuencia	Baja
Estabilidad	Alta
Comentarios	

5. Desarrollo del proyecto

Tabla 5.51.: UC-05 Listar Clientes

Actores	
Requisitos asociados	
Descripción	Permite a un usuario examinar la lista de clientes.
Precondiciones	El usuario tiene que estar logueado en el sistema.
Escenario principal	<ol style="list-style-type: none">1. El Sistema muestra un listado paginado de clientes con los siguientes datos: NIF/CIF, Nombre, Apellidos, Dirección, Ciudad, Provincia, Teléfono, Teléfono2, Email y Fecha de Alta.
Postcondiciones	Se muestra un listado de los clientes.
Escenario alternativo	
Frecuencia	Alta
Estabilidad	Alta
Comentarios	

Tabla 5.52.: UC-06 Ver Mascotas del Cliente

Actores	
Requisitos asociados	
Descripción	Permite a un usuario consultar las mascotas asignadas a un cliente.
Precondiciones	El usuario tiene que estar logueado en el sistema y encontrarse en la ventana del perfil del Cliente.
Escenario principal	<ol style="list-style-type: none">1. El usuario pulsa sobre la pestaña «Mascotas».2. El Sistema muestra un listado con las mascotas del cliente.
Postcondiciones	
Escenario alternativo	<p>*. El usuario pulsa sobre el botón «Nuevo».</p> <ol style="list-style-type: none">1. El Sistema abre la ventana con el UC-xx Alta Mascota.

Frecuencia	Baja
Estabilidad	Alta
Comentarios	

Tabla 5.53.: UC-07 Ver Citas del Cliente

Actores	
Requisitos asociados	
Descripción	Permite a un usuario consultar las citas asignadas a un cliente.
Precondiciones	El usuario tiene que estar logueado en el sistema y encontrarse en la ventana del perfil del Cliente.
Escenario principal	<ol style="list-style-type: none"> 1. El usuario pulsa sobre la pestaña «Citas». 2. El Sistema muestra un listado con las citas del cliente.
Postcondiciones	
Escenario alternativo	<p>*. El usuario pulsa sobre el botón «Nuevo».</p> <ol style="list-style-type: none"> 1. El Sistema abre la ventana con el UC-xx Nueva Cita.
Frecuencia	Baja
Estabilidad	Alta
Comentarios	

Tabla 5.54.: UC-08 Ver Ventas del Cliente

Actores	
Requisitos asociados	
Descripción	Permite a un usuario consultar las ventas realizadas por un cliente.
Precondiciones	El usuario tiene que estar logueado en el sistema y encontrarse en la ventana del perfil del Cliente.

5. Desarrollo del proyecto

Escenario principal	<ol style="list-style-type: none">1. El usuario pulsa sobre la pestaña «Ventas».2. El Sistema muestra un listado con las ventas del cliente.
Postcondiciones	
Escenario alternativo	<p>*. El usuario pulsa sobre la Venta con el botón derecho y hace clic en «Ir a».</p> <ol style="list-style-type: none">1. El Sistema le lleva a la ventana con la información de la Venta seleccionada UC-xx.
Frecuencia	Baja
Estabilidad	Alta
Comentarios	

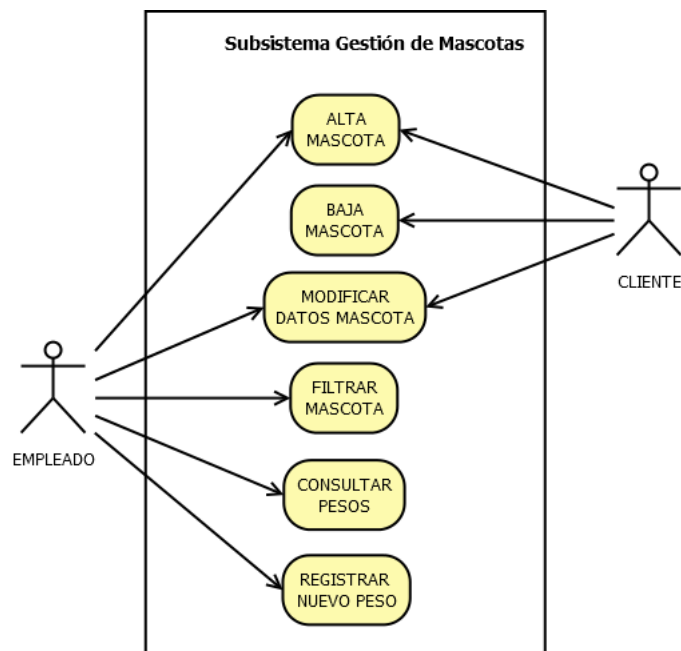


Figura 5.4.: Diagrama Gestión de Mascotas

5.3.1.2 Subsistema Gestión de Mascotas

En la siguiente sección se describirán los detalles referente al subsistema Gestión de Mascotas:

Tabla 5.55.: UC-09 Alta Mascota

Actores	
Requisitos asociados	
Descripción	Permite a un usuario dar de alta a una nueva mascota en el sistema.
Precondiciones	El usuario tiene que estar logueado en el sistema y situado en la venta de Cliente.

5. Desarrollo del proyecto

Escenario principal	<ol style="list-style-type: none"> 1. El cliente acude a la clínica y aporta los datos de su mascota al usuario. 2. El usuario introduce los datos de la mascota en el sistema: N° Chip, Nombre, Fecha de nacimiento, Sexo, Peso, Altura, Especie, Raza, Pelo y observaciones. 3. El Sistema comprueba que no hay espacios en blanco en los campos obligatorios: N° Chip, Nombre, Fecha de nacimiento y Sexo, y que no existe ninguna Mascota con el mismo N° Chip. 4. El Sistema lo registra.
Postcondiciones	Se registra a la mascota en el sistema con los datos aportados.
Escenario alternativo	<ol style="list-style-type: none"> 3a. El sistema detecta que existen espacios en blanco en algún campo obligatorio. Vuelve al paso 1). 3b. El N° Chip introducido ya se encuentra registrado en el sistema. Vuelve al paso 1). <p>*. En cualquier momento el usuario puede cancelar el alta, pulsando en el boton «Cerrar» en la esquina superior derecha.</p>
Frecuencia	Media
Estabilidad	Alta
Comentarios	

Tabla 5.56.: UC-10 Baja Mascota

Actores	
Requisitos asociados	
Descripción	Permite a un usuario dar de baja a una mascota registrado en el sistema.
Precondiciones	El usuario tiene que estar logueado en el sistema y situado en la ventana «Mascotas».

Escenario principal	<ol style="list-style-type: none"> 1. El usuario selecciona del listado mostrado por el Sistema la mascota que desea dar de baja y pulsa sobre el botón «Modificar». 2. El Sistema activa el botón «Eliminar». 3. El usuario pulsa sobre el botón «Eliminar». 4. El Sistema deshabilita el botón «Eliminar» y da de baja a la mascota del Sistema.
Postcondiciones	La mascota es dada de baja en el Sistema.
Escenario alternativo	
Frecuencia	Baja
Estabilidad	Alta
Comentarios	

Tabla 5.57.: UC-11 Filtrar Mascota

Actores	
Requisitos asociados	
Descripción	Permite a un usuario hacer un filtrado con los datos introducidos.
Precondiciones	El usuario tiene que estar logueado en el sistema.
Escenario principal	<ol style="list-style-type: none"> 1. El usuario introduce los datos en los campos que desea filtrar. 2. El Sistema muestra un listado de las mascotas que contienen los datos introducidos.
Postcondiciones	Se muestra un listado de las mascotas.
Escenario alternativo	2a. Los datos introducidos no coinciden con ninguna mascota y no se muestra ninguna Mascota.
Frecuencia	Media
Estabilidad	Alta
Comentarios	

5. Desarrollo del proyecto

Tabla 5.58.: UC-12 Modificar Mascota

Actores	
Requisitos asociados	
Descripción	Permite a un usuario modificar los datos sobre una mascota.
Precondiciones	El usuario tiene que estar logueado en el sistema.
Escenario principal	<ol style="list-style-type: none">1. El usuario selecciona a una mascota específica de un listado que muestra el Sistema y hace doble click sobre él.2. El Sistema muestra el perfil de la Mascota.3. El usuario pulsa en el botón «Modificar».4. El Sistema habilita los campos para que se pueda modificar.5. El usuario modifica los campos los campos necesarios.6. El Sistema comprueba que los campos obligatorios no están vacíos.7. El usuario pulsa en el botón «Guardar».8. El Sistema deshabilita los campos y el botón «Guardar». A continuación guarda los datos modificados en el sistema.
Postcondiciones	Se modifica en el Sistema la información de la Mascota con los datos especificados.
Escenario alternativo	<ol style="list-style-type: none">6a. El Sistema detecta que hay algún campo obligatorio vacío en aquellos que son obligatorios y muestra el Error.8a. El Sistema detecta que existen campos vacíos en aquellos que son obligatorios, no se realiza la modificación.
Frecuencia	Baja
Estabilidad	Alta

Comentarios	
-------------	--

Tabla 5.59.: UC-13 Listar Mascotas

Actores	
Requisitos asociados	
Descripción	Permite a un usuario examinar la lista de mascotas.
Precondiciones	El usuario tiene que estar logueado en el sistema.
Escenario principal	<ol style="list-style-type: none"> 1. El Sistema muestra un listado paginado de mascotas con los siguientes datos: N° Chip, Nombre, Propietario, Sexo, Fecha nacimiento, Fecha defunción, Raza, Especie, Fecha alta, Fecha baja.
Postcondiciones	Se muestra un listado de las mascotas.
Escenario alternativo	
Frecuencia	Alta
Estabilidad	Alta
Comentarios	

Tabla 5.60.: UC-14 Consultar Pesos

Actores	
Requisitos asociados	
Descripción	Permite a un usuario consultar los pesos de las mascotas.
Precondiciones	El usuario tiene que estar logueado en el sistema.

5. Desarrollo del proyecto

Escenario principal	<ol style="list-style-type: none">1. El usuario selecciona la mascota del listado mostrado por el sistema y pulsa con el botón derecho del ratón.2. El Sistema muestra el menú emergente.3. El usuario pulsa sobre «Ver Pesos».4. El Sistema muestra una ventana emergente con un listado de los pesos registrados en el Sistema.
Postcondiciones	
Escenario alternativo	
Frecuencia	Baja
Estabilidad	Alta
Comentarios	

Tabla 5.61.: UC-15 Registrar Nuevo Peso

Actores	
Requisitos asociados	
Descripción	Permite a un usuario registrar nuevos pesos de una mascota.
Precondiciones	El usuario tiene que estar logueado en el sistema.

Escenario principal	<ol style="list-style-type: none"> 1. El usuario selecciona la mascota del listado mostrado por el sistema y pulsa con el botón derecho del ratón. 2. El Sistema muestra el menú emergente. 3. El usuario pulsa sobre «Ver Pesos». 4. El Sistema muestra una ventana emergente con un listado paginado con los pesos registrados en el Sistema con los campos: Fecha y Peso. 5. El usuario introduce el peso en kilogramos de la mascota y a continuación pulsa sobre el botón «Nuevo». 6. El Sistema registra el Nuevo Peso.
Postcondiciones	
Escenario alternativo	*. En cualquier momento el usuario puede Cancelar el registro pulsando sobre el botón «Cerrar» en la esquina superior derecha.
Frecuencia	Baja
Estabilidad	Alta
Comentarios	

5. Desarrollo del proyecto

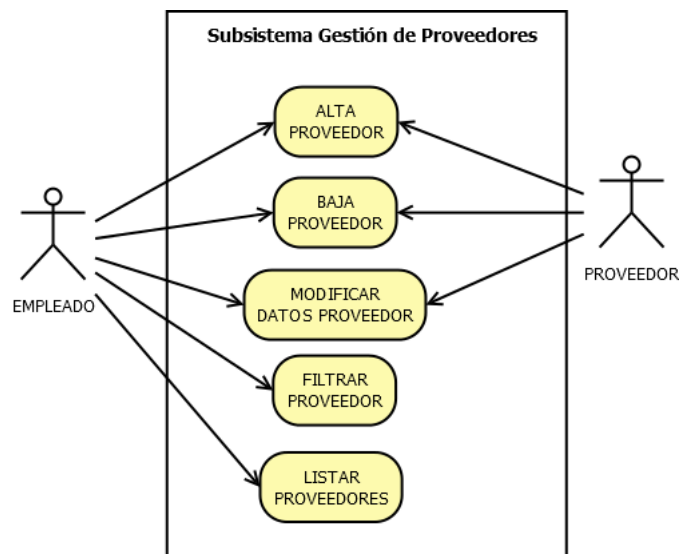


Figura 5.5.: Diagrama Gestión de Proveedores

5.3.1.3 Subsistema de Gestión de Proveedores

En el siguiente apartado lo dedicaremos a los casos de uso referentes a los proveedores.

Tabla 5.62.: UC-16 Alta Proveedor

Actores	
Requisitos asociados	
Descripción	Permite a un usuario dar de alta a un nuevo proveedor en el sistema.
Precondiciones	El usuario tiene que estar logueado en el sistema.

Escenario principal	<ol style="list-style-type: none"> 1. El cliente acude a la clínica y aporta sus datos al usuario. 2. El usuario introduce los datos del proveedor en el sistema: CIF, Nombre, Dirección, Población, Provincia, Teléfono, Móvil, Email y persona de Contacto. 3. El Sistema comprueba que no hay espacios en blanco en los campos obligatorios: CIF, Nombre, Dirección, Población, Provincia, Teléfono y Email, y que no existe ningún Cliente con el mismo NIF/CIF. 4. El Sistema lo registra.
Postcondiciones	Se registra al proveedor en el sistema con los datos aportados.
Escenario alternativo	<ol style="list-style-type: none"> 3a. El sistema detecta que existen espacios en blanco en algún campo obligatorio. Vuelve al paso 1). 3b. El CIF introducido ya se encuentra registrado en el sistema. Vuelve al paso 1). <p>*. En cualquier momento el usuario puede cancelar el alta, pulsando en el boton «Cerrar» en la esquina superior derecha.</p>
Frecuencia	Media
Estabilidad	Alta
Comentarios	

Tabla 5.63.: UC-17 Baja Proveedor

Actores	
Requisitos asociados	
Descripción	Permite a un usuario dar de baja a un cliente registrado en el sistema.
Precondiciones	El usuario tiene que estar logueado en el sistema.

5. Desarrollo del proyecto

Escenario principal	<ol style="list-style-type: none"> 1. El usuario hace un filtrado con los datos del proveedor en el Sistema. 2. El Sistema muestra un listado con los datos de filtrado introducidos y el usuario selecciona el cliente. 3. El usuario selecciona en «Eliminar» y se abre un mensaje emergente preguntando la confirmación de eliminación. 4. El usuario confirma la eliminación y se elimina del Sistema.
Postcondiciones	El proveedor es dado de baja en el Sistema.
Escenario alternativo	<ol style="list-style-type: none"> 3a. En el listado mostrado no se encuentra el proveedor, se muestra un listado en blanco. Se vuelve al paso 2). 4a. El usuario no confirma la eliminación del proveedor. No se realiza ningún cambio en el Sistema.
Frecuencia	Baja
Estabilidad	Alta
Comentarios	

Tabla 5.64.: UC-18 Filtrar Proveedor

Actores	
Requisitos asociados	
Descripción	Permite a un usuario hacer un filtrado con los datos introducidos.
Precondiciones	El usuario tiene que estar logueado en el sistema.
Escenario principal	<ol style="list-style-type: none"> 1. El usuario introduce los datos en los campos que desea filtrar. 2. El Sistema muestra un listado de los proveedores que contienen los datos introducidos.

5.3. Análisis del sistema

Postcondiciones	Se muestra un listado de los proveedores con los datos contenidos.
Escenario alternativo	2a. Los datos introducidos no coinciden con ningún Proveedor y no se muestra ningún Proveedor.
Frecuencia	Media
Estabilidad	Alta
Comentarios	

5. Desarrollo del proyecto

Tabla 5.65.: UC-19 Modificar Proveedor

Actores	El usuario introduce los datos en los campos que desea filtrar. El Sistema muestra un listado de los proveedores que contienen los datos introducidos.
Requisitos asociados	
Descripción	Permite a un usuario modificar los datos sobre un proveedor.
Precondiciones	El usuario tiene que estar logueado en el sistema.
Escenario principal	<ol style="list-style-type: none">1. El usuario selecciona a un proveedor específico de un listado que muestra el Sistema y hace doble click sobre él.2. El Sistema muestra una ventana emergente con los datos del Proveedor.3. El usuario modifica los campos necesarios.4. El Sistema comprueba que los campos obligatorios no están vacíos.5. El usuario pulsa en el botón «Guardar».6. El Sistema deshabilita los campos, el botón «Guardar» y cierra la ventana emergente. A continuación guarda los datos modificados en el sistema.
Postcondiciones	Se modifica en el Sistema la información del Proveedor con los datos especificados.
Escenario alternativo	<ol style="list-style-type: none">6a. El Sistema detecta que hay algún campo obligatorio vacío en aquellos que son obligatorios y muestra el Error.6b. El Sistema detecta que existen campos vacíos en aquellos que son obligatorios, no se realiza la modificación.
Frecuencia	Baja
Estabilidad	Alta

Comentarios	
-------------	--

Tabla 5.66.: UC-20 Listar Proveedores

Actores	
Requisitos asociados	
Descripción	Permite a un usuario examinar la lista de proveedores.
Precondiciones	El usuario tiene que estar logueado en el sistema.
Escenario principal	<ol style="list-style-type: none"> 1. El Sistema muestra un listado paginado de proveedores con los siguientes datos: CIF, Nombre, Apellidos, Dirección, Ciudad, Provincia, Teléfono, Teléfono2, Fax, Email y Contacto.
Postcondiciones	Se muestra un listado de los proveedores.
Escenario alternativo	
Frecuencia	Alta
Estabilidad	Alta
Comentarios	

5.3.1.4 Subsistema de Gestión de Pedidos

Tabla 5.67.: UC-21 Registrar Nuevo Pedido

Actores	Empleado
Requisitos asociados	
Descripción	Permite a un usuario registrar un nuevo pedido en el sistema y los productos de los que se compone el pedido.
Precondiciones	El usuario tiene que estar logueado en el sistema.

5. Desarrollo del proyecto

Escenario principal	<ol style="list-style-type: none">1. El usuario se encuentra en la ventana de Pedidos.2. El usuario busca el proveedor al que hacer el pedido, haciendo click en el botón con forma de lupa en la esquina superior izquierda.3. El Sistema abre una ventana emergente con una lista de proveedores registrados, con los siguientes datos: CIF, Nombre y Población.4. El usuario selecciona el proveedor de la lista, y a continuación pulsa sobre el botón «Enviar» de la esquina superior izquierda.5. El Sistema cierra la venta emergente y en el campo Proveedor queda seleccionado el proveedor.6. El usuario pulsa en el botón «Nuevo».7. El Sistema redirige a la ventana Descripción de Pedido con la información referente al pedido, mostrando un listado en blanco.8. El usuario para añadir un nuevo producto en el pedido, lo busca pulsando en el botón con forma de lupa, dentro del epígrafe «Producto».9. El Sistema abre una venta emergente con un listado de productos pertenecientes al Proveedor seleccionado en el paso 4.10. El usuario selecciona del listado uno o varios productos para el pedido y hace click en el botón «Confirmar».11. El sistema cierra la venta emergente y añade los productos seleccionados en el listado del pedido con una cantidad predeterminada de 1.
---------------------	---

Postcondiciones	Se registra un nuevo pedido en el sistema con los productos seleccionados por el usuario.
Escenario alternativo	
Frecuencia	Media
Estabilidad	Alta
Comentarios	

Tabla 5.68.: UC-22 Cambiar Estado Pagado

Actores	Empleado
Requisitos asociados	
Descripción	Permite a un usuario cambiar el estado de un pedido.
Precondiciones	El usuario tiene que estar logueado en el sistema.
Escenario principal	<ol style="list-style-type: none"> 1. El usuario selecciona el Pedido del listado haciendo click con el botón derecho sobre él. 2. El Sistema muestra un menú emergente, con el epígrafe «Estado». 3. El usuario se posiona sobre «Estado». 4. El Sistema muestra las opciones disponible: Pagado y Sin Pagar. 5. El usuario pulsa sobre la opción «Pagado». 6. El Sistema modifica el estado del pedido.
Postcondiciones	
Escenario alternativo	
Frecuencia	Media
Estabilidad	Alta
Comentarios	

Tabla 5.69.: UC-23 Modificar Pedido

Actores	Empleado
---------	----------

5. Desarrollo del proyecto

Requisitos asociados	
Descripción	Permite a un usuario modificar un pedido.
Precondiciones	El usuario tiene que estar logueado en el sistema.
Escenario principal	<ol style="list-style-type: none"> 1. El usuario selecciona el Pedido del listado haciendo doble click sobre él. 2. El Sistema muestra la descripción detallada del Pedido. 3. El usuario se posiona sobre uno de los productos del Pedido, hace click sobre él y pulsa en «Modificar». 4. El Sistema habilita el botón «Guardar» y «Eliminar». 5. El usuario modifica la cantidad de producto solicitado al Proveedor y a continuación pulsa sobre «Guardar». 6. El Sistema guarda los cambios en el sistema.
Postcondiciones	
Escenario alternativo	<ol style="list-style-type: none"> 5a. El usuario desea añadir un nuevo producto en el pedido: se realiza el caso de uso UC-21 Paso 8. (Ver Paso 8) 5b. El usuario pulsa el botón «Eliminar». <ol style="list-style-type: none"> 1. El Sistema elimina el producto del pedido.
Frecuencia	Baja
Estabilidad	Alta
Comentarios	

Tabla 5.70.: UC-24 Filtrar Pedido

Actores	Empleado
Requisitos asociados	
Descripción	Permite a un usuario hacer un filtrado con los datos introducidos.

Precondiciones	El usuario tiene que estar logueado en el sistema.
Escenario principal	<ol style="list-style-type: none"> 1. El usuario introduce los datos en los campos que desea filtrar. 2. El Sistema muestra un listado de los pedidos que contienen los datos introducidos.
Postcondiciones	
Escenario alternativo	2a. Los datos introducidos no coinciden con ningún Pedido y no se muestra ningún Pedido.
Frecuencia	Baja
Estabilidad	Alta
Comentarios	

Tabla 5.71.: UC-25 Listar Pedidos

Actores	Empleado
Requisitos asociados	
Descripción	Permite a un usuario examinar la lista de pedido.
Precondiciones	El usuario tiene que estar logueado en el sistema.
Escenario principal	<ol style="list-style-type: none"> 1. El Sistema muestra un listado paginado de pedidos con los siguientes datos: Número, Coste total, Número de artículos, Fecha, Fecha de entrega, Fecha del pago, Proveedor, Pagado, Empleado.
Postcondiciones	Se muestra un listado de los pedidos.
Escenario alternativo	
Frecuencia	Alta
Estabilidad	Alta
Comentarios	

Tabla 5.72.: UC-26 Eliminar Pedido

5. Desarrollo del proyecto

Actores	Empleado
Requisitos asociados	
Descripción	Permite a un usuario eliminar un pedido.
Precondiciones	El usuario tiene que estar logueado en el sistema.
Escenario principal	<ol style="list-style-type: none">1. El usuario selecciona el pedido de la lista.2. El Sistema habilita el botón «Eliminar».3. El usuario pulsa sobre el botón «Eliminar».4. El Sistema abre un mensaje emergente para que el usuario confirme la eliminación.5. El usuario confirma la eliminación.6. El Sistema elimina el pedido.
Postcondiciones	Se elimina el pedido del sistema.
Escenario alternativo	
Frecuencia	Baja
Estabilidad	Alta
Comentarios	

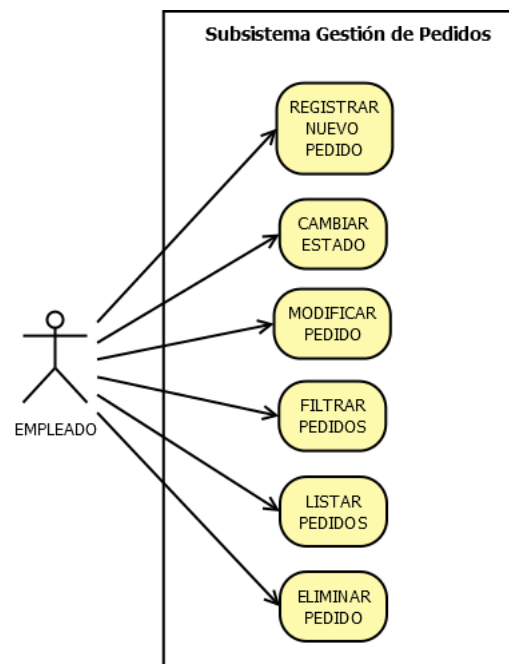


Figura 5.6.: Diagrama Gestión de Pedidos

5. Desarrollo del proyecto

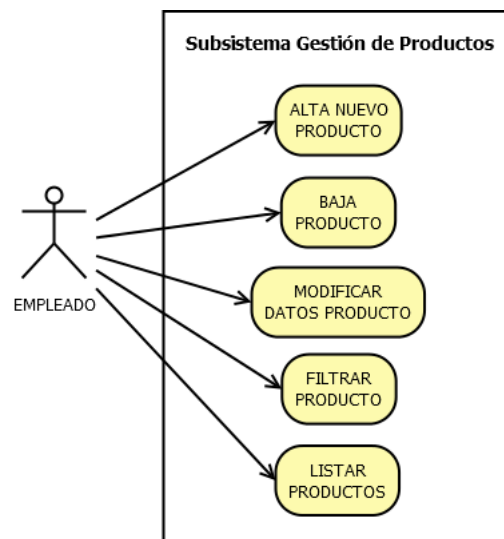


Figura 5.7.: Diagrama Gestión de Productos

5.3.1.5 Subsistema de Gestión de Productos

Tabla 5.73.: UC-27 Alta Nuevo Producto

Actores	Empleado
Requisitos asociados	
Descripción	Permite a un usuario registrar un nuevo producto en el sistema.
Precondiciones	El usuario tiene que estar logueado en el sistema.

Escenario principal	<ol style="list-style-type: none"> 1. El usuario se encuentra en la ventana de Productos y busca el proveedor al que pertenece el producto. 2. El Sistema abre una ventana con un listado de los proveedores. 3. El usuario selecciona el proveedor, pulsa «Enviar» y cierra la ventana. 4. El usuario hace click sobre el botón «Nuevo» en la esquina superior izquierda. 5. El Sistema abre una ventana emergente con los campos en blanco para que el usuario introduzca los datos del producto: Nombre, Familia, Proveedor, IVA, Precio, PVP, Stock e Imagen. 6. El usuario introduce los datos del producto y a continuación pulsa sobre «Guardar». 7. El Sistema comprueba que no existen espacios en blanco, cierra la venta emergente y guarda los datos.
Postcondiciones	Se registra un nuevo producto en el sistema con los datos especificados por el usuario.
Escenario alternativo	
Frecuencia	Media
Estabilidad	Alta
Comentarios	

5. Desarrollo del proyecto

Tabla 5.74.: UC-28 Baja Producto

Actores	Empleado
Requisitos asociados	
Descripción	Permite a un usuario dar de baja a un producto en el sistema.
Precondiciones	El usuario tiene que estar logueado en el sistema.
Escenario principal	<ol style="list-style-type: none">1. El usuario selecciona el Producto y pulsa sobre el botón «Eliminar».2. El Sistema muestra un mensaje pidiendo la confirmación por parte del usuario.3. El usuario confirma la eliminación.4. El Sistema elimina el producto.
Postcondiciones	
Escenario alternativo	<ol style="list-style-type: none">3a. El usuario cancela la eliminación.<ol style="list-style-type: none">1. El producto se mantiene en el Sistema.
Frecuencia	Media
Estabilidad	Alta
Comentarios	

Tabla 5.75.: UC-29 Modificar Producto

Actores	Empleado
Requisitos asociados	
Descripción	Permite a un usuario modificar un producto.
Precondiciones	El usuario tiene que estar logueado en el sistema.

Escenario principal	<ol style="list-style-type: none"> 1. El usuario selecciona el Producto del listado haciendo doble click sobre él. 2. El Sistema abre una ventana emergente con los datos del Producto. 3. El usuario modifica los campos requeridos y pulsa en «Guardar». 4. El Sistema comprueba que no haya campos vacíos, deshabilita los campos y guarda la datos modificados. 5. El usuario cierra la ventana emergente.
Postcondiciones	
Escenario alternativo	<ol style="list-style-type: none"> 3a. El usuario cancela la modificación de los datos y no se producen cambios en el producto. 4a. El Sistema detecta que existen campos vacíos. <ol style="list-style-type: none"> 1. El sistema muestra el error.
Frecuencia	Baja
Estabilidad	Alta
Comentarios	

Tabla 5.76.: UC-30 Filtrar Producto

Actores	Empleado
Requisitos asociados	
Descripción	Permite a un usuario hacer un filtrado con los datos introducidos.
Precondiciones	El usuario tiene que estar logueado en el sistema.
Escenario principal	<ol style="list-style-type: none"> 1. El usuario introduce los datos en los campos que desea filtrar. 2. El Sistema muestra un listado de los productos que contienen los datos introducidos.

5. Desarrollo del proyecto

Postcondiciones	
Escenario alternativo	2a. Los datos introducidos no coinciden con ningún Producto y no se muestra ningún Producto.
Frecuencia	Media
Estabilidad	Alta
Comentarios	

Tabla 5.77.: UC-31 Listar Productos

Actores	Empleado
Requisitos asociados	
Descripción	Permite a un usuario examinar la lista de productos.
Precondiciones	El usuario tiene que estar logueado en el sistema.
Escenario principal	1. El Sistema muestra un listado paginado de productos con los siguientes datos: Código, Nombre, Familia, Proveedor, PVP, IVA, Precio, Stock y Fecha alta.
Postcondiciones	Se muestra un listado de los productos.
Escenario alternativo	
Frecuencia	Alta
Estabilidad	Alta
Comentarios	

5.3.1.6 Subsistema de Gestión de Usuarios

Tabla 5.78.: UC-32 Alta Nuevo Usuario

Actores	Administrador (Principal), Empleado (Secundario)
Requisitos asociados	
Descripción	Permite a un usuario registrar un nuevo usuario en el sistema.
Precondiciones	El administrador tiene que estar logueado en el sistema.

Escenario principal	<ol style="list-style-type: none"> 1. El administrador se encuentra en la ventana de Usuarios y hace click sobre el botón «Nuevo» en la esquina superior izquierda. 2. El Sistema abre una ventana emergente con los campos en blanco para que el usuario introduzca los datos del usuario: Nick, Contraseña, NIF, Nombre, Apellidos, Rol, Dirección, Provincia, Población, Teléfono, Móvil, Email y Número de la seguridad social. 3. El administrador introduce los datos del usuario y a continuación pulsa sobre «Guardar». 4. El Sistema comprueba que no existen espacios en blanco y guarda los datos. 5. El administrador cierra la ventana emergente.
Postcondiciones	Se registra un nuevo usuario en el sistema con los datos especificados por el administrador.
Escenario alternativo	<ol style="list-style-type: none"> 3a. El administrador no pulsa en «Guardar». <ol style="list-style-type: none"> 1. No se añade el nuevo usuario en el sistema. 4a. El Sistema detecta que existen espacios en blanco y muestra el error.
Frecuencia	Baja
Estabilidad	Alta
Comentarios	

5. Desarrollo del proyecto

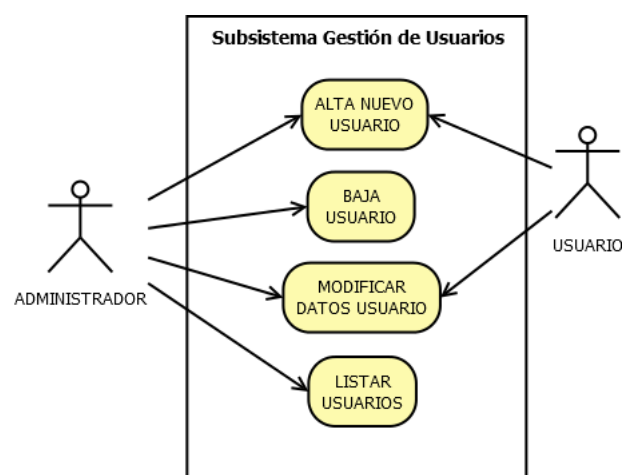


Figura 5.8.: Diagrama Gestión de Usuarios

Tabla 5.79.: UC-33 Baja Usuario

Actores	Administrador
Requisitos asociados	
Descripción	Permite a un administrador dar de baja a un usuario en el sistema.
Precondiciones	El administrador tiene que estar logueado en el sistema.
Escenario principal	<ol style="list-style-type: none"> 1. El administrador selecciona el usuario y pulsa sobre el botón «Eliminar». 2. El Sistema muestra un mensaje solicitando la confirmación por parte del administrador. 3. El administrador confirma la eliminación. 4. El Sistema elimina el usuario.
Postcondiciones	
Escenario alternativo	<ol style="list-style-type: none"> 2a. El usuario seleccionado tiene el rol de Administrador. <ol style="list-style-type: none"> 1. El Sistema muestra el error. 3a. El administrador cancela la eliminación. <ol style="list-style-type: none"> 1. El usuario se mantiene en el Sistema.
Frecuencia	Media
Estabilidad	Alta
Comentarios	

Tabla 5.80.: UC-34 Modificar Datos Usuario

Actores	Administrador (Principal), Empleado(Secundario)
Requisitos asociados	
Descripción	Permite a un administrador modificar un usuario.
Precondiciones	El administrador tiene que estar logueado en el sistema.

5. Desarrollo del proyecto

Escenario principal	<ol style="list-style-type: none"> 1. El administrador selecciona el usuario del listado haciendo doble click sobre él. 2. El Sistema abre una ventana emergente con los datos del Usuario. 3. El administrador modifica los campos requeridos y pulsa en «Guardar». 4. El Sistema comprueba que no haya campos vacíos, deshabilita los campos y guarda la datos modificados. 5. El administrador cierra la ventana emergente.
Postcondiciones	
Escenario alternativo	<ol style="list-style-type: none"> 3a. El administrador cancela la modificación de los datos y no se producen cambios en el usuario. 4a. El Sistema detecta que existen campos vacíos. <ol style="list-style-type: none"> 1. El sistema muestra el error.
Frecuencia	Baja
Estabilidad	Alta
Comentarios	

Tabla 5.81.: UC-35 Listar Usuarios

Actores	Administrador
Requisitos asociados	
Descripción	Permite a un administrador examinar la lista de usuarios.
Precondiciones	El administrador tiene que estar logueado en el sistema.

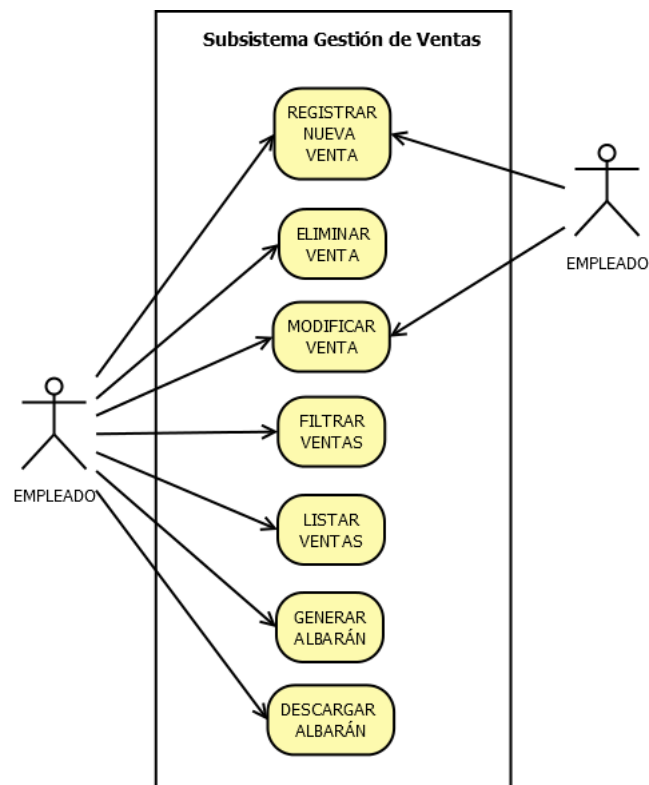


Figura 5.9.: Diagrama Gestión de Ventas

Escenario principal	1. El Sistema muestra un listado paginado de usuarios con los siguientes datos: Usuario, Nombre, Apellidos, NIF, Rol, Dirección, Población, Provincia, Teléfono, Móvil, Email, Número de la seguridad social, Fecha de alta.
Postcondiciones	Se muestra un listado de los usuarios.
Escenario alternativo	
Frecuencia	Media
Estabilidad	Alta
Comentarios	

5.3.1.7 Subsistema de Gestión de Ventas

5. Desarrollo del proyecto

Tabla 5.82.: UC-36 Registrar Nueva Venta

Actores	Empleado (Principal), Cliente (Secundario)
Requisitos asociados	
Descripción	Permite a un empleado registrar una nueva venta en el sistema y los productos de los que se compone la venta.
Precondiciones	El empleado tiene que estar logueado en el sistema.

Escenario principal	<ol style="list-style-type: none"> 1. El usuario pulsa sobre «Ventas>>Nueva Venta» en el Menú. 2. El Sistema crea una nueva venta con un listado vacío y con el campo cliente «Venta Rápida» de forma predeterminada. 3. El usuario busca al cliente de la lista Clientes, haciendo click sobre el botón buscar. 4. El Sistema abre una ventana emergente con un listado de los clientes registrados en el sistema. 5. El usuario selecciona a uno de los clientes del listado y pulsa sobre «Enviar». 6. El sistema modifica el campo cliente, insertando el cliente seleccionado. 7. El usuario para añadir un producto en la venta, pulsa sobre el botón buscar en la parte correspondiente a «Producto». 8. El Sistema abre una ventana emergente con los ítems registrados en el sistema. 9. El usuario selecciona uno o varios ítems del listado paginado haciendo click sobre el checkbox y a continuación pulsa en «Confirmar». 10. El sistema cierra la venta emergente y añade los ítems seleccionados en el listado de la venta con una cantidad predeterminada de 1, descuenta la cantidad en el stock y realiza el cálculo del coste total de la venta. 11. Se realiza los pasos del 7-10 hasta que el usuario da por terminada la venta.
Postcondiciones	Se registra una nueva venta en el sistema con los productos y/o servicios seleccionados por el usuario.

5. Desarrollo del proyecto

Escenario alternativo	<p>*. El cliente recibió un servicio veterinario y quien registró la venta fue el propio veterinario, el campo «Veterinario» tendrá el nombre del veterinario.</p> <p>*. El cliente recibió un servicio veterinario y quien registró la venta fue un empleado no veterinario.</p> <p>1. El empleado selecciona el «Veterinario» del listado de veterinarios registrados en el sistema.</p> <p>7a. El usuario para añadir un servicio en la venta, pulsa sobre el botón buscar en la parte correspondiente a «Servicio».</p> <p>10a. El usuario selecciona el ítem de la línea de venta, modifica la cantidad y pulsa [INTRO].</p> <p>1. El Sistema comprueba que existe cantidad suficiente de stock, registra el cambio de la cantidad del ítem seleccionado y recalcula el coste total de la venta.</p>
Frecuencia	Alta
Estabilidad	Alta
Comentarios	

Tabla 5.83.: UC-37 Modificar Venta

Actores	Empleado (Principal), Cliente (Secundario)
Requisitos asociados	
Descripción	Permite a un usuario modificar una Venta.
Precondiciones	El usuario tiene que estar logueado en el sistema.

Escenario principal	<ol style="list-style-type: none"> 1. El usuario selecciona la Venta del listado haciendo doble click sobre él. 2. El Sistema muestra la descripción detallada de la Venta. 3. El usuario se posiona sobre uno de los ítems de la Venta, hace click sobre él. 4. El Sistema selecciona el ítem. 5. El usuario modifica la cantidad de ítem y a continuación pulsa [INTRO]. 6. El Sistema comprueba que existe stock y guarda los cambios en el sistema.
Postcondiciones	
Escenario alternativo	<ol style="list-style-type: none"> 6a. El Sistema detecta que no hay stock suficiente. <ol style="list-style-type: none"> 1. El Sistema muestra el error y no realiza cambios.
Frecuencia	Baja
Estabilidad	Alta
Comentarios	

Tabla 5.84.: UC-38 Filtrar Venta

Actores	Empleado
Requisitos asociados	
Descripción	Permite a un usuario hacer un filtrado con los datos introducidos.
Precondiciones	El usuario tiene que estar logueado en el sistema.

5. Desarrollo del proyecto

Escenario principal	<ol style="list-style-type: none"> 1. El usuario introduce los datos en los campos que desea filtrar. 2. El Sistema muestra un listado de las ventas que contienen los datos introducidos.
Postcondiciones	
Escenario alternativo	2a. Los datos introducidos no coinciden con ninguna Venta y no se muestra ningun Venta.
Frecuencia	Baja
Estabilidad	Alta
Comentarios	

Tabla 5.85.: UC-39 Listar Ventas

Actores	Empleado
Requisitos asociados	
Descripción	Permite a un usuario examinar la lista de ventas.
Precondiciones	El usuario tiene que estar logueado en el sistema.
Escenario principal	<ol style="list-style-type: none"> 1. El Sistema muestra un listado paginado de ventas con los siguientes datos: Fecha, Hora, Cliente, Facturado, Artículos, Coste, IVA, Coste total, Empleado, Veterinario.
Postcondiciones	Se muestra un listado de las ventas.
Escenario alternativo	
Frecuencia	Alta
Estabilidad	Alta
Comentarios	

Tabla 5.86.: UC-40 Eliminar Venta

Actores	Empleado
Requisitos asociados	

5.3. Análisis del sistema

Descripción	Permite a un usuario eliminar una venta.
Precondiciones	El usuario tiene que estar logueado en el sistema.
Escenario principal	<ol style="list-style-type: none">1. El usuario selecciona una venta de la lista.2. El Sistema habilita el botón «Eliminar».3. El usuario pulsa sobre el botón «Eliminar».4. El Sistema abre un mensaje emergente para que el usuario confirme la eliminación.5. El usuario confirma la eliminación.6. El Sistema elimina el pedido.
Postcondiciones	Se elimina la venta del sistema.
Escenario alternativo	
Frecuencia	Baja
Estabilidad	Alta
Comentarios	

5. Desarrollo del proyecto

Tabla 5.87.: UC-41 Generar Albarán

Actores	Empleado
Requisitos asociados	
Descripción	Permite a un usuario generar el albarán de una venta.
Precondiciones	El usuario tiene que estar logueado en el sistema.
Escenario principal	<ol style="list-style-type: none">1. El usuario selecciona una venta de la lista.2. El Sistema habilita el botón «Generar Albarán».3. El usuario pulsa sobre el botón «Generar Albarán».4. El Sistema genera un fichero PDF con los datos de venta y lo guarda en el sistema.
Postcondiciones	Se genera el albarán de la venta.
Escenario alternativo	
Frecuencia	Baja
Estabilidad	Alta
Comentarios	El formato del Albarán fue proporcionado por el cliente

Tabla 5.88.: UC-42 Descargar Albarán

Actores	Empleado
Requisitos asociados	
Descripción	Permite a un usuario descargar un albarán de una venta.
Precondiciones	El usuario tiene que estar logueado en el sistema.

Escenario principal	<ol style="list-style-type: none"> 1. El usuario selecciona una venta de la lista. 2. El Sistema habilita el botón «Descargar Albarán». 3. El usuario pulsa sobre el botón «Descargar Albarán». 4. El Sistema abre una ventana de descarga. 5. El usuario selecciona dónde desea descargar el albarán en su ordenador, pulsa en «Aceptar» y se inicia la descarga.
Postcondiciones	
Escenario alternativo	<ol style="list-style-type: none"> 4a. El Sistema detecta que no se ha generado el albarán con anterioridad. 1. El Sistema muestra el error.
Frecuencia	Alta
Estabilidad	Alta
Comentarios	

5.3.1.8 Subsistema de Gestión de Citas

Tabla 5.89.: UC-43 Alta Nueva Cita

Actores	Empleado (Principal), Cliente (Secundario)
Requisitos asociados	
Descripción	Permite a un usuario registrar una nueva cita en el sistema.
Precondiciones	El usuario tiene que estar logueado en el sistema.

5. Desarrollo del proyecto

Escenario principal	<ol style="list-style-type: none">1. Un Cliente desea concertar una Cita.2. El usuario selecciona al cliente de la ventan Clientes y pulsa con el botón derecho del ratón.3. El Sistema abre un menú emergente con la opción «Asignar Cita».4. El usuario hace click.5. El Sistema abre una ventana emergente para crear una nueva cita, con los siguientes campos: un select con las mascotas del cliente, fecha, hora, servicio, informe.
Postcondiciones	Se registra una nueva cita en el sistema con los datos especificados por el usuario.
Escenario alternativo	
Frecuencia	Media
Estabilidad	Alta
Comentarios	

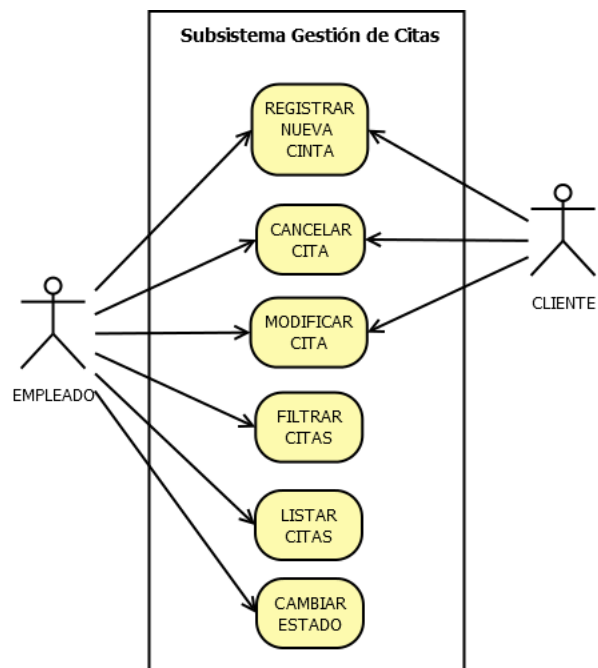


Figura 5.10.: Diagrama Gestión de Citas

5. Desarrollo del proyecto

Tabla 5.90.: UC-44 Cancelar Cita

Actores	Empleado
Requisitos asociados	
Descripción	Permite a un usuario cancelar una cita registrada en el sistema.
Precondiciones	El usuario tiene que estar logueado en el sistema.
Escenario principal	<ol style="list-style-type: none">1. El usuario selecciona una Cita y pulsa con el botón derecho.2. El Sistema muestra un menú emergente con las opciones disponibles: Avisar, Acudido, Pendiente, Cancelar.3. El usuario pulsa sobre «Cancelar».4. El Sistema cancela la cita y deja libre la hora de la cita.
Postcondiciones	
Escenario alternativo	
Frecuencia	Baja
Estabilidad	Alta
Comentarios	

Tabla 5.91.: UC-45 Modificar Cita

Actores	Empleado (Principal), Cliente (Secundario)
Requisitos asociados	
Descripción	Permite a un usuario modificar una cita.
Precondiciones	El usuario tiene que estar logueado en el sistema.

Escenario principal	<ol style="list-style-type: none"> 1. Un Cliente desea modificar una cita 2. El usuario selecciona la cita y hace doble click. 3. El Sistema abre una venta emergente para que el usuario modifique la Cita. 4. El usuario modifica los campos requeridos y pulsa en «Guardar». 5. El Sistema comprueba que no haya campos vacíos, deshabilita los campos y guarda la datos modificados. 6. El usuario cierra la ventana emergente.
Postcondiciones	
Escenario alternativo	<ol style="list-style-type: none"> 4a. El usuario cancela la modificación de los datos y no se producen cambios en la cita. 5a. El Sistema detecta que existen campos vacíos. <ol style="list-style-type: none"> 1. El sistema muestra el error.
Frecuencia	Baja
Estabilidad	Alta
Comentarios	

Tabla 5.92.: UC-46 Filtrar Cita

Actores	Empleado
Requisitos asociados	
Descripción	Permite a un usuario hacer un filtrado con los datos introducidos.
Precondiciones	El usuario tiene que estar logueado en el sistema.

5. Desarrollo del proyecto

Escenario principal	<ol style="list-style-type: none"> 1. El usuario introduce los datos en los campos que desea filtrar. 2. El Sistema muestra un listado de las citas que contienen los datos introducidos.
Postcondiciones	
Escenario alternativo	2a. Los datos introducidos no coinciden con ninguna Cita y no se muestra ninguna Cita.
Frecuencia	Media
Estabilidad	Alta
Comentarios	

Tabla 5.93.: UC-47 Listar Citas

Actores	Empleado
Requisitos asociados	
Descripción	Permite a un usuario examinar la lista de citas.
Precondiciones	El usuario tiene que estar logueado en el sistema.
Escenario principal	<ol style="list-style-type: none"> 1. El Sistema muestra un listado paginado de citas con los siguientes datos: Fecha, Hora, Estado, Cliente, Mascota, Tipo, Observaciones, Empleado.
Postcondiciones	Se muestra un listado de las citas.
Escenario alternativo	
Frecuencia	Alta
Estabilidad	Alta
Comentarios	

Tabla 5.94.: UC-48 Cambiar Estado Cita

Actores	Empleado
Requisitos asociados	

Descripción	Permite a un usuario cambiar el estado de una cita.
Precondiciones	El usuario tiene que estar logueado en el sistema.
Escenario principal	<ol style="list-style-type: none"> 1. El usuario selecciona una Cita y pulsa con el botón derecho. 2. El Sistema muestra un menú emergente con las opciones disponibles: Avisa, Acudido, Pendiente, Cancelar. 3. El usuario pulsa sobre alguna de las opciones. 4. El Sistema cambia el estado de la cita.
Postcondiciones	
Escenario alternativo	
Frecuencia	Baja
Estabilidad	Alta
Comentarios	

5.3.1.9 Subsistema de Gestión de Facturas

Tabla 5.95.: UC-49 Crear Nueva Factura

Actores	Empleado
Requisitos asociados	
Descripción	Permite a un usuario crear una nueva factura en el sistema a partir de una o varias ventas realizadas por un mismo cliente.
Precondiciones	El empleado tiene que estar logueado en el sistema.

5. Desarrollo del proyecto

Escenario principal	<ol style="list-style-type: none">1. El usuario se encuentra en la ventana de «Facturas».2. El usuario busca al cliente de la lista Clientes, haciendo click sobre el botón buscar.3. El Sistema abre una ventana emergente con un listado de los clientes registrados en el sistema.4. El usuario selecciona a uno de los clientes del listado y pulsa sobre «Enviar».5. El sistema modifica el campo Cliente, insertando el cliente seleccionado y habilita el botón «Nuevo».6. El usuario pulsa sobre el botón «Nuevo».7. El Sistema redirige a una nueva ventana, con los detalles de la factura, con los siguientes campos: Código, Nombre, Fecha, Tipo, Descripción, Cantidad, PVP, IVA, Precio Total.8. El usuario para añadir una venta a la factura, debe pulsar sobre el botón buscar en la parte correspondiente a «Venta».9. El Sistema abre una ventana emergente con las ventas no facturadas realizadas por el Cliente.10. El usuario selecciona uno o varias ventas del listado paginado haciendo click sobre el checkbox y a continuación pulsa en «Confirmar».11. El sistema cierra la venta emergente y añade los ítems de los que se compone las ventas seleccionadas. Pone a estado «facturado» las ventas seleccionadas y realiza el cálculo del coste total de la factura, detallando el coste del IVA.
---------------------	--

5.3. Análisis del sistema

Postcondiciones	Se crea una factura en el sistema con las ventas seleccionadas por el usuario.
Escenario alternativo	
Frecuencia	Alta
Estabilidad	Alta
Comentarios	

5. Desarrollo del proyecto

Tabla 5.96.: UC-50 Modificar Factura

Actores	Empleado
Requisitos asociados	
Descripción	Permite a un usuario modificar una Factura.
Precondiciones	El usuario tiene que estar logueado en el sistema.
Escenario principal	<ol style="list-style-type: none">1. El usuario selecciona la Factura del listado haciendo doble click sobre él.2. El Sistema muestra la descripción detallada de la Factura.3. El usuario se posiona sobre uno de los ítems de la Factura, y hace click.4. El Sistema selecciona el ítem.5. El usuario pulsa el botón «Eliminar».6. El Sistema elimina los ítems de la venta de la Factura y recalcula el coste de la factura y del IVA
Postcondiciones	

Escenario alternativo	<p>3-6. El usuario para añadir una venta a la factura, pulsa sobre el botón buscar en la parte correspondiente a «Venta».</p> <ol style="list-style-type: none"> 1. El Sistema abre una ventana emergente con las ventas no facturadas realizadas por el Cliente. 2. El usuario selecciona uno o varias ventas del listado paginado haciendo click sobre el checkbox y a continuación pulsa en «Confirmar». 3. El sistema cierra la venta emergente y añade los ítems de los que se compone las ventas seleccionadas. Pone a estado «facturado» las ventas seleccionadas y realiza el cálculo del coste total de la factura y del coste del IVA.
Frecuencia	Baja
Estabilidad	Alta
Comentarios	

Tabla 5.97.: UC-38 Filtrar Factura

Actores	Empleado
Requisitos asociados	
Descripción	Permite a un usuario hacer un filtrado con los datos introducidos.
Precondiciones	El usuario tiene que estar logueado en el sistema.
Escenario principal	<ol style="list-style-type: none"> 1. El usuario introduce los datos en los campos que desea filtrar. 2. El Sistema muestra un listado de las facturas que contienen los datos introducidos.
Postcondiciones	

5. Desarrollo del proyecto

Escenario alternativo	2a. Los datos introducidos no coinciden con ninguna Factura. 1. No se muestra ninguna Venta.
Frecuencia	Baja
Estabilidad	Alta
Comentarios	

Tabla 5.98.: UC-52 Listar Facturas

Actores	Empleado
Requisitos asociados	
Descripción	Permite a un usuario examinar la lista de facturas.
Precondiciones	El usuario tiene que estar logueado en el sistema.
Escenario principal	1. El Sistema muestra un listado paginado de las facturas con los siguientes datos: Id, Número, Fecha, Coste, IVA, Coste total, Cliente, Empleado.
Postcondiciones	Se muestra un listado de las facturas.
Escenario alternativo	
Frecuencia	Alta
Estabilidad	Alta
Comentarios	

Tabla 5.99.: UC-53 Eliminar Factura

Actores	Empleado
Requisitos asociados	
Descripción	Permite a un usuario eliminar una factura.
Precondiciones	El usuario tiene que estar logueado en el sistema.

Escenario principal	<ol style="list-style-type: none"> 1. El usuario selecciona una factura de la lista. 2. El Sistema habilita el botón «Eliminar». 3. El usuario pulsa sobre el botón «Eliminar». 4. El Sistema abre un mensaje emergente para que el usuario confirme la eliminación. 5. El usuario confirma la eliminación. 6. El Sistema elimina la factura.
Postcondiciones	Se elimina la factura del sistema.
Escenario alternativo	
Frecuencia	Baja
Estabilidad	Alta
Comentarios	

5. Desarrollo del proyecto

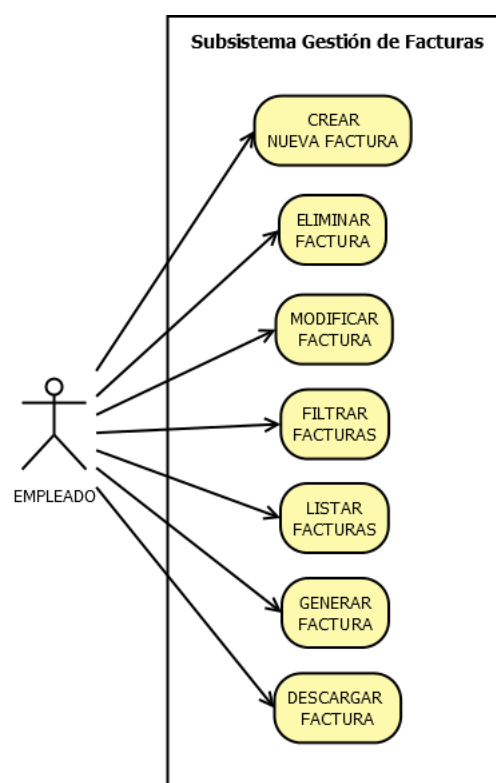


Figura 5.11.: Diagrama Gestión de Facturas

Tabla 5.100.: UC-54 Generar Factura

Actores	Empleado
Requisitos asociados	
Descripción	Permite a un usuario generar una factura.
Precondiciones	El usuario tiene que estar logueado en el sistema.
Escenario principal	<ol style="list-style-type: none"> 1. El usuario selecciona una factura de la lista. 2. El Sistema habilita el botón «Generar Factura». 3. El usuario pulsa sobre el botón «Generar Factura». 4. El Sistema genera un fichero PDF con los datos de venta y lo guarda en el sistema.
Postcondiciones	Se genera la factura de las ventas.
Escenario alternativo	
Frecuencia	Baja
Estabilidad	Alta
Comentarios	El formato de la Factura fue aprobado por el cliente.

Tabla 5.101.: UC-55 Descargar Factura

Actores	Empleado
Requisitos asociados	
Descripción	Permite a un usuario descargar una factura.
Precondiciones	El usuario tiene que estar logueado en el sistema.

5. Desarrollo del proyecto

Escenario principal	<ol style="list-style-type: none">1. El usuario selecciona una factura de la lista.2. El Sistema habilita el botón «Descargar Factura».3. El usuario pulsa sobre el botón «Descargar Factura».4. El Sistema abre una ventana de descarga.5. El usuario selecciona dónde desea descargar el fichero en su ordenador, pulsa en «Aceptar» y se inicia la descarga.
Postcondiciones	
Escenario alternativo	<ol style="list-style-type: none">4a. El Sistema detecta que no se ha generado el fichero con anterioridad.<ol style="list-style-type: none">1. El Sistema muestra el error.
Frecuencia	Alta
Estabilidad	Alta
Comentarios	

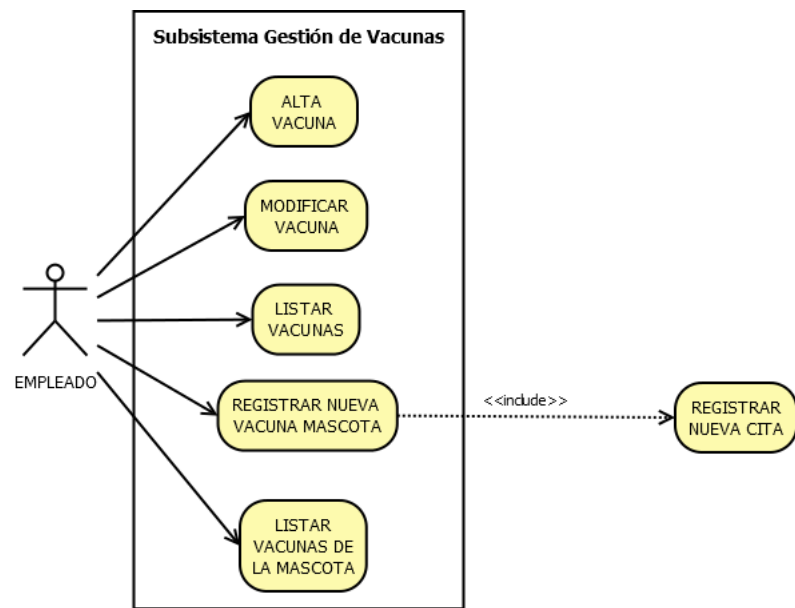


Figura 5.12.: Diagrama Gestión de Vacunas

5.3.1.10 Subsistema de Gestión de Vacunas

Tabla 5.102.: UC-56 Alta Vacuna

Actores	Empleado
Requisitos asociados	
Descripción	Permite a un usuario dar de alta una nueva vacuna en el sistema.
Precondiciones	El empleado tiene que estar logueado en el sistema.

5. Desarrollo del proyecto

Escenario principal	<ol style="list-style-type: none"> 1. El usuario se pulsa en «Configuración > Vacuna nueva». 2. El Sistema abre una ventana emergente con un listado de las vacunas registradas en el sistema. 3. El usuario introduce los datos de la nueva vacuna y a continuación pulsa en «Nuevo». 4. El sistema añade una nueva vacuna con los datos aportados por el usuario.
Postcondiciones	
Escenario alternativo	
Frecuencia	Baja
Estabilidad	Alta
Comentarios	

Tabla 5.103.: UC-57 Modificar Vacuna

Actores	Empleado
Requisitos asociados	
Descripción	Permite a un usuario modificar una Vacuna.
Precondiciones	El usuario tiene que estar logueado en el sistema.
Escenario principal	<ol style="list-style-type: none"> 1. El usuario selecciona una vacuna de la ventana «Vacuna nueva» , modifica los datos necesarios y pulsa en «Modificar». 2. El Sistema comprueba que no existen espacios en blanco en los campos obligatorios y guarda los cambios.
Postcondiciones	
Escenario alternativo	2a. El Sistema detecta que existen espacios en blanco y muestra el error.
Frecuencia	Baja

Estabilidad	Alta
Comentarios	

Tabla 5.104.: UC-58 Listar Vacunas

Actores	Empleado
Requisitos asociados	
Descripción	Permite a un usuario examinar la lista de vacunas dadas de alta en el sistema.
Precondiciones	El usuario tiene que estar logueado en el sistema.
Escenario principal	<ol style="list-style-type: none"> 1. El Sistema muestra un listado de las vacunas con los siguientes datos: Vacuna, Días, Especie, Descripción.
Postcondiciones	Se muestra un listado de las vacunas.
Escenario alternativo	
Frecuencia	Baja
Estabilidad	Alta
Comentarios	

Tabla 5.105.: UC-59 Registrar Nueva Vacuna Mascota

Actores	Empleado
Requisitos asociados	
Descripción	Permite a un usuario registrar una vacuna en el sistema para una mascota.
Precondiciones	El empleado tiene que estar logueado en el sistema.

5. Desarrollo del proyecto

Escenario principal	<ol style="list-style-type: none"> 1. El usuario se encuentra en la pestaña «Vacunas», dentro del perfil de la mascota. A continuación pulsa en «Nuevo». 2. El Sistema abre una ventana emergente con un listado de las vacunas registradas en el sistema correspondientes a la especie de la mascota. Se muestra los siguientes datos: Vacuna, Días. 3. El usuario selecciona aquellas vacunas que se administrarán a la mascota y pulsa «Guardar». 4. El sistema registra las vacunas seleccionadas por el usuario y a continuación registra una nueva cita en el sistema con fecha dependiendo de los días del recordatorio de la vacuna suministrada.
Postcondiciones	Se registra una vacuna a una mascota y se genera una cita con los días de recordatorio de la vacuna suministrada.
Escenario alternativo	
Frecuencia	Baja
Estabilidad	Alta
Comentarios	

Tabla 5.106.: UC-60 Listar Vacunas de la Mascota

Actores	Empleado
Requisitos asociados	
Descripción	Permite a un usuario examinar la lista de vacunas suministradas a una mascota.
Precondiciones	El usuario tiene que estar logueado en el sistema.

5.3. Análisis del sistema

Escenario principal	1. El Sistema muestra un listado de las vacunas suministradas a una Macota con los siguientes datos: Fecha, Hora, Vacuna, Número de días, Descripción, Veterinario.
Postcondiciones	Se muestra un listado de las vacunas.
Escenario alternativo	
Frecuencia	Baja
Estabilidad	Alta
Comentarios	

5. Desarrollo del proyecto

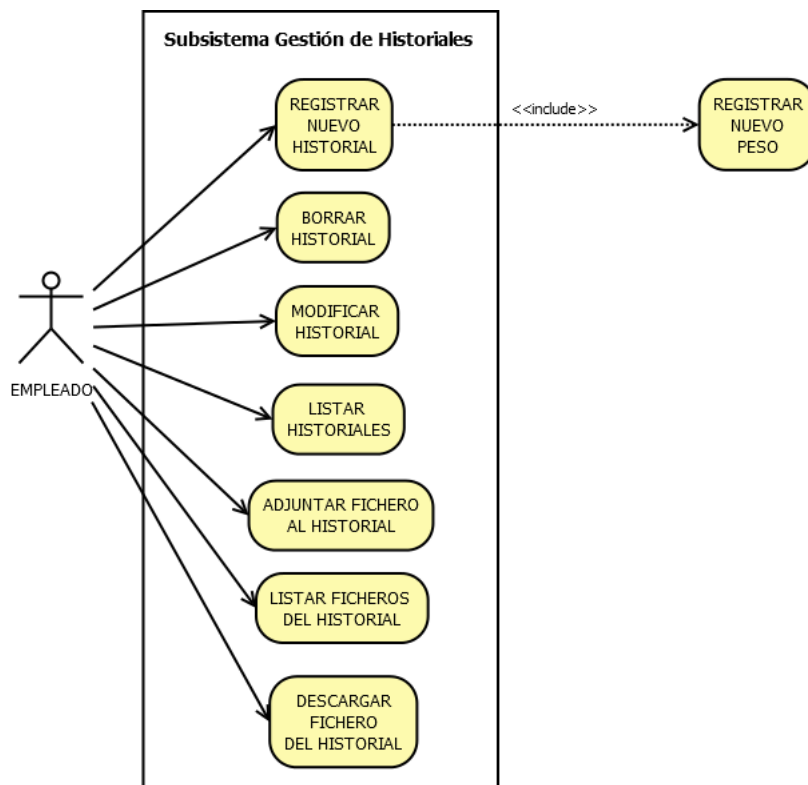


Figura 5.13.: Diagrama Gestión de Historiales

5.3.1.11 Subsistema de Gestión de Historiales

Tabla 5.107.: UC-61 Registrar Nuevo Historial

Actores	Empleado
Requisitos asociados	
Descripción	Permite a un usuario registrar un historial de una mascota en el sistema.
Precondiciones	El empleado tiene que estar logueado en el sistema.

Escenario principal	<ol style="list-style-type: none"> 1. El usuario pulsa en «Nuevo». 2. El Sistema abre una ventana con un historial en blanco, con los siguientes campos: Mascota, Peso, Tipo visita, Anamnesis, Diagnóstico, Tratamiento. Además existe una pestaña «Documentos» con los ficheros relacionados con el historial en curso. 3. El usuario rellena los datos y a continuación pulsa en «Guardar». 4. El sistema añade un nuevo historial correspondiente a la mascota con los datos aportados por el usuario.
Postcondiciones	
Escenario alternativo	
Frecuencia	Alta
Estabilidad	Alta
Comentarios	

5. Desarrollo del proyecto

Tabla 5.108.: UC-62 Borrar Historial

Actores	Empleado
Requisitos asociados	
Descripción	Permite a un usuario borrar un historial de una mascota.
Precondiciones	El usuario tiene que estar logueado en el sistema.
Escenario principal	<ol style="list-style-type: none">1. El usuario selecciona un historial de la lista.2. El Sistema habilita el botón «Eliminar».3. El usuario pulsa sobre el botón «Eliminar».4. El Sistema borra el historial.
Postcondiciones	Se elimina el historial del sistema.
Escenario alternativo	
Frecuencia	Baja
Estabilidad	Alta
Comentarios	

Tabla 5.109.: UC-63 Modificar Historial

Actores	Empleado
Requisitos asociados	
Descripción	Permite a un usuario modificar un Historial.
Precondiciones	El usuario tiene que estar logueado en el sistema.

Escenario principal	<ol style="list-style-type: none"> 1. El usuario selecciona un historial de la pestaña «Historial» y hace doble click. 2. El Sistema abre una ventana con los datos del historial seleccionado. 3. El usuario modifica los datos necesarios y pulsa «Guardar». 4. El Sistema guarda los cambios.
Postcondiciones	
Escenario alternativo	
Frecuencia	Baja
Estabilidad	Alta
Comentarios	

5. Desarrollo del proyecto

Tabla 5.110.: UC-64 Listar Historiales

Actores	Empleado
Requisitos asociados	
Descripción	Permite a un usuario examinar la lista de historiales registrados en el sistema.
Precondiciones	El usuario tiene que estar logueado en el sistema.
Escenario principal	<ol style="list-style-type: none">1. El Sistema muestra un listado de los historiales con los siguientes datos: Fecha, Hora, Tipo visita, Anamnesis, Diagnóstico, Tratamiento, Veterinario.
Postcondiciones	Se muestra un listado de los historiales de una mascota.
Escenario alternativo	
Frecuencia	Alta
Estabilidad	Alta
Comentarios	

Tabla 5.111.: UC-65 Adjuntar Fichero al Historial

Actores	Empleado
Requisitos asociados	
Descripción	Permite a un usuario adjuntar un fichero a un historial de un mascota.
Precondiciones	El empleado tiene que estar logueado en el sistema.

Escenario principal	<ol style="list-style-type: none"> 1. El usuario se encuentra en la pestaña «Documentos», dentro del historial de la mascota. A continuación pulsa en «Añadir Fichero». 2. El Sistema abre una ventana para que el usuario seleccione un fichero de su ordenador. 3. El usuario selecciona el fichero y pulsa «Abrir». 4. El Sistema adjunta el fichero al historial y lo guarda en el sistema.
Postcondiciones	
Escenario alternativo	<ol style="list-style-type: none"> 3a. El usuario selecciona «Cancelar». <ol style="list-style-type: none"> 1. El Sistema cancela el caso de uso y no se añaden ficheros.
Frecuencia	Baja
Estabilidad	Alta
Comentarios	

5. Desarrollo del proyecto

Tabla 5.112.: UC-66 Listar Ficheros del Historial

Actores	Empleado
Requisitos asociados	
Descripción	Permite a un usuario examinar la lista de ficheros relacionados con un historial.
Precondiciones	El usuario tiene que estar logueado en el sistema.
Escenario principal	<ol style="list-style-type: none">1. El Sistema muestra un listado de los ficheros relacionados con un historial.
Postcondiciones	
Escenario alternativo	
Frecuencia	Baja
Estabilidad	Alta
Comentarios	

Tabla 5.113.: UC-67 Descargar Fichero del Historial

Actores	Empleado
Requisitos asociados	
Descripción	Permite a un usuario examinar la lista de vacunas suministradas a una mascota.
Precondiciones	El usuario tiene que estar logueado en el sistema.
Escenario principal	<ol style="list-style-type: none">1. El usuario selecciona un fichero de la lista.2. El Sistema habilita el botón «Descargar».3. El usuario pulsa sobre el botón «Descargar».4. El Sistema abre una ventana de descarga.5. El usuario selecciona dónde desea descargar el fichero en su ordenador, pulsa en «Aceptar» y se inicia la descarga.
Postcondiciones	Se muestra un listado de las ficheros.

5.3. Análisis del sistema

Escenario alternativo	4a. El Sistema detecta que no existe el fichero. 1. El Sistema muestra el error.
Frecuencia	Baja
Estabilidad	Alta
Comentarios	

5. Desarrollo del proyecto

5.3.1.12 Subsistema de Estadísticas

Tabla 5.114.: UC-68 Consulta de Estadísticas

Actores	Empleado
Requisitos asociados	
Descripción	Permite a un usuario consultar las estadísticas generadas por el sistema.
Precondiciones	El usuario tiene que estar logueado en el sistema.
Escenario principal	<ol style="list-style-type: none">1. El usuario desea consultar las estadísticas del año en curso.2. El Sistema genera automáticamente las estadísticas del año en curso y la gráfica correspondiente.
Postcondiciones	
Escenario alternativo	<ol style="list-style-type: none">1a. El usuario desea consultar las estadísticas de otros años, para ello introduce el año que desea.<ol style="list-style-type: none">1. El Sistema muestra las estadísticas del año indicado.
Frecuencia	Media
Estabilidad	Alta
Comentarios	

5.3.2. Modelo conceptual de datos del dominio

A continuación, procederemos a describir gráficamente con el diagrama del modelo conceptual del sistema que se presenta. Debido a la complejidad del mismo, se ha dividido en tres partes el diagrama para que su lectura sea más cómoda: Figura 5.14, Figura 5.15 y Figura 5.16.

Y por último se presenta el diagrama al completo (Véase Figura 5.17) para así tener una visión global del proyecto.

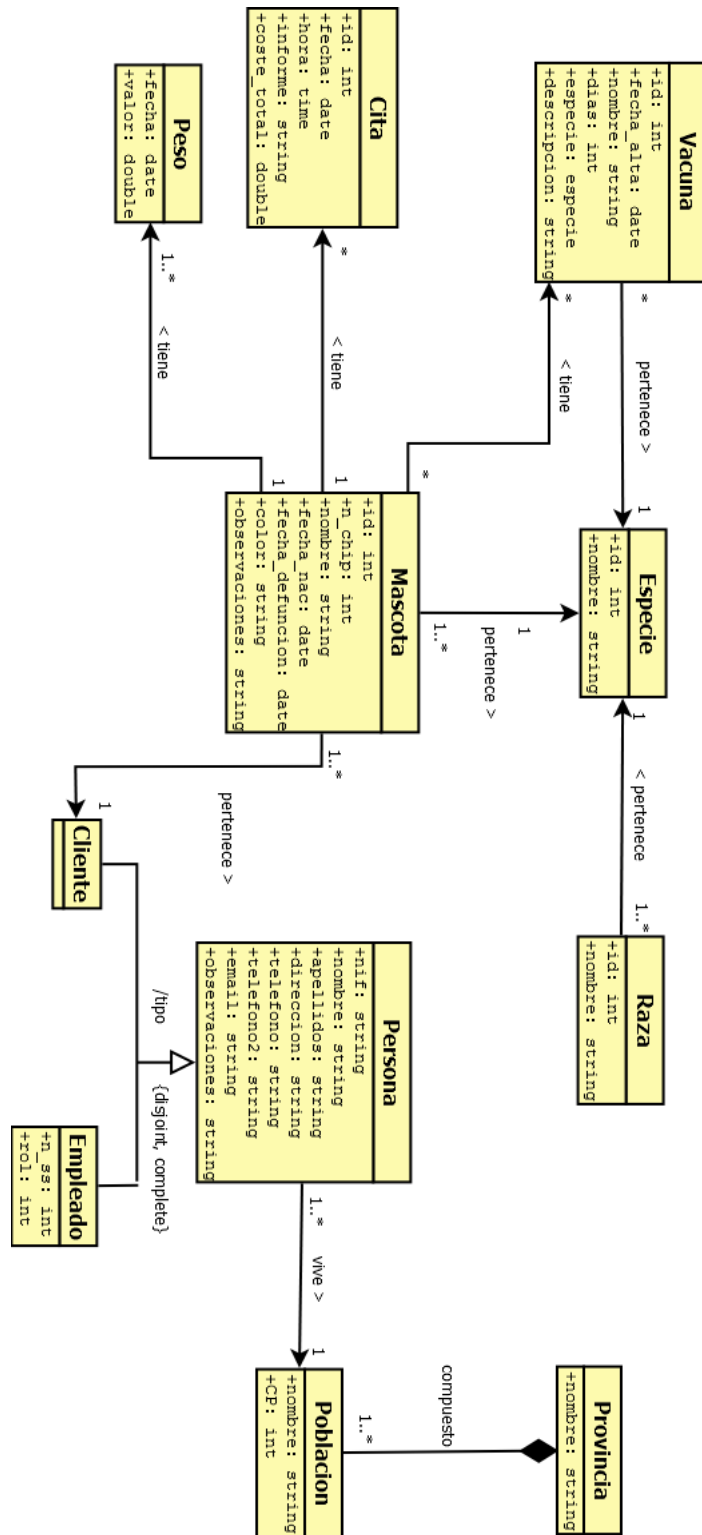


Figura 5.14.: Diagrama de clases conceptuales I

5. Desarrollo del proyecto

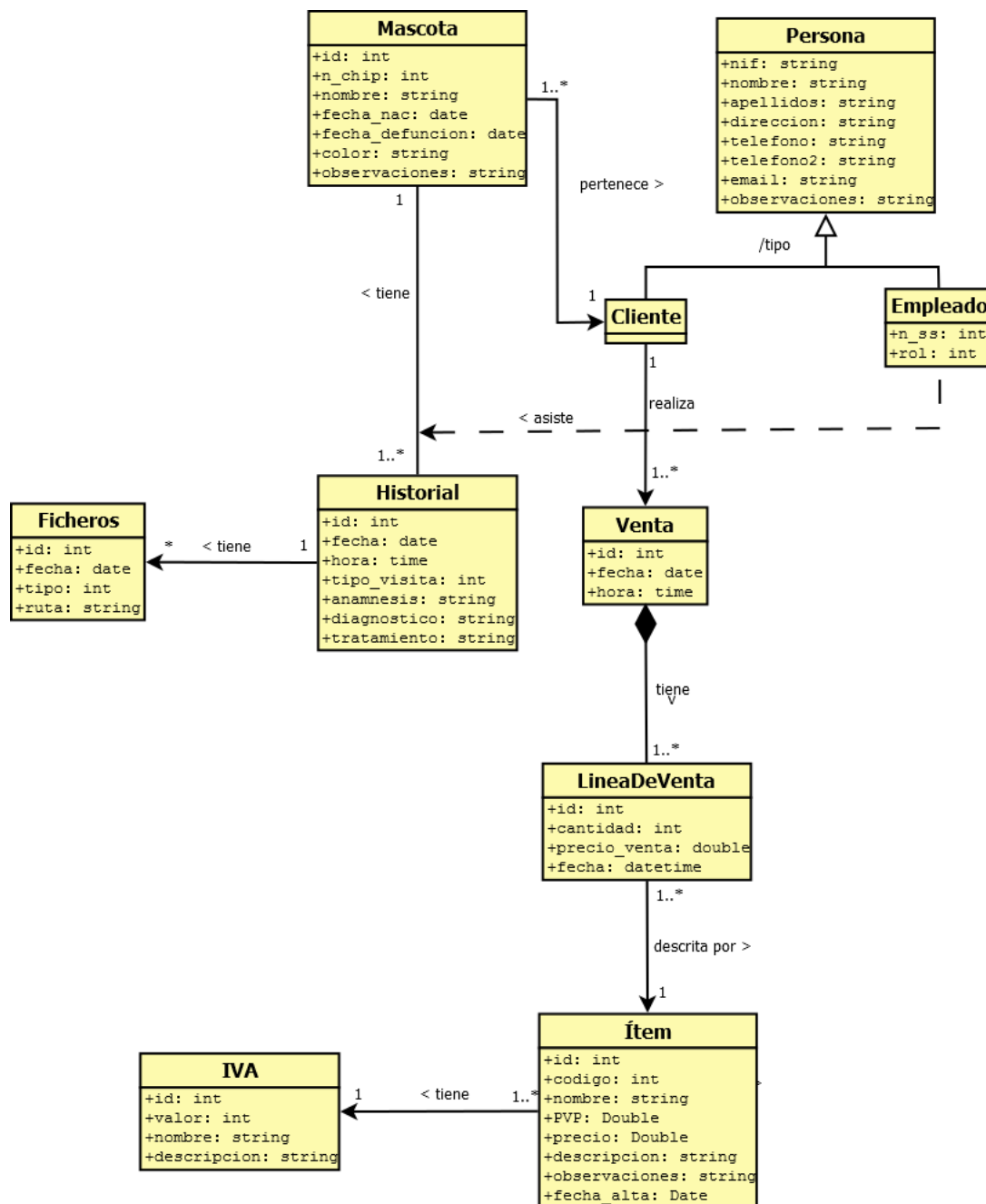


Figura 5.15.: Diagrama de clases conceptuales II

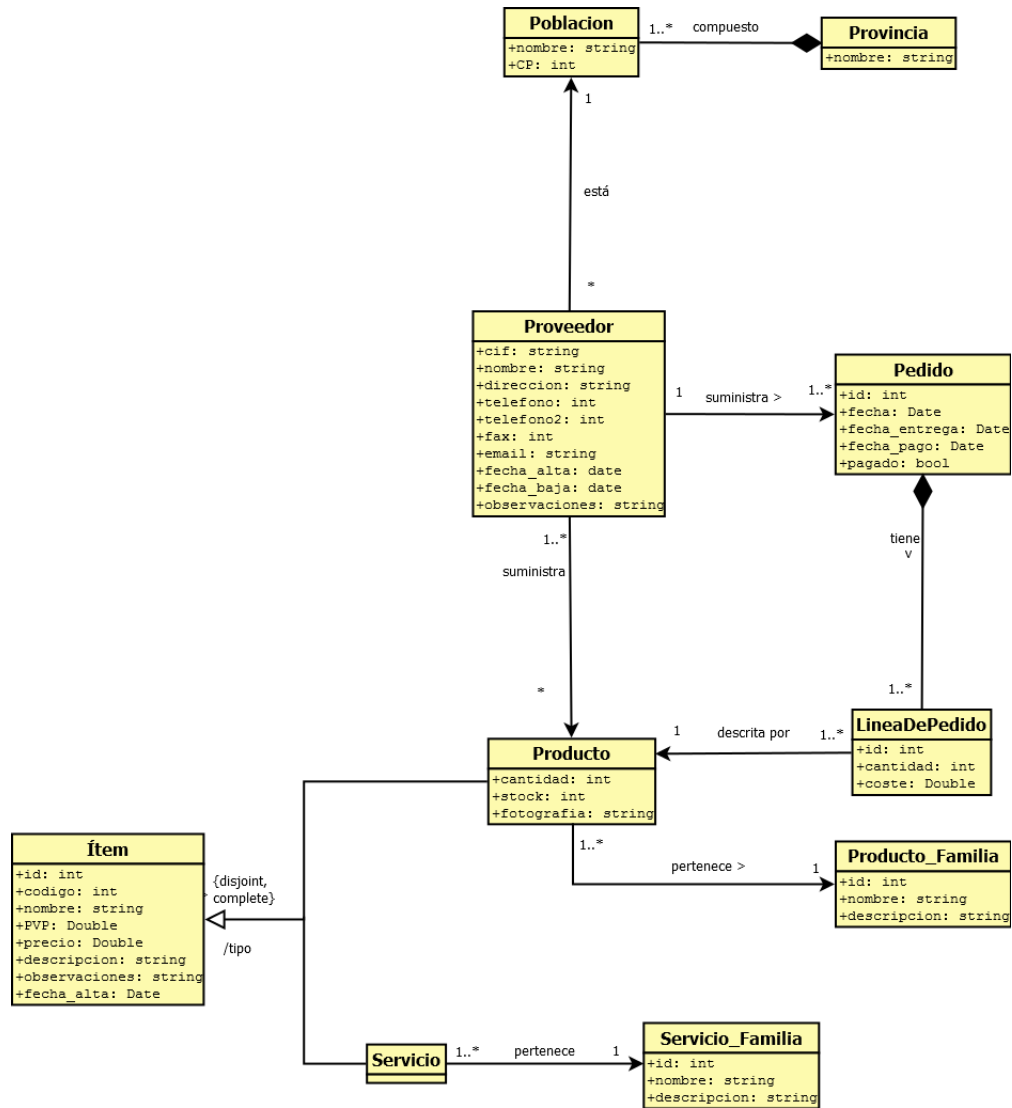


Figura 5.16.: Diagrama de clases conceptuales III

5. Desarrollo del proyecto

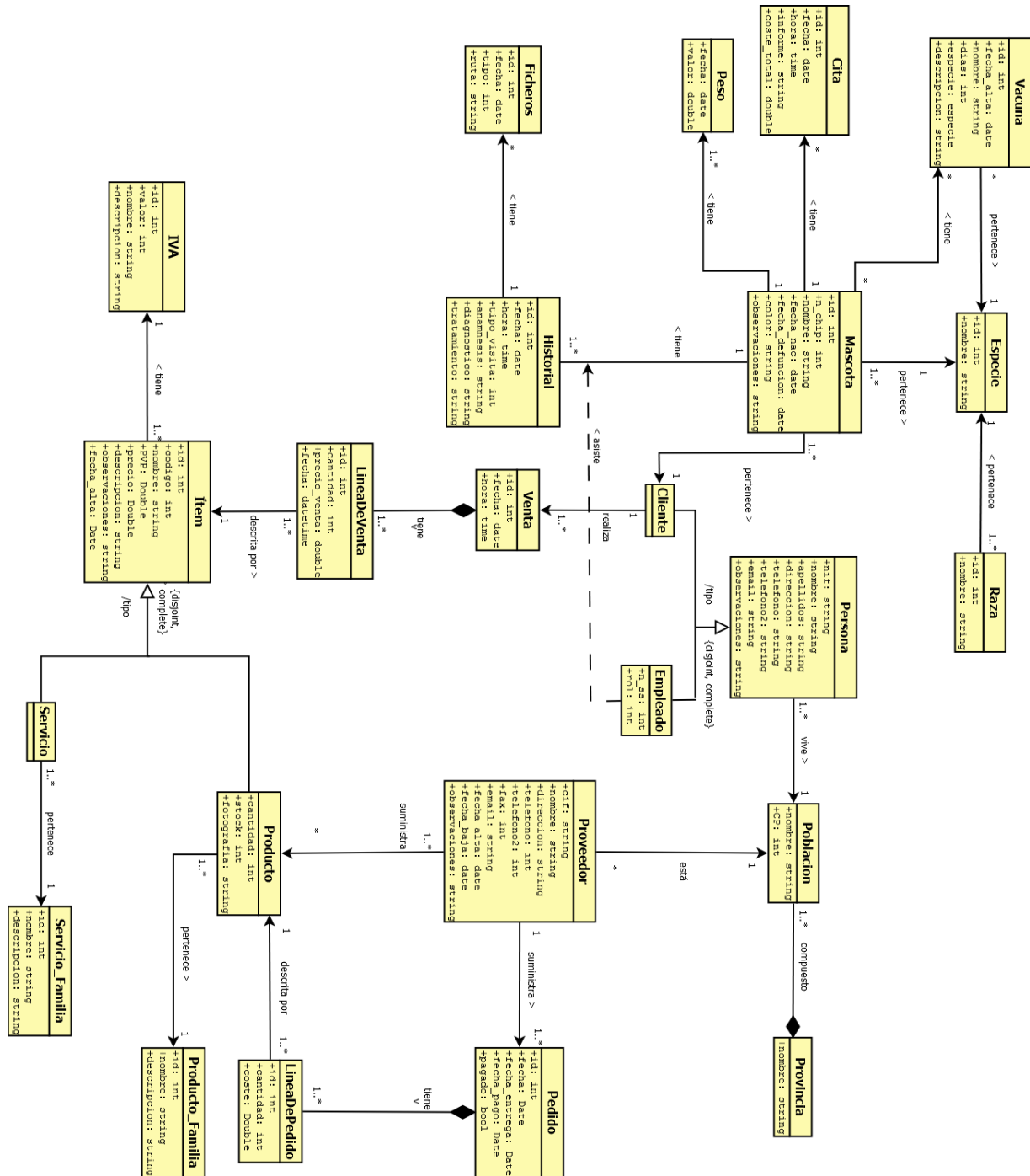


Figura 5.17.: Diagrama de clases conceptuales completo

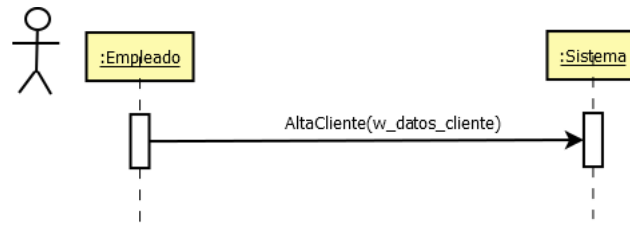


Figura 5.18.: Diagrama Secuencia UC-01 Alta Cliente



Figura 5.19.: Diagrama Secuencia UC-02 Baja Cliente

5.3.3. Diagramas de secuencia

Un diagrama de secuencia muestra una interacción, que representa la secuencia de mensajes entre las instancias de clases, componentes o actores. El tiempo discurre hacia abajo en el diagrama y muestra el flujo de control de un participante a otro.

En esta sección omitiremos algunos diagramas, ya que muchos tienen interacciones similares, únicamente, cambiarían los datos de entrada y de salida.

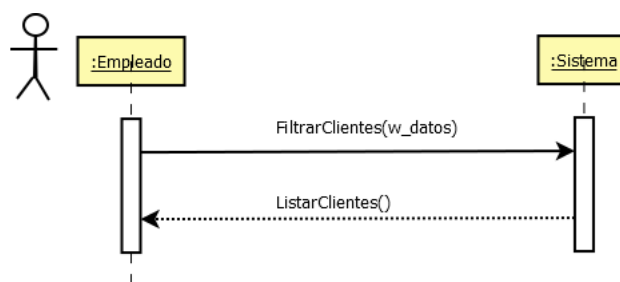


Figura 5.20.: Diagrama Secuencia UC-03 Filtrar Cliente

5. Desarrollo del proyecto

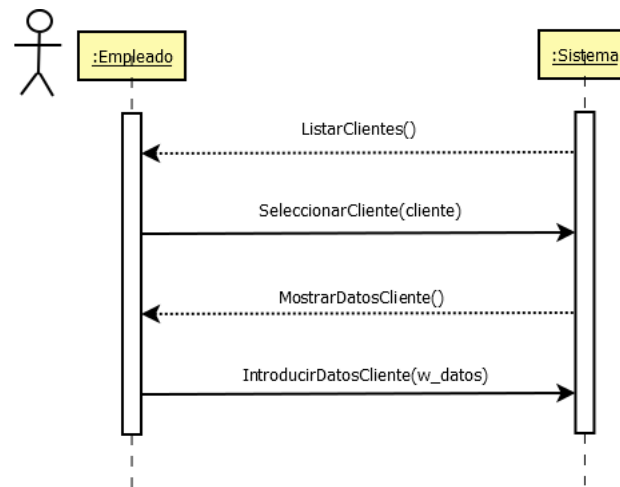


Figura 5.21.: Diagrama Secuencia UC-04 Modificar Cliente



Figura 5.22.: Diagrama Secuencia UC-05 Listar Cliente

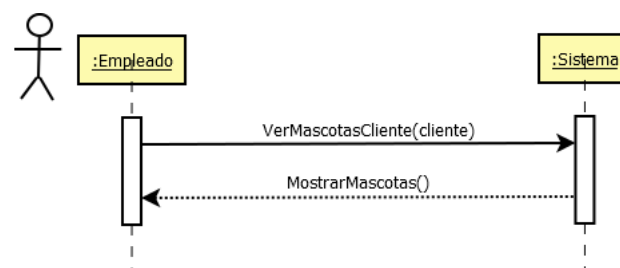


Figura 5.23.: Diagrama Secuencia UC-06 Ver Mascotas Cliente

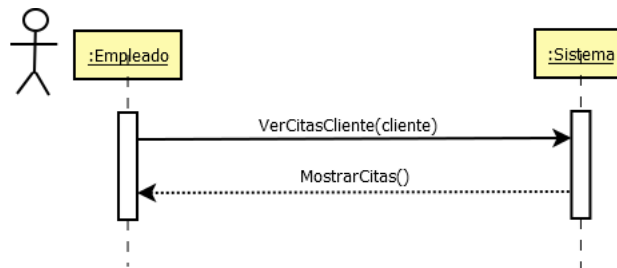


Figura 5.24.: Diagrama Secuencia UC-07 Ver Citas Cliente

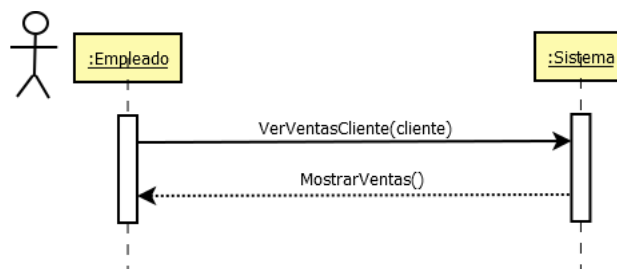


Figura 5.25.: Diagrama Secuencia UC-08 Ver Ventas Cliente

5. Desarrollo del proyecto

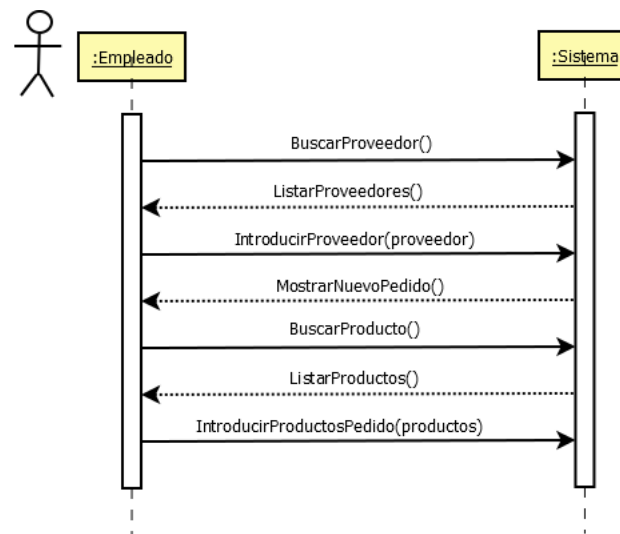


Figura 5.26.: Diagrama Secuencia UC-21 Registrar Nuevo Pedido

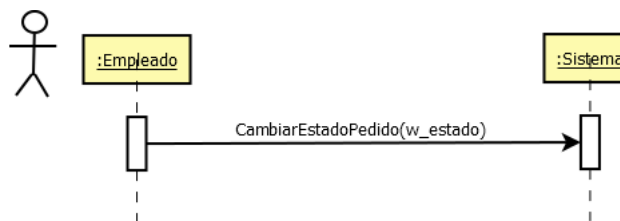


Figura 5.27.: Diagrama Secuencia UC-22 Cambiar Estado Pedido

El diagrama de secuencia del caso de uso UC-41 Generar Albarán (Véase Figura 5.29) es compartido con el caso de uso UC-54 Generar Factura.

El diagrama de secuencia del caso de uso UC-42 Descargar Albarán (Véase Figura 5.30) es compartido con el caso de uso UC-55 Descargar Factura (Véase Tabla ??) y con el caso de uso Descargar Fichero del Historial UC-67 (Véase Tabla ??).

5.4. Diseño del sistema

5.4.1. Arquitectura

Para la realización de este PFC se ha utilizado el patrón arquitectónico MVVM o *Model-View-ViewModel* el cual tiene su origen en Microsoft [1]. MVVM, es una modificación del patrón MVC o *Model-View-Controller*.

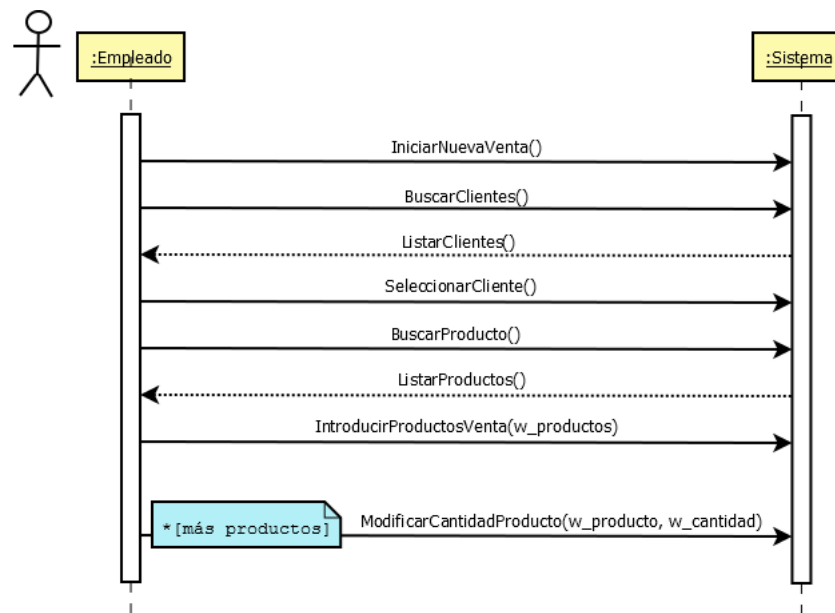


Figura 5.28.: Diagrama Secuencia UC-36 Registrar Nueva Venta

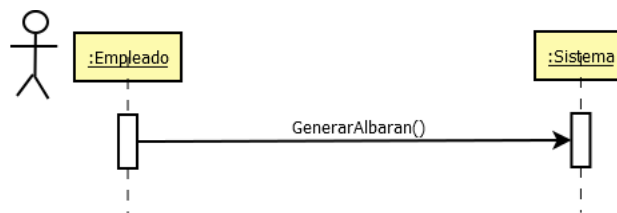


Figura 5.29.: Diagrama Secuencia UC-41 Generar Albarán

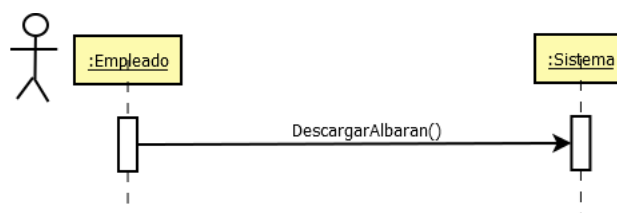


Figura 5.30.: Diagrama Secuencia UC-42 Descargar Albaran

5. Desarrollo del proyecto

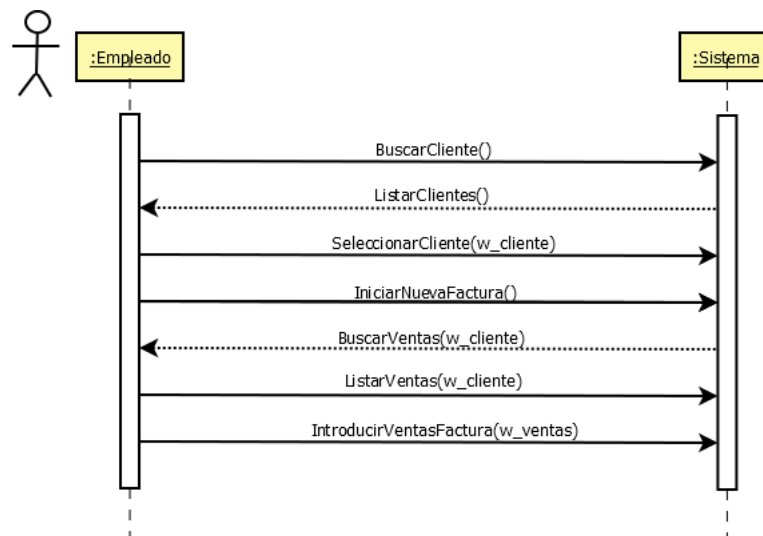


Figura 5.31.: Diagrama Secuencia UC-49 Crear Nueva Factura

MVC Este patrón de diseño divide los datos de la aplicación, la interfaz de usuario y la parte lógica de negocio en tres componentes distintos. En MVC la lógica de negocio es la encargada de actualizar la vista cuando se realiza un cambio.

MVVM Este patrón al ser una modificación del anterior tienen más similitudes que diferencias, ambos comparten las funcionalidades del Model y la Vista. En cambio las principales diferencias que encontramos es que la actualización del View se realiza de forma automática, esto es debido a que tenemos un *ViewModel*, que actúa como un *Controller* especial para el *View*.

Model Son los datos o la lógica de negocio, completamente independiente de la interfaz de usuario (UI), que almacena el estado y realiza el procesamiento del dominio del problema. El *Model* es el código escrito o datos codificados representados por tablas relaciones o XML.

View Es el componente que hace referencia a los elementos visuales, botones, ventanas, tablas, etc.

ViewModel Es el enlace que relaciona el *Model* con el *View*, podríamos decir que es una abstracción del *View*, que contiene el estado del *View* y su comportamiento.

Las principales ventajas de utilizar MVVM son:

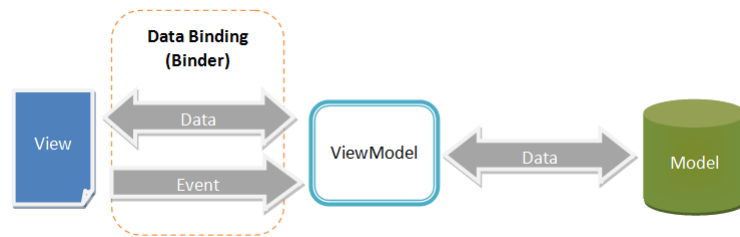


Figura 5.32.: Arquitectura ZK Bind

1. Si las especificaciones del sistema están detalladas y completadas, entonces el desarrollo de la interfaz de usuario, que sería el *View*, se podría hacer al mismo tiempo que la implementación del sistema, así que los diseñadores de la interfaz y programadores trabajarían simultáneamente, disminuyendo los tiempos de desarrollo.
2. Bajo acoplamiento con el *View*, con ello conseguimos que el diseño de la interfaz de usuario puede ser fácilmente cambiada, sin modificar el *ViewModel*, siempre y cuando las especificaciones del sistema no cambien.
3. Alta reusabilidad, gracias a que es más fácil diseñar distintas vistas para distintos dispositivos, usando el mismo *ViewModel*.
4. Facilidad para realizar pruebas, ya que se puede probar el sistema sin que la vista esté operativa.

Como el *ViewModel* no contiene referencias a los componentes de la interfaz de usuario, es necesario un mecanismo que sincronice los datos entre el *View* y el *ViewModel*. Además, este mecanismo tiene que ser capaz de aceptar las peticiones de usuario de la capa *View* y servir de puente de las peticiones de la capa *ViewModel*. Este mecanismo, es llamado *ZK Bind* en el *framework ZK 6*. En resumen, el *binder* es el responsable de la comunicación entre el *View* y el *ViewModel*. (Véase Figura 5.32)

5. Desarrollo del proyecto

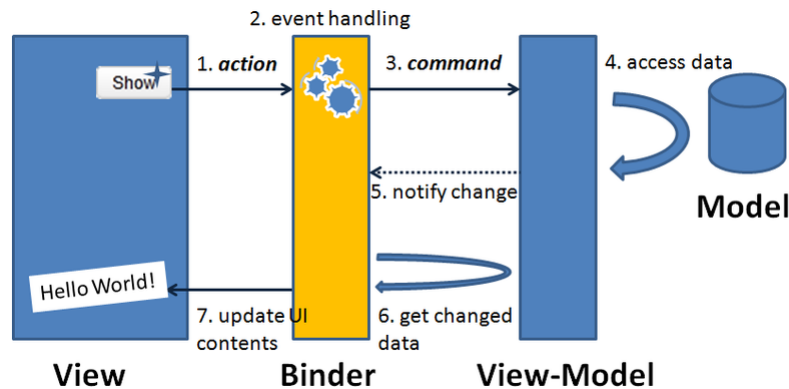


Figura 5.33.: Arquitectura MVVM

Veamos a continuación, como funciona el flujo de operaciones del MVVM con un ejemplo. (Véase Figura 5.33)

1. Un usuario presiona un botón en la pantalla.
2. El evento correspondiente enciende el *binder*.
3. El *binder* encuentra la acción lógica correspondiente en el *ViewModel* y lo ejecuta.
4. La acción lógica accede a los datos desde el *Model* y actualiza las propiedades del correspondiente *ViewModel*.
5. *ViewModel* notifica al *binder* que alguna propiedad ha sido modificada.
6. Por el hecho de haber sido modificadas las propiedades, el *binder* carga los datos desde el *ViewModel*.
7. Entonces el *binder* actualiza el correspondiente componente de la interfaz de usuario para que muestre los cambios al usuario.

5.4.2. Diseño conceptual de la base de datos

A continuación realizaremos el diseño conceptual de la base de datos, donde detallaremos las entidades y sus atributos, con las relaciones que interactúan.

En la Tabla 5.115 podemos observar cada una de las entidades y el tipo de entidad. En la Figura 5.4.2 podemos ver el diagrama Entidad-Relación de nuestra base de datos.

Tipos de relaciones

Pertenece: Relación de pertenencia entre Raza-Especie, Mascota-Raza, Mascota-Cliente, Venta-Cliente, Producto-ProductoFamilia, Servicio-ServicioFamilia.

Tiene: Relación existente entre las entidades Mascota-Peso, Mascota-Cita, Mascota-Vacuna, Mascota-Historial, Historial-Fichero, Venta-Factura, Pedido-PedidoLinea, Venta-VentaLinea, Ítem-IVA.

Vive: Relación existente entre las entidades Cliente-Población.

Está: Relación existente entre las entidades Proveedor-Población.

Describe: Relación existente entre las entidades VentaLinea-Ítem, PedidoLinea-Producto.

Suministra: Relación existente entre las entidades Proveedor-Producto y Proveedor-Pedido.

Tabla 5.115.: Tabla de Entidades

Entidad	Atributos	Tipo
Cita	Id, Fecha, Hora, Informe, Tipo, Estado	Fuerte
Cliente	NIF, Nombre, Apellidos, Direccion, Telefono, Telefono2, Email, Fecha alta, Código postal	Fuerte
Especie	Id, Especie	Fuerte
Factura	Id, Numero, Factura, Fecha	Fuerte
Ficheros	Id, Tipo, Ruta, Fecha	Fuerte
Historial	Id, Fecha, Tipo visita, Anamnesis, Diagnóstico, Tratamiento	Fuerte
IVA	Id, Valor, Nombre, Descripción	Fuerte
Mascota	Chip, Nombre, Sexo, Fecha nacimiento, Fecha defunción, Peso, Altura, Observaciones, Pelo, Fecha alta, Fecha baja	Fuerte
Pedido	Id, Fecha, Fecha entrega, Fecha pago, Pagado	Fuerte
Pedido_linea	Id, Cantidad, Coste, Fecha	Fuerte
Peso	Valor, Fecha	Fuerte

5. Desarrollo del proyecto

Poblacion	Id, Población	Fuerte
Producto	Código, Nombre, PVP, IVA, Precio, Stock, Descripción, Fotografía, Observaciones, Fecha alta	Fuerte
Producto_familia	Nombre, Descripción, Tratamiento	Fuerte
Proveedor	CIF, Nombre, Dirección, Teléfono, Teléfono2, Fax, Email, Fecha alta, Contacto	Fuerte
Provincia	Id, Provincia	Fuerte
Raza	Id, Raza	Fuerte
Servicio	Código, Servicio, Descripción, Precio	Fuerte
Servicio_familia	Nombre, Descripción	Fuerte
Usuario	User, Contraseña, Tipo, Fecha alta, Nombre, Apellidos, NIF, Dirección, Teléfono, Móvil, Email, NSS	Fuerte
Vacuna	Nombre, Descripción, Días, Fecha	Fuerte
Venta	Id, Fecha, Albarán, Facturado	Fuerte
Venta_linea	Cantidad, Fecha, Tipo, PVP, IVA	Fuerte

A continuación veamos con más detalle cada una de las entidades:

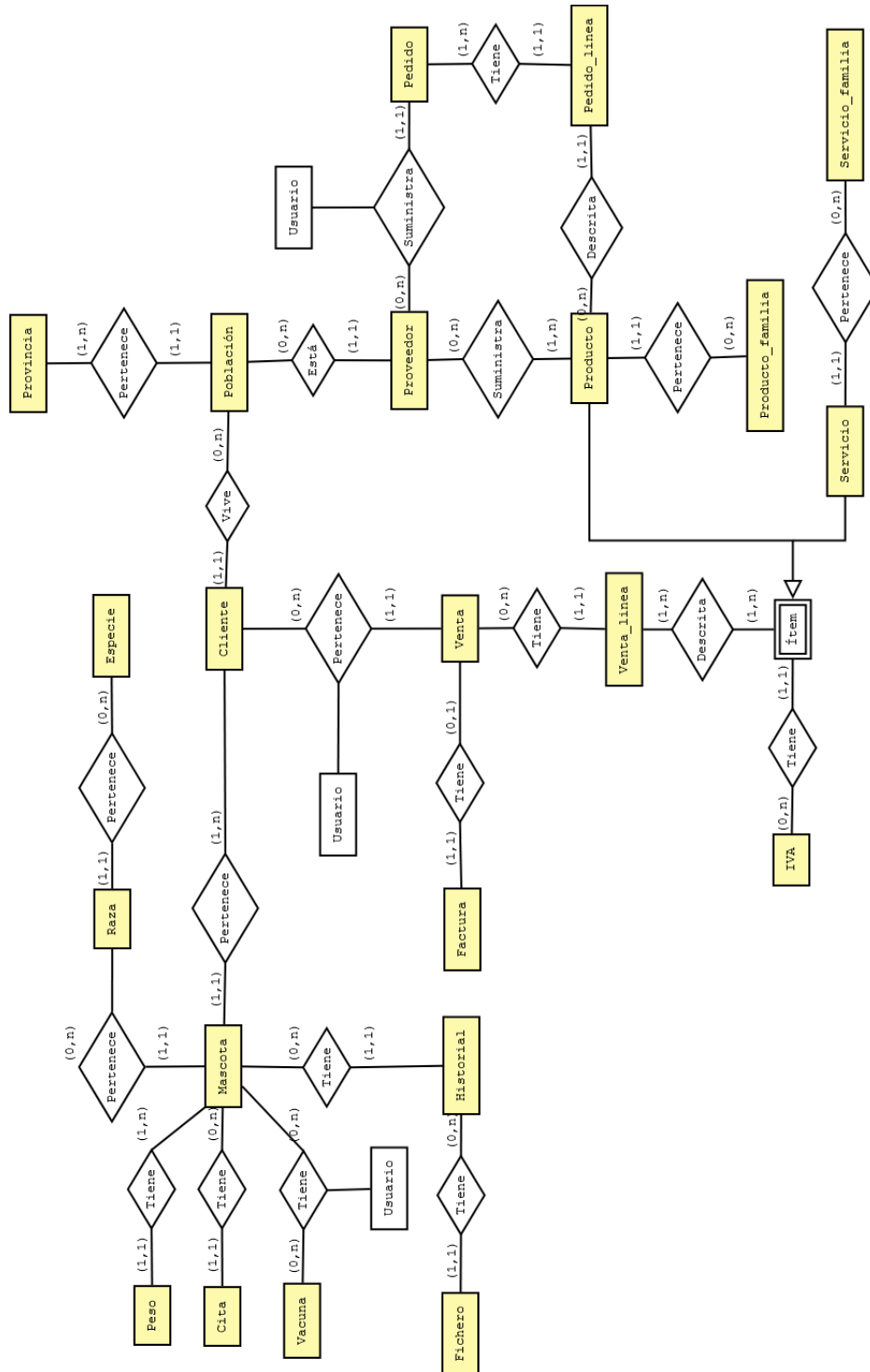


Figura 5.34.: Diagrama Entidad-Relación

Tabla 5.116.: Entidad Cita

Atributo	Descripción	Tipo de atributo	Tipo de dato	Nulo	CP
Id	Identificador	Simple	INT(11)	No	Sí
Fecha	Fecha de la cita.	Compuesto	DATE	No	No
Hora	Hora de la cita.	Compuesto	TIME	No	No
Informe	Información extra sobre la cita.	Compuesto	VARCHAR(500)	Sí	No
Tipo	Tipo de cita.	Simple	INT(1)	Sí	No
Estado	Estado de la cita: Pendiente, Acudido, Avisado, Cancelado.	Simple	INT(1)	No	No

Tabla 5.118.: Entidad Cliente

Atributo	Descripción	Tipo de atributo	Tipo de dato	Nulo	CP
NIF	Número de identificación fiscal.	Compuesto	VARCHAR(9)	No	Sí
Nombre	Nombre del cliente.	Compuesto	VARCHAR(150)	No	No
Apellidos	Apellidos del cliente.	Compuesto	VARCHAR(150)	No	No
Dirección	Dirección del domicilio del cliente.	Compuesto	VARCHAR(250)	No	No
Teléfono	Número de teléfono del cliente.	Simple	INT(9)	No	No
Teléfono 2	Segundo número de teléfono del cliente.	Simple	INT(9)	Sí	No
Email	Dirección de correo electrónico del cliente.	Compuesto	VARCHAR(200)	Sí	No
Fecha_alta	Fecha de alta en el sistema.	Compuesto	TIMESTAMP	No	No
Código postal	Código postal del domicilio del cliente.	Simple	INT(5)	Sí	No

Tabla 5.120.: Entidad Especie

Atributo	Descripción	Tipo de atributo	Tipo de dato	Nulo	CP
Id	Identificador de la especie.	Simple	INT(11)	No	Sí
Especie	Nombre de la especie.	Compuesto	VARCHAR(200)	No	No

Tabla 5.122.: Entidad Factura

Atributo	Descripción	Tipo de atributo	Tipo de dato	Nulo	CP
Id	Identificador de la factura.	Simple	INT(11)	No	Sí
Número	Número de factura.	Simple	INT(11)	No	No
Factura	Dirección de la ruta de la factura.	Compuesto	VARCHAR(300)	No	No
Fecha	Fecha de creación de la factura.	Compuesto	TIMESTAMP	No	No

Tabla 5.124.: Entidad Ficheros

Atributo	Descripción	Tipo de atributo	Tipo de dato	Nulo	CP
Id	Identificador del fichero.	Simple	INT(11)	No	Sí
Tipo	Tipo de fichero al que va relacionado.	Simple	INT(11)	No	No
Ruta	Dirección de la ruta del fichero.	Compuesto	VARCHAR(300)	No	No
Fecha	Fecha de creación del fichero.	Compuesto	TIMESTAMP	No	No

Tabla 5.126.: Entidad Historial

Atributo	Descripción	Tipo de atributo	Tipo de dato	Nulo	CP
Id	Identificador del fichero.	Simple	INT(11)	No	Sí
Fecha	Fecha de creación del historial.	Compuesto	TIMESTAMP	No	No
Tipo_visita	Tipo de visita del historial.	Simple	INT(1)	No	No
Anamnesis	Campo anamnesis.	Compuesto	VARCHAR(500)	Sí	No
Diagnóstico	Campo diagnóstico.	Compuesto	VARCHAR(500)	Sí	No
Tratamiento	Campo tratamiento.	Compuesto	VARCHAR(500)	Sí	No

Tabla 5.128.: Entidad IVA

Atributo	Descripción	Tipo de atributo	Tipo de dato	Nulo	CP
Valor	Valor en % del IVA.	Simple	INT(11)	No	Sí
Nombre	Nombre asociado al IVA.	Compuesto	VARCHAR(200)	No	No
Descripción	Breve descripción del tipo de IVA.	Compuesto	VARCHAR(250)	Sí	No

Tabla 5.130.: Entidad Mascota

Atributo	Descripción	Tipo de atributo	Tipo de dato	Nulo	CP
Id	Número de identificador.	Simple	INT(11)	No	Sí
Chip	Número de microchip de la mascota.	Compuesto	VARCHAR(20)	Sí	No
Nombre	Nombre de la mascota.	Compuesto	VARCHAR(50)	No	No
Sexo	Sexo de la mascota.	Compuesto	VARCHAR(6)	No	No
Fecha nacimiento	Fecha de nacimiento de la mascota.	Compuesto	DATE	No	No
Fecha defunción	Fecha de defunción de la mascota.	Compuesto	DATE	Sí	No
Altura	Altura en centímetros de la mascota.	Simple	FLOAT	No	No
Observaciones	Observaciones sobre la mascota.	Compuesto	VARCHAR(250)	Sí	No
Pelo	Tipo de pelo de la mascota.	Compuesto	VARCHAR(20)	Sí	No
Fecha alta	Fecha de alta en el sistema.	Compuesto	TIMESTAMP	Sí	No
Fecha baja	Fecha de baja en el sistema.	Compuesto	DATETIME	No	No

5. Desarrollo del proyecto

Atributo	Descripción	Tipo de atributo	Tipo de dato	Nulo	CP
Id	Identificador de la línea de pedido.	Simple	INT(11)	No	Sí
Cantidad	Cantidad del artículo a pedir.	Simple	INT(11)	No	No
Coste	Coste del artículo.	Simple	FLOAT	No	No
Fecha	Fecha de la línea de pedido.	Compuesto	TIMESTAMP	No	No

Tabla 5.133.: Entidad Pedido Línea

Tabla 5.134.: Entidad Peso

Atributo	Descripción	Tipo de atributo	Tipo de dato	Nulo	CP
Id	Identificador del peso.	Simple	INT(11)	No	Sí
Valor	Valor del peso.	Simple	FLOAT	No	No
Fecha	Fecha del peso.	Compuesto	TIMESTAMP	No	No

Tabla 5.136.: Entidad Población

Atributo	Descripción	Tipo de atributo	Tipo de dato	Nulo	CP
Id	Identificador de la población.	Simple	INT(11)	No	Sí
Población	Nombre de la población.	Compuesto	VARCHAR(200)	No	No

Tabla 5.138.: Entidad Producto

Atributo	Descripción	Tipo de atributo	Tipo de dato	Nulo	CP
Código	Código del producto.	Compuesto	VARCHAR(11)	No	Sí
Nombre	Nombre del producto.	Compuesto	VARCHAR(20)	No	No
PVP	Precio de venta al público.	Simple	FLOAT	No	No
Precio	Precio de compra al proveedor.	Simple	FLOAT	No	No
Stock	Stock disponible de dicho producto.	Simple	INT(11)	No	No
Descripción	Breve descripción del producto.	Compuesto	VARCHAR(250)	Sí	No
Fotografía	Fotografía del producto.	Compuesto	VARCHAR(250)	Sí	No
Observaciones	Observaciones sobre el producto.	Compuesto	VARCHAR(250)	Sí	No
Fecha alta	Fecha de alta en el sistema.	Compuesto	TIMESTAMP	No	No

Tabla 5.140.: Entidad Producto Familia

Atributo	Descripción	Tipo de atributo	Tipo de dato	Nulo	CP
Id	Identificador de la familia producto.	Simple	INT(11)	No	Sí
Nombre	Nombre de la familia producto.	Compuesto	VARCHAR(200)	No	No
Descripción	Breve descripción sobre la familia producto.	Compuesto	VARCHAR(500)	Sí	No
Tratamiento	Tipo tratamiento.	Simple	TINYINT(1)	No	No

Tabla 5.142.: Entidad Proveedor

Atributo	Descripción	Tipo de atributo	Tipo de dato	Nulo	CP
CIF	CIF del proveedor.	Compuesto	(9)	No	Sí
Nombre	Nombre del proveedor.	Compuesto	VARCHAR(200)	No	No
Dirección	Dirección del proveedor.	Compuesto	VARCHAR(200)	No	No
Teléfono	Teléfono principal del proveedor.	Simple	INT(9)	No	No
Teléfono 2	Teléfono secundario del proveedor.	Simple	INT(9)	Sí	No
Fax	Número de fax del proveedor.	Simple	INT(9)	No	No
Email	Dirección de correo electrónico.	Compuesto	VARCHAR(100)	No	No
Contacto	Persona de contacto.	Compuesto	VARCHAR(250)	Sí	No
Fecha alta	Fecha de alta en el sistema.	Compuesto	TIMESTAMP	No	No

Tabla 5.144.: Entidad Provincia

Atributo	Descripción	Tipo de atributo	Tipo de dato	Nulo	CP
Id	Identificador de la provincia.	Simple	INT(11)	No	Sí
Provincia	Nombre de la provincia.	Compuesto	VARCHAR(200)	No	No

Tabla 5.146.: Entidad Raza

Atributo	Descripción	Tipo de atributo	Tipo de dato	Nulo	CP
Id	Identificador de la raza.	Simple	INT(11)	No	Sí
Raza	Nombre de la raza.	Compuesto	VARCHAR(200)	No	No

Tabla 5.148.: Entidad Servicio

Atributo	Descripción	Tipo de atributo	Tipo de dato	Nulo	CP
Código	Código identificativo del servicio	Compuesto	VARCHAR(9)	No	Sí
Servicio	Nombre del servicio.	Compuesto	VARCHAR(200)	No	No
Descripción	Breve descripción del servicio.	Compuesto	VARCHAR(500)	Sí	No
Precio	Precio del servicio.	Simple	FLOAT	No	No

Tabla 5.150.: Entidad Servicio Familia

Atributo	Descripción	Tipo de atributo	Tipo de dato	Nulo	CP
Id	Identificador del servicio.	Simple	INT(11)	No	Sí
Nombre	Nombre de la familia servicio.	Compuesto	VARCHAR(200)	No	No
Descripción	Breve descripción de la familia servicio.	Compuesto	VARCHAR(200)	Sí	No

Tabla 5.152.: Entidad Usuario

Atributo	Descripción	Tipo de atributo	Tipo de dato	Nulo	CP
User	Nombre de usuario.	Compuesto	VARCHAR(20)	No	Sí
Password	Contraseña del usuario.	Compuesto	VARCHAR(50)	No	No
Tipo	Tipo de usuario.	Simple	INT(1)	No	No
Nombre	Nombre del usuario.	Compuesto	VARCHAR(100)	No	No
Apellidos	Apellidos del usuario.	Compuesto	VARCHAR(100)	No	No
NIF	Número de identificación fiscal.	Compuesto	VARCHAR(10)	No	No
Dirección	Dirección del domicilio del usuario.	Compuesto	VARCHAR(100)	No	No
Teléfono	Número de teléfono del usuario.	Simple	INT(9)	Sí	No
Teléfono 2	Segundo número de teléfono del usuario.	Simple	INT(9)	Sí	No
Email	Dirección de correo electrónico del usuario.	Compuesto	VARCHAR(200)	Sí	No
NSS	Número de la Seguridad Social.	Compuesto	VARCHAR(9)	Sí	No
Fecha_alta	Fecha de alta en el sistema.	Compuesto	TIMESTAMP	No	No

Tabla 5.154.: Entidad Vacuna

Atributo	Descripción	Tipo de atributo	Tipo de dato	Nulo	CP
Id	Identificador de la vacuna	Simple	INT(11)	No	Sí
Nombre	Nombre de la vacuna.	Compuesto	VARCHAR(255)	No	No
Descripción	Breve descripción de la vacuna.	Compuesto	VARCHAR(500)	Sí	No
Días	Número de días para el recordatorio de la vacuna.	Simple	INT(4)	No	No
Fecha	Fecha de alta en el sistema.	Compuesto	TIMESTAMP	No	No

Tabla 5.156.: Entidad Venta

Atributo	Descripción	Tipo de atributo	Tipo de dato	Nulo	CP
Id	Número identificativo de la venta.	Simple	INT(11)	No	Sí
Fecha	Fecha de la venta.	Compuesto	TIMESTAMP	No	No
Albarán	Dirección de la ruta del albarán.	Compuesto	VARCHAR(255)	Sí	No
Facturado	Campo para comprobar si la venta ha sido facturada.	Simple	TINYINT(1)	No	No

Tabla 5.158.: Entidad Venta Línea

Atributo	Descripción	Tipo de atributo	Tipo de dato	Nulo	CP
Id	Número identificativo de la línea de venta.	Simple	INT(11)	No	Sí
Cantidad	Cantidad del artículo para esa línea de venta.	Simple	INT(11)	No	No
Fecha	Fecha de la línea de venta.	Compuesto	TIMESTAMP	No	No
Tipo	Tipo de artículo: 1)Producto 2)Servicio.	Simple	TINYINT(1)	No	No
PVP	PVP del momento de la venta.	Simple	FLOAT	No	No
IVA	IVA del momento de la venta.	Simple	INT(2)	No	No

5.4.3. Diseño lógico de la base de datos

Normalización:

- **1FN:** Las relaciones que hemos descrito están en 1FN, ya que para toda intersección de fila y columna tendremos un único valor. También los atributos de cada una de las tablas son atómicos, aunque tenemos atributos *Teléfono*, estos atributos son atómicos ya que contienen su propio campo.
- **2FN:** Una relación está en 2FN si está en 1FN y todo atributo que no es clave primaria depende completamente de la clave primaria. Las relaciones que aquí se describen están en 2FN.
- **3FN:** Una relación está en 3FN si está en 2FN y ningún atributo que no es clave primaria depende transitivamente de la clave primaria. Las relaciones descritas se encuentran en 3FN.
- **FNBC:** Las relaciones se encuentran en FNBC, ya que cumplen con la 3FN y en todas las relaciones no existen dependencias no triviales a una parte de la clave primaria.

El resultado del diseño lógico es el siguiente:

Tabla 5.160.: Tabla Cita Lógica

Atributos	fecha, hora, informe, tipo, estado
Clave primaria	id
Única	
Clave foránea	mascota, cliente, empleado

Tabla 5.161.: Tabla Cliente Lógica

Atributos	nombre, apellidos, direccion, telefono, telefono2, email, fecha_alta, codigopostal
Clave primaria	nif
Única	
Clave foránea	

Tabla 5.162.: Tabla Especie Lógica

Atributos	especie
Clave primaria	id
Única	
Clave foránea	

Tabla 5.163.: Tabla Factura Lógica

Atributos	número, factura, fecha
Clave primaria	id
Única	
Clave foránea	cliente, empleado

Tabla 5.164.: Tabla Ficheros Lógica

Atributos	tipo, ruta, fecha
Clave primaria	id
Única	
Clave foránea	id_externo

Tabla 5.165.: Tabla Historial Lógica

Atributos	fecha, tipo_visita, anamnesis, diagnóstico, tratamiento
Clave primaria	id
Única	
Clave foránea	id_mascota, id_veterinario, id_peso

5. Desarrollo del proyecto

Tabla 5.166.: Tabla IVA Lógica

Atributos	valor, nombre, descripción
Clave primaria	id
Única	
Clave foránea	

Tabla 5.167.: Tabla Mascota Lógica

Atributos	chip, nombre, sexo, fecha_nac, fecha_def, altura, observaciones
Clave primaria	id
Única	chip
Clave foránea	peso, raza, cliente

Tabla 5.168.: Tabla Mascota-Vacuna Lógica

Atributos	fecha
Clave primaria	id
Única	
Clave foránea	mascota, vacuna, veterinario

Tabla 5.169.: Tabla Pedido Lógica

Atributos	fecha, fecha_entrega, fecha_pago, pagado
Clave primaria	id
Única	
Clave foránea	proveedor, empleado

Tabla 5.170.: Tabla Pedido Línea Lógica

Atributos	cantidad, coste
Clave primaria	id
Única	
Clave foránea	pedido, producto

Tabla 5.171.: Tabla Peso Lógica

Atributos	valor, fecha
Clave primaria	id
Única	
Clave foránea	mascota,

Tabla 5.172.: Tabla Población Lógica

Atributos	problacion
Clave primaria	id
Única	
Clave foránea	provincia

Tabla 5.173.: Tabla Producto Lógica

Atributos	nombre, pvp, precio, stock, descripción, fotografía, observaciones, fecha__alta
Clave primaria	código
Única	
Clave foránea	id_familia, proveedor, id_iva

5. Desarrollo del proyecto

Tabla 5.174.: Tabla Producto-Familia Lógica

Atributos	nombre, descripción, tratamiento
Clave primaria	id
Única	nombre
Clave foránea	

Tabla 5.175.: Tabla Proveedor Lógica

Atributos	nombre, dirección, teléfono, teléfono2, fax, email, contacto, fecha_alta
Clave primaria	cif
Única	
Clave foránea	población, provincia

Tabla 5.176.: Tabla Provincia Lógica

Atributos	provincia
Clave primaria	id
Única	provincia
Clave foránea	

Tabla 5.177.: Tabla Raza Lógica

Atributos	raza
Clave primaria	id
Única	raza
Clave foránea	especie

Tabla 5.178.: Tabla Servicio Lógica

Atributos	servicio, descripción, precio
Clave primaria	código
Única	
Clave foránea	id_iva, id_familia

Tabla 5.179.: Tabla Servicio-Familia Lógica

Atributos	descripción
Clave primaria	id
Única	nombre
Clave foránea	

Tabla 5.180.: Tabla Usuario Lógica

Atributos	password, tipo, fecha_alta, nombre, apellidos, nif, dirección, teléfono, teléfono2, email, nss
Clave primaria	user
Única	
Clave foránea	población, provincia

Tabla 5.181.: Tabla Vacuna Lógica

Atributos	nombre, descripción, días
Clave primaria	id
Única	
Clave foránea	especie

5. Desarrollo del proyecto

Tabla 5.182.: Tabla Venta Lógica

Atributos	fecha, albarán, facturado
Clave primaria	id
Única	
Clave foránea	cliente, vendedor, veterinario, factura

Tabla 5.183.: Tabla Venta-Línea Lógica

Atributos	cantidad, fecha, tipo, pvp, iva
Clave primaria	id
Única	
Clave foránea	id_venta, producto

5.5. Implementación

Para implementar nuestra aplicación RIA se ha utilizado ZK como tecnología principal. ZK es un *framework* basado en JavaScript, que nos permite obtener una aplicación AJAX sin la necesidad de saber AJAX. ZK incluye su propio lenguaje para la creación de la interfaz de usuario, estamos hablando de ZUML, compuesto de componentes XUL y componentes XHTML. Como lenguaje de programación para la parte lógica se ha utilizado Java, orientado a objetos y como SGBD hemos utilizado MySQL *Community*. Al tratarse nuestro PFC de una aplicación RIA, necesitamos de un servidor web, para este cometido hemos optado por Apache Tomcat 7.0. La creación de documentos en PDF se realiza con la librería iText® [3] y para la generación de gráficas en JavaScript hemos utilizado el componente ZHighCharts [13], diseñado para incluir la librería HighCharts JS [2] en ZK.

Hoy en día para desarrollar aplicaciones complejas, es necesaria las ayudas que incorporan los IDE, en esta ocasión hemos utilizado NetBeans, que con el complemento REM, hemos conseguido un entorno de desarrollo productivo.

5.5.1. Inserción, actualización y eliminación de elementos

A continuación procederemos a explicar cómo se produce la inserción de nuevos elementos en nuestra aplicación, y para ello tomaremos como ejemplo la creación de una nueva especie (Véase Listado 5.1), actualización de una especie (Véase Listado 5.2) y la eliminación de un producto (Véase Listado 5.3):

Listado 5.1: Creación Nueva Especie

```

1 @Command( "add" )
2 @NotifyChange( "events" )
3 public void add() {
4     ...
5     eventDao.insert( this.newEvent ); //Se realiza la
        inserción
6     ...
7     this.newEvent = new Especie();
8 }
```

5. Desarrollo del proyecto

Listado 5.2: Actualización de Especie

```
1 @Command("update")
2 @NotifyChange("events")
3 public void update() {
4     if(!eventDao.update(this.selectedEvent)){
5         MessageBox.show("La especie no se actualizó",
6             "Aviso", MessageBox.OK,
7             MessageBox.EXCLAMATION);
8     }
9     this.selectedEvent = null;
10 }
```

Listado 5.3: Eliminación de Producto

```
1 @Command("delete")
2 @NotifyChange({"events", "productos"})
3 public void delete() {
4     if(this.selectedEvent != null) {
5         eventDao.delete(this.selectedEvent);
6         this.selectedEvent = null;
7     }
8 }
```

5.5.2. Generación de gráficas

A continuación una porción de código, donde se configura unos parámetros básicos del componente `<zhighcharts>`. (Véase Listado 5.4)

Listado 5.4: Generación de Gráficas

```
1 <zhighcharts id="chart "
2     model="@load(vm.grafica2)"
3     type="column"
4     width="900px"
5     height="320px"
6     style="margin-top:10px;
7         font-weight:bold;
8         float:left"
9 />
```


5.5.3. Conexión con la base de datos

En el campo *url* configuraremos la dirección de la la base de datos, en el campo *user* se configura el nombre de usuario y en el campo *pwd* es su respectiva contraseña (Veáse Listado 5.5)

Listado 5.5: Conexión con la Base de Datos

```

1 public Statement getStatement() throws SQLException {
2     Statement stmt;
3     conn = DriverManager.getConnection(url, user, pwd);
4     stmt = conn.createStatement();
5     return stmt;
6 }

```

5.5.4. Consulta a la base de datos

En el Listado 5.6 tenemos un ejemplo del código necesario para realizar una consulta a la base de datos. El campo *consulta* tiene que estar en lenguaje MySQL.

Listado 5.6: Consulta a la Base de Datos

```

1 public List<Especie> findAll(String consulta) {
2     List<Especie> allEvents = new ArrayList<Especie>();
3     try{
4         Statement stmt = ds.getStatement();
5         ResultSet rs = stmt.executeQuery(consulta);
6         Especie esp;
7         while (rs.next()) {
8             esp = new Especie();
9             esp.setId(rs.getInt(1));
10            esp.setEspecie(rs.getString(2));
11            allEvents.add(esp);
12        }
13    } catch (SQLException e) {
14        e.printStackTrace();
15    } finally {
16        ds.close();
17    }
18    return allEvents;
19 }

```

5.5.5. Listado de elementos

A continuación describiremos cómo se funciona el listado de una ventana, por ejemplo el listado de mascotas. Para ello primero, se carga el *ViewModel* con el siguiente código (Véase Listado 5.7),

Listado 5.7: Carga ViewModel

```
1 <window id="mascotalistado "  
2     title="Productos "  
3     border="normal "  
4     apply="org.zkoss.bind.BindComposer "  
5     viewModel="@id('vm')  
6     @init('es.clinica.veterinaria.mascotas.MascotaViewModel  
7         ')" //Especifica el ViewModel  
8     visible="false "  
9 >  
10 ...  
11 </window>
```

En el componente `<listbox>`, será dónde se liste los elementos, que en este caso será un listado de mascotas (Véase Listado 5.8)

Listado 5.8: Listado de mascotas

```
1 <listbox id="box "  
2     ...  
3     model="@bind(vm.mascotas) " //Especifica el listado de  
4         mascotas  
5     selectedItem="@bind(vm.selectedEvent) "  
6     emptyMessage="Ningún ítem encontrado ">
```

Ahora veamos el método *getMascotas()*, que se encuentra en *MascotaViewModel.java* (Véase Listado 5.9), en un primer momento se encuentra vacío y dependiendo de lo que vayamos introduciendo en el filtrado, se irá añadiendo objetos al listado con el método *doSearch()* siempre y cuando concuerde con lo buscado (Véase Listado 5.10).

Listado 5.9: getMascotas()

```
1 private ListModelList<Mascota> mascotas;  
2 ....  
3 public ListModelList<Mascota> getMascotas() {  
4     if(mascotas == null) {  
5         mascotas = new ListModelList<Mascota>();
```

```

6     }
7     return mascotas;
8 }

```

Listado 5.10: doSearch()

```

1 @Command
2 @NotifyChange( "mascotas" )
3 public void doSearch() {
4     mascotas.clear();
5     List<Mascota> allEvents;
6     ...
7     else {
8         allEvents = eventDao.findAll();
9         for (Iterator<Mascota> it = allEvents.iterator(); it.
10             hasNext(); ) {
11             Mascota masc = it.next();
12             ...
13             if ((masc.getChip().toLowerCase().startsWith(filterChip
14                 .toLowerCase())) && ...
15                 (especie.toLowerCase().startsWith(filterEspecie.
16                     toLowerCase())) {
17                 mascotas.add(masc);
18             }
19         }
20     }
21 }

```

5.5.6. Diseño físico de la base de datos

Como sistema de gestión de base de datos se ha optado por MySQL en su versión *Community*.

Listado 5.11: Tabla Cita

```

1 CREATE TABLE IF NOT EXISTS 'zk_cita' (
2     'id' int(11) NOT NULL AUTO_INCREMENT,
3     'fecha' date NOT NULL,
4     'hora' time DEFAULT NULL,
5     'informe' varchar(500) DEFAULT NULL,
6     'tipo' int(1) DEFAULT NULL,
7     'id_mascota' int(11) NOT NULL,

```

5. Desarrollo del proyecto

```
8  'id_cliente' int(11) NOT NULL,  
9  'estado' tinyint(4) NOT NULL DEFAULT '2' COMMENT '0)  
    Cancelado 1)Acudido 2)Pendiente 3)Cita',  
10 'id_empleado' varchar(20) NOT NULL,  
11 PRIMARY KEY ('id'),  
12 CONSTRAINT 'FK_cita_masc' FOREIGN KEY ('id_mascota')  
13 REFERENCES 'zk_mascota' ('id'),  
14 CONSTRAINT 'FK_cita_cli' FOREIGN KEY ('id_cliente')  
15 REFERENCES 'zk_cliente' ('id'),  
16 CONSTRAINT 'FK_cita_emp' FOREIGN KEY ('id_empleado')  
17 REFERENCES 'zk_usuario' ('user')  
18  
19 ) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

Listado 5.12: Tabla Cliente

```
1 CREATE TABLE IF NOT EXISTS 'zk_cliente' (  
2   'id' int(11) NOT NULL AUTO_INCREMENT,  
3   'nombre' varchar(150) NOT NULL,  
4   'apellidos' varchar(150) NOT NULL,  
5   'nif' varchar(10) NOT NULL,  
6   'direccion' varchar(250) NOT NULL,  
7   'ciudad' int(11) NOT NULL,  
8   'provincia' int(11) NOT NULL,  
9   'telefono' int(9) NOT NULL,  
10  'telefono2' int(9) DEFAULT NULL,  
11  'email' varchar(200) DEFAULT NULL,  
12  'fecha_alta' timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,  
13  'codigopostal' int(5) DEFAULT NULL,  
14  PRIMARY KEY ('id'),  
15  UNIQUE KEY 'nif' ('nif'),  
16  CONSTRAINT 'FK_cliente_pob' FOREIGN KEY ('ciudad')  
17  REFERENCES 'zk_poblacion' ('id'),  
18  CONSTRAINT 'FK_cliente_prov' FOREIGN KEY ('provincia')  
19  REFERENCES 'zk_provincia' ('id')  
20 ) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

Listado 5.13: Tabla Especie

```
1 CREATE TABLE IF NOT EXISTS 'zk_especie' (  
2   'id' int(11) NOT NULL AUTO_INCREMENT,  
3   'especie' varchar(200) NOT NULL,
```

```

4 PRIMARY KEY ( 'id ' )
5 ) ENGINE=InnoDB DEFAULT CHARSET=utf8 ;

```

Listado 5.14: Tabla Factura

```

1 CREATE TABLE IF NOT EXISTS 'zk_factura' (
2   'id' int(11) NOT NULL AUTO_INCREMENT,
3   'numero' int(11) NOT NULL,
4   'cliente' int(11) NOT NULL,
5   'empleado' varchar(20) NOT NULL,
6   'factura' varchar(300) NOT NULL,
7   'fecha' timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,
8   PRIMARY KEY ( 'id ' ),
9   CONSTRAINT 'FK_fact_cli' FOREIGN KEY ( 'cliente ' )
10    REFERENCES 'zk_cliente' ( 'id ' ),
11   CONSTRAINT 'FK_fact_emp' FOREIGN KEY ( 'empleado ' )
12    REFERENCES 'zk_usuario' ( 'user ' )
13 ) ENGINE=InnoDB DEFAULT CHARSET=utf8 ;

```

Listado 5.15: Tabla Ficheros

```

1 CREATE TABLE IF NOT EXISTS 'zk_ficheros' (
2   'id' int(11) NOT NULL AUTO_INCREMENT,
3   'id_externo' int(11) NOT NULL,
4   'tipo' smallint(2) NOT NULL COMMENT '1) Historial',
5   'ruta' varchar(250) NOT NULL,
6   'fecha' timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,
7   PRIMARY KEY ( 'id ' )
8 ) ENGINE=InnoDB DEFAULT CHARSET=utf8 ;

```

Listado 5.16: Tabla Historial

```

1 CREATE TABLE IF NOT EXISTS 'zk_historial' (
2   'id' int(11) NOT NULL AUTO_INCREMENT,
3   'id_mascota' int(11) NOT NULL,
4   'id_veterinario' varchar(20) NOT NULL,
5   'fecha' timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,
6   'tipo_visita' int(1) NOT NULL,
7   'id_peso' int(11) NOT NULL,
8   'anamnesis' varchar(500) DEFAULT NULL,
9   'diagnostico' varchar(500) DEFAULT NULL,
10  'tratamiento' varchar(500) DEFAULT NULL,

```

5. Desarrollo del proyecto

```
11 PRIMARY KEY ( 'id' ),
12 CONSTRAINT 'FK_hist_masc' FOREIGN KEY ( 'id_mascota' )
13 REFERENCES 'zk_mascota' ( 'id' ),
14 CONSTRAINT 'FK_hist_emp' FOREIGN KEY ( 'id_veterinario' )
15 REFERENCES 'zk_usuario' ( 'user' ),
16 CONSTRAINT 'FK_hist_peso' FOREIGN KEY ( 'id_peso' )
17 REFERENCES 'zk_peso' ( 'id' )
18 ) ENGINE=InnoDB DEFAULT CHARSET=utf8 ;
```

Listado 5.17: Tabla IVA

```
1 CREATE TABLE IF NOT EXISTS 'zk_iva' (
2   'id' int(11) NOT NULL AUTO_INCREMENT,
3   'valor' int(2) NOT NULL,
4   'nombre' varchar(200) DEFAULT NULL,
5   'descripcion' varchar(250) DEFAULT NULL,
6   PRIMARY KEY ( 'id' )
7 ) ENGINE=InnoDB DEFAULT CHARSET=utf8 ;
```

Listado 5.18: Tabla Mascota

```
1 CREATE TABLE IF NOT EXISTS 'zk_mascota' (
2   'id' int(11) NOT NULL AUTO_INCREMENT,
3   'chip' varchar(20) DEFAULT NULL,
4   'nombre' varchar(50) NOT NULL,
5   'sexo' varchar(6) DEFAULT NULL COMMENT '0)Macho 1)Hembra' ,
6   'fecha_nac' date NOT NULL,
7   'fecha_def' date DEFAULT NULL,
8   'peso' int(11) DEFAULT NULL,
9   'altura' float DEFAULT NULL,
10  'observaciones' varchar(250) DEFAULT NULL,
11  'pelo' varchar(100) DEFAULT NULL COMMENT '1)Corto 2)Duro
12      3)Largo 4)Medio 5)Medio-Largo 6)Propio 7)Rizado' ,
13  'raza' int(11) DEFAULT NULL,
14  'fecha_alta' timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,
15  'fecha_baja' datetime DEFAULT NULL,
16  'id_cliente' int(11) NOT NULL,
17  PRIMARY KEY ( 'id' ),
18  UNIQUE KEY 'chip' ( 'chip' ),
19  CONSTRAINT 'FK_masc_peso' FOREIGN KEY ( 'peso' )
20  REFERENCES 'zk_peso' ( 'id' ),
  CONSTRAINT 'FK_masc_raza' FOREIGN KEY ( 'raza' )
```

```

21 REFERENCES 'zk_raza' ('id'),
22 CONSTRAINT 'FK_masc_cli' FOREIGN KEY ('id_cliente')
23 REFERENCES 'zk_cliente' ('id')
24 ) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

Listado 5.19: Tabla Mascota-Vacuna

```

1 CREATE TABLE IF NOT EXISTS 'zk_mascota_vacuna' (
2   'id' int(11) NOT NULL AUTO_INCREMENT,
3   'mascota' int(11) NOT NULL,
4   'vacuna' int(11) NOT NULL,
5   'fecha' timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,
6   'veterinario' varchar(20) NOT NULL,
7   PRIMARY KEY ('id'),
8   CONSTRAINT 'FK_mv_masc' FOREIGN KEY ('mascota')
9     REFERENCES 'zk_mascota' ('user'),
10  CONSTRAINT 'FK_mv_vac' FOREIGN KEY ('vacuna')
11    REFERENCES 'zk_vacuna' ('id'),
12  CONSTRAINT 'FK_mv_emp' FOREIGN KEY ('veterinario')
13    REFERENCES 'zk_usuario' ('user')
14
15 ) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

Listado 5.20: Tabla Pedido

```

1 CREATE TABLE IF NOT EXISTS 'zk_pedido' (
2   'id' int(11) NOT NULL AUTO_INCREMENT,
3   'fecha' timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,
4   'fecha_entrega' timestamp NULL DEFAULT NULL,
5   'fecha_pago' timestamp NULL DEFAULT NULL,
6   'pagado' tinyint(1) NOT NULL,
7   'id_proveedor' varchar(9) NOT NULL,
8   'id_empleado' varchar(20) NOT NULL,
9   PRIMARY KEY ('id'),
10  CONSTRAINT 'FK_ped_provee' FOREIGN KEY ('id_proveedor')
11    REFERENCES 'zk_proveedor' ('cif'),
12  CONSTRAINT 'FK_ped_emp' FOREIGN KEY ('id_empleado')
13    REFERENCES 'zk_usuario' ('user')
14 ) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

Listado 5.21: Tabla Pedido Línea

5. Desarrollo del proyecto

```
1 CREATE TABLE IF NOT EXISTS 'zk_pedido_linea' (  
2   'id' int(11) NOT NULL AUTO_INCREMENT,  
3   'cantidad' int(11) NOT NULL,  
4   'coste' double NOT NULL,  
5   'id_pedido' int(11) NOT NULL,  
6   'id_producto' varchar(11) NOT NULL,  
7   'fecha' timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,  
8 PRIMARY KEY ('id'),  
9   CONSTRAINT 'FK_pl_ped' FOREIGN KEY ('id_pedido')  
10    REFERENCES 'zk_pedido' ('id'),  
11   CONSTRAINT 'FK_pl_prod' FOREIGN KEY ('id_producto')  
12    REFERENCES 'zk_producto' ('codigo')  
13 ) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

Listado 5.22: Tabla Peso

```
1 CREATE TABLE IF NOT EXISTS 'zk_peso' (  
2   'id' int(11) NOT NULL AUTO_INCREMENT,  
3   'mascota' int(11) NOT NULL,  
4   'valor' float NOT NULL,  
5   'fecha' timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,  
6 PRIMARY KEY ('id'),  
7   CONSTRAINT 'FK_peso_masc' FOREIGN KEY ('mascota')  
8    REFERENCES 'zk_mascota' ('id')  
9 ) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

Listado 5.23: Tabla Poblacion

```
1 CREATE TABLE IF NOT EXISTS 'zk_poblacion' (  
2   'id' int(11) NOT NULL AUTO_INCREMENT,  
3   'provincia' int(11) NOT NULL,  
4   'poblacion' varchar(200) NOT NULL,  
5 PRIMARY KEY ('id'),  
6   CONSTRAINT 'FK_pob_prov' FOREIGN KEY ('provincia')  
7    REFERENCES 'zk_provincia' ('id')  
8 ) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

Listado 5.24: Tabla Producto

```
1 CREATE TABLE IF NOT EXISTS 'zk_producto' (  
2   'codigo' varchar(11) NOT NULL,  
3   'id_familia' int(11) DEFAULT NULL,
```



```

4  'id_proveedor' varchar(9) DEFAULT NULL,
5  'nombre' varchar(100) NOT NULL,
6  'pvp' float NOT NULL,
7  'iva' int(11) NOT NULL,
8  'precio' float DEFAULT NULL,
9  'stock' int(11) NOT NULL,
10 'descripcion' varchar(250) DEFAULT NULL,
11 'fotografia' varchar(250) DEFAULT NULL,
12 'observaciones' varchar(250) DEFAULT NULL,
13 'fecha_alta' timestamp NULL DEFAULT CURRENT_TIMESTAMP,
14 PRIMARY KEY ('codigo'),
15 CONSTRAINT 'FK_prod_fam' FOREIGN KEY ('id_familia')
16 REFERENCES 'zk_producto_familia' ('id'),
17 CONSTRAINT 'FK_prod_provee' FOREIGN KEY ('id_proveedor')
18 REFERENCES 'zk_proveedor' ('cif'),
19 CONSTRAINT 'FK_prod_iva' FOREIGN KEY ('iva')
20 REFERENCES 'zk_iva' ('id')
21 ) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

Listado 5.25: Tabla Producto Familia

```

1 CREATE TABLE IF NOT EXISTS 'zk_producto_familia' (
2   'id' int(11) NOT NULL AUTO_INCREMENT,
3   'nombre' varchar(200) NOT NULL,
4   'descripcion' varchar(500) DEFAULT ' ',
5   'tratamiento' tinyint(1) NOT NULL DEFAULT '0' COMMENT '1)
   Tratamiento 0) Otro',
6   PRIMARY KEY ('id'),
7   UNIQUE KEY 'nombre' ('nombre')
8 ) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

Listado 5.26: Tabla Proveedor

```

1 CREATE TABLE IF NOT EXISTS 'zk_proveedor' (
2   'cif' varchar(9) NOT NULL,
3   'nombre' varchar(200) NOT NULL,
4   'direccion' varchar(200) NOT NULL,
5   'poblacion' int(11) NOT NULL,
6   'provincia' int(11) NOT NULL,
7   'telefono' int(11) NOT NULL,
8   'telefono2' int(11) DEFAULT NULL,
9   'fax' int(11) DEFAULT NULL,

```

5. Desarrollo del proyecto

```
10 'email' varchar(100) DEFAULT NULL,
11 'fecha_alta' timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,
12 'fecha_baja' timestamp NULL DEFAULT NULL,
13 'observaciones' varchar(400) DEFAULT NULL,
14 'contacto' varchar(100) DEFAULT NULL,
15 PRIMARY KEY ('cif'),
16 CONSTRAINT 'FK_provee_pob' FOREIGN KEY ('poblacion')
17 REFERENCES 'zk_poblacion' ('id'),
18 CONSTRAINT 'FK_provee_prov' FOREIGN KEY ('provincia')
19 REFERENCES 'zk_provincia' ('id')
20 ) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

Listado 5.27: Tabla Provincia

```
1 CREATE TABLE IF NOT EXISTS 'zk_provincia' (
2   'id' int(11) NOT NULL AUTO_INCREMENT,
3   'provincia' varchar(200) NOT NULL,
4   PRIMARY KEY ('id'),
5   UNIQUE KEY 'provincia' ('provincia')
6 ) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

Listado 5.28: Tabla Raza

```
1 CREATE TABLE IF NOT EXISTS 'zk_raza' (
2   'id' int(11) NOT NULL AUTO_INCREMENT,
3   'especie' int(11) NOT NULL,
4   'raza' varchar(200) NOT NULL,
5   PRIMARY KEY ('id'),
6   UNIQUE KEY 'raza' ('raza'),
7   CONSTRAINT 'FK_raza_esp' FOREIGN KEY ('especie')
8   REFERENCES 'zk_especie' ('id')
9 ) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

Listado 5.29: Tabla Servicio

```
1 CREATE TABLE IF NOT EXISTS 'zk_servicio' (
2   'codigo' varchar(9) NOT NULL,
3   'servicio' varchar(200) NOT NULL,
4   'descripcion' varchar(200) DEFAULT NULL,
5   'precio' float NOT NULL,
6   'id_iva' int(11) NOT NULL,
7   'id_familia' int(11) DEFAULT NULL,
```

```

8 PRIMARY KEY ( 'codigo' ),
9 CONSTRAINT 'FK_serv_iva' FOREIGN KEY ( 'id_iva' )
10 REFERENCES 'zk_iva' ( 'id' ),
11 CONSTRAINT 'FK_serv_fam' FOREIGN KEY ( 'id_familia' )
12 REFERENCES 'zk_servicio_familia' ( 'id' )
13 ) ENGINE=InnoDB DEFAULT CHARSET=utf8 ;

```

Listado 5.30: Tabla Servicio Familia

```

1 CREATE TABLE IF NOT EXISTS 'zk_servicio_familia' (
2   'id' int(11) NOT NULL AUTO_INCREMENT,
3   'nombre' varchar(200) NOT NULL,
4   'descripcion' varchar(500) DEFAULT ' ',
5   PRIMARY KEY ( 'id' ),
6   UNIQUE KEY 'nombre' ( 'nombre' )
7 ) ENGINE=InnoDB DEFAULT CHARSET=utf8 ;

```

Listado 5.31: Tabla Usuario

```

1 CREATE TABLE IF NOT EXISTS 'zk_usuario' (
2   'user' varchar(20) NOT NULL,
3   'password' varchar(50) NOT NULL,
4   'tipo' int(2) NOT NULL DEFAULT '3' COMMENT '1)Admin 2)
5     Veterinario 3)Empleado',
6   'fecha_alta' timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,
7   'nombre' varchar(100) NOT NULL,
8   'apellidos' varchar(100) NOT NULL,
9   'nif' varchar(10) NOT NULL,
10  'direccion' varchar(100) NOT NULL,
11  'ciudad' int(11) NOT NULL,
12  'provincia' int(11) NOT NULL,
13  'telefono' int(9) DEFAULT NULL,
14  'movil' int(9) DEFAULT NULL,
15  'email' varchar(100) DEFAULT NULL,
16  'nss' varchar(100) DEFAULT NULL,
17  PRIMARY KEY ( 'user' ),
18  CONSTRAINT 'FK_user_pob' FOREIGN KEY ( 'ciudad' )
19    REFERENCES 'zk_poblacion' ( 'id' ),
20  CONSTRAINT 'FK_user_prov' FOREIGN KEY ( 'provincia' )
21    REFERENCES 'zk_provincia' ( 'id' )
22 ) ENGINE=InnoDB DEFAULT CHARSET=utf8 ;

```

5. Desarrollo del proyecto

Listado 5.32: Tabla Vacuna

```
1 CREATE TABLE IF NOT EXISTS 'zk_vacuna' (  
2   'id' int(11) NOT NULL AUTO_INCREMENT,  
3   'nombre' varchar(255) NOT NULL,  
4   'descripcion' varchar(1000) DEFAULT NULL,  
5   'especie' int(11) NOT NULL,  
6   'dias' int(4) NOT NULL,  
7   PRIMARY KEY ('id'),  
8   CONSTRAINT 'FK_vac_esp' FOREIGN KEY ('especie')  
9     REFERENCES 'zk_especie' ('id')  
10 ) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

Listado 5.33: Tabla Venta

```
1 CREATE TABLE IF NOT EXISTS 'zk_venta' (  
2   'id' int(11) NOT NULL AUTO_INCREMENT,  
3   'fecha' timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,  
4   'id_cliente' int(11) NOT NULL,  
5   'id_vendedor' varchar(20) NOT NULL,  
6   'id_veterinario' varchar(20) DEFAULT NULL,  
7   'albaran' varchar(255) DEFAULT NULL COMMENT 'ruta del  
8     albaran',  
9   'factura' int(11) NOT NULL DEFAULT '0' COMMENT 'id de la  
10     factura',  
11   'facturado' tinyint(1) NOT NULL DEFAULT '0',  
12   PRIMARY KEY ('id'),  
13   CONSTRAINT 'FK_venta_cli' FOREIGN KEY ('id_cliente')  
14     REFERENCES 'zk_cliente' ('id'),  
15   CONSTRAINT 'FK_venta_emp' FOREIGN KEY ('id_vendedor')  
16     REFERENCES 'zk_usuario' ('user'),  
17   CONSTRAINT 'FK_venta_vet' FOREIGN KEY ('id_veterinario')  
18     REFERENCES 'zk_usuario' ('user'),  
19   CONSTRAINT 'FK_venta_fact' FOREIGN KEY ('factura')  
20     REFERENCES 'zk_factura' ('id')  
21 ) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

Listado 5.34: Tabla Venta Línea

```
1 CREATE TABLE IF NOT EXISTS 'zk_venta_linea' (  
2   'id' int(11) NOT NULL AUTO_INCREMENT,  
3   'id_venta' int(11) NOT NULL,  
4   'id_producto' varchar(11) NOT NULL,
```

```

5  'cantidad' int(11) NOT NULL,
6  'fecha' timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,
7  'tipo' smallint(1) NOT NULL COMMENT '1)Producto 2)Servicio
   ',
8  'pvp' float DEFAULT NULL,
9  'iva' int(2) DEFAULT NULL,
10 PRIMARY KEY ('id'),
11 CONSTRAINT 'FK_vl_venta' FOREIGN KEY ('id_venta')
12 REFERENCES 'zk_venta' ('id')
13 ) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

5.6. Pruebas y validación

Una de las etapas más importantes dentro del desarrollo del *software* es la realización de pruebas periódicas. Todo sistema debe ser probado lo más exhaustivamente posible antes de la entrega al cliente final.

5.6.1. Plan de pruebas

El plan de pruebas ha consistido en:

- Pruebas de unidad: probaremos la lógica del módulo (pruebas de caja blanca) y los distintos aspectos de las funciones que debe realizar el módulo (pruebas de caja negra).
- Pruebas de integración: probaremos la integración de los componentes *software* entre sí.
- Pruebas de sistema: probaremos la integración de todos los elementos del sistema: *hardware*, *software* y usuario.
- Prueba de aceptación: el usuario comprueba y está satisfecho con el producto obtenido.

Según las recomendaciones de IEEE 829 [18], lo idóneo es que el desarrollador no sea el único que prueba la aplicación, ya que éste tiende a probar casos en los que los resultados sean satisfactorios, con lo que cuantas más personas ajenas al desarrollo del mismo la prueben más fructíferas serán estas pruebas. En este apartado hay que agradecer las pruebas por parte del tutor y del propio cliente.

5.6.2. Especificación del diseño de pruebas

- *Durante el desarrollo del software*: las pruebas que se realizaron en esta etapa fueron pruebas de unidad y de integración. Según se iba desarrollando un módulo se iba probando introduciendo los datos requeridos por este, realizando comprobaciones de todas las combinaciones posibles. Además hemos incluido restricciones de entrada como es el caso del NIF/CIF, números de teléfono, email, etc.

Por parte del cliente, durante el desarrollo del *software*, hemos tenido distintas reuniones en las que se le ha presentado el estado en el que se encontraba; en estas reuniones hemos obtenido sugerencias aportadas por el propio cliente, como es crear un acceso directo en el menú para crear una nueva venta rápida, en la que no se asigna la venta a un cliente, son casos en los que prima una transacción rápida; o también la forma en la que añadimos nuevos productos en esa venta.

Uno de los puntos críticos, es la iniciación de una nueva venta, en la que se ha se probado que los cálculos fueran correctos, además de esto, la generación de los albaranes y las facturas también son partes importantes, así que se probó con mayor detenimiento.

Ya que la aplicación permite subir documentos para asociarlos a los Historiales de las Mascotas, se probó que los documentos se suben correctamente en los directorios establecidos, que se utiliza un direccionamiento relativo y que a la hora de descargarlo el documento sea el correcto.

- *Software finalizado*: una vez que la aplicación logró cumplir con todos los requisitos, se alojó en un *host* gratuito, con esto probamos de una forma más exhaustiva nuestra aplicación, realizando pruebas desde su instalación, configuración y pruebas realizadas con anterioridad.

En estas pruebas realizamos todo tipo de pruebas como producto finalizado. Iniciamos la instalación en un servidor de Internet, para ver cómo se comporta en un entorno real, se configuró con la base de datos elegida, MySQL. En un primer momento la creación automática de las tablas por parte de los ficheros DataSourceClase.java, fueron erróneas, pero tras las comprobaciones y modificaciones oportunas se generaron correctamente las tablas con sus respectivos atributos.

Tras estas pruebas se continuó creando nuevas especies, nuevas mascotas, nuevas familias de producto y servicios, nuevo tipo de IVA y una nueva vacuna.

A continuación, se creó un nuevo Cliente, se probó por parte del cliente que los componentes «*combobox*» no eran lo suficientemente claros, así que se optó por

cambiarlos por los clásicos «*selectbox*», propensos a menores errores por parte del cliente. Después se crearon otros objetos como eran: Mascotas, Proveedores, Productos, Servicios. Se detectó lugares donde los campos opcionales, una vez guardados en la base de datos, se representaban en la aplicación como *null*, se corrigió estos fallos de interfaz sin mucha dificultad.

Finalmente, se comprobaron minuciosamente los albaranes y facturas, ya que son documentos fiscales que deberán ser entregados en Hacienda para realizar la declaración de la renta.

5. Desarrollo del proyecto

6. Resumen

En esta sección se expone un resumen sobre el PFC que se presenta.

6.1. Descripción general

La Aplicación de Internet Enriquecida para la Gestión Informática de Clínicas Veterinarias, nace a petición de la clínica veterinaria «El Rey de la Casa». Esta joven empresa realizaba toda la gestión de una forma clásica y manual, sin ningún tipo de ayuda informática, con los inconvenientes que esto conlleva. Para este desarrollo hemos utilizado *software* gratuito, ya que el cliente no podía asumir costes de licencia. Cabe destacar que durante el desarrollo hemos mantenido una comunicación directa con el cliente, ya sea a través del correo electrónico o las reuniones periódicas que se han mantenido.

6.2. Características generales

Durante el desarrollo del proyecto hemos utilizado el modelo de ciclo de vida secuencial, además debemos destacar la utilización del *framework* ZK para la creación de aplicaciones RIA, en su versión gratuita; el uso de la programación orientada a objetos y la reutilización de código. Además de esto hemos seguido como patrón de diseño MVVM, basado en el patrón MVC. Como lenguaje de programación en el apartado lógico se ha utilizado Java y como sistema de gestión de base de datos MySQL en su versión Community.

Debemos destacar el carácter *cloud computing* de nuestra aplicación, con una disponibilidad e independencia total, ya que el único requisito por parte del cliente para acceder a la aplicación es utilizar un navegador web de Internet.

6.3. Objetivos

- Gestionar los datos de los clientes, mascotas, historiales, productos, proveedores, pedidos, ventas y citas, así como administrar y generar de manera dinámica facturas, informes y listados.

6. Resumen

- Dotar de accesibilidad a la aplicación web las 24 horas del día, desde cualquier dispositivo a través de Internet y sin necesidad de instalación de ningún componente, basta con un navegador web. Desarrollar una interfaz de usuario amigable e intuitiva junto con una buena documentación que ayude a utilizarla.
- Gestionar y consultar estadísticas y gráficas generadas dinámicamente a partir de los datos almacenados en la base de datos.

7. Conclusiones y trabajo futuro

7.1. Conclusiones

En este PFC se ha desarrollado una aplicación RIA capaz de llevar la gestión informática *online* de una clínica veterinaria, y de una forma amigable e intuitiva para el usuario final, es decir, para los veterinarios y gestores de la clínica.

En concreto, esta aplicación permite gestionar los datos de los clientes, mascotas, historiales, productos, proveedores, pedidos, ventas y citas, así como administrar y generar de manera dinámica facturas, informes y listados. Además, facilita la gestión y las consultas de estadísticas y gráficas generadas dinámicamente a partir de los datos almacenados en la base de datos.

Por otro lado, se trata de una aplicación web disponible las 24 horas del día, accesible desde cualquier dispositivo a través de Internet y sin necesidad de instalación de ningún componente, basta con un navegador web.

Para la creación de nuestra aplicación RIA se ha utilizado ZK, un *framework open source*, de la compañía Potix Corporation, con el cual conseguimos como resultado una aplicación en AJAX, sin la necesidad de saber AJAX, ni instalar ningún complemento adicional en la máquina del cliente. La lógica de negocio se ha programado en Java utilizando orientación a objetos y como SGBD hemos utilizado MySQL. Para la creación de la interfaz de usuario, hemos utilizado ZUML, lenguaje similar a XHTML propia del *framework* ZK, con el que conseguimos una interfaz amigable en relativamente poco tiempo.

7.2. Valoración personal

Al comienzo del desarrollo del PFC, los conocimientos de las principales tecnologías que se iba a utilizar eran escasos, incluso podemos decir que eran nulos, como es el caso del *framework* ZK, el patrón MVVM o el lenguaje de programación Java.

Java es el lenguaje de programación más utilizado en el mundo [15] y con él hemos programado la parte lógica de la aplicación, aunque al principio fue lento debido al aprendizaje, en la actualidad estoy satisfecho con los conocimientos y capacidades adquiridos.

Las apreciaciones referentes al *framework* ZK no pueden ser mejores, en un primer momento, el hecho de aprender un *framework* RIA del que no tenía cono-

7. Conclusiones y trabajo futuro

cimientos no me terminaron de convencer, pero pronto los resultados obtenidos me hicieron cambiar de opinión. La gran cantidad de componentes para el diseño de la interfaz, la posibilidad de programar la parte lógica en un lenguaje de alto nivel, la facilidad que da al desarrollar con la utilización del patrón de diseño MVVM, son argumentos de peso.

Con el patrón MVVM, pasa algo similar, carecía de conocimientos sobre éste, pero su integración con ZK lo hace altamente recomendable.

En general estoy satisfecho con los conocimientos y aptitudes adquiridas durante el desarrollo, he crecido como ingeniero y me ha dado la posibilidad de obtener una visión más cercana de proyectos de mayor envergadura. En proyectos así es donde recobra con más fuerza la importancia de la Ingeniería del Software.

7.3. Trabajo futuro

Como trabajo futuro proponemos las siguientes mejoras:

- Dotar a la aplicación de la capacidad de enviar SMS a los clientes para avisar de Citas próximas. Algunas de las compañías que ofrecen este servicio con una API para integrarlo en nuestra aplicación son: Lleida.net (<http://lleida.net>) o bien Altiria (<http://www.altiria.com>).
- Adaptación y utilización de la aplicación para otros comercios. Como proyecto próximo, tenemos el encargo de un aplicación para la gestión de una tienda de venta de Manzanilla en Sanlúcar de Barrameda.
- Traducción al inglés.

8. Agradecimientos

Después de estos años de formación, quería agradecer el apoyo de los compañeros de carrera, compartiendo horas de estudio y trabajo. A mi tutor Juan, por su dedicación, apoyo, innumerables correos electrónicos con prácticos consejos y por probar la aplicación, documentando y especificando los errores y fallos encontrados.

A mi madre, Carmen, gracias por estar ahí en los momentos malos y buenos. A mi novia, Laura, por su compañía, apoyo, comprensión y horas de estudio compartido. Y a mi padre, por ser lo que hoy soy.

8. *Agradecimientos*

A. Manual de instalación

En esta sección describiremos los pasos para la instalación correcta de la aplicación.

A.1. Instalación en el lado del cliente

Los requerimientos de *software* y de configuración por parte del cliente, como ya se comentó con anterioridad en los puntos anteriores 4.1.1, basta con un navegador web moderno, ZK es compatible con Internet Explorer 6-9, Firefox, Chrome, Safari, iOS Safari, Opera, Opera Mini... así que valdría con cualquiera de ellos para disfrutar de nuestra aplicación.

A.2. Instalación en el lado del servidor

Los requerimientos de *software* en este caso son más exigentes, además hay que realizar la configuración del servidor. Para ello, tenemos que tener instalado el siguiente *software*:

- JRE (*Java Runtime Environment*).
- Apache Tomcat 7.0.
- MySQL.

A.2.1. JRE (Java Runtime Environment)

Seguiremos los siguientes pasos:

1. Descargarse el JRE correspondiente al sistema operativo y a la arquitectura del servidor desde la página de Oracle: <http://www.oracle.com/technetwork/es/java/javase/downloads/index.html>.
2. Procedemos a su instalación según los pasos indicados en la propia página.

A.2.2. MySQL

A continuación describiremos los pasos para realizar una instalación y configuración correcta del sistema de gestión de base de datos MySQL:

1. Descargarlo de la página de MySQL según el sistema operativo del servidor:
<http://dev.mysql.com/downloads/mysql/>.
2. Procedemos a la instalación, si el sistema operativo es Windows, ésta se lleva a cabo como cualquiera otra con el asistente de instalación.
3. Indicar la contraseña para *Administrador*.
4. Creamos la base de datos: *CREATE database clinica*.

5. En nuestra aplicación tendremos que configurar los datos de acceso a la base de datos, esto se realiza en los ficheros DataSource.java:

```
private static final String url = "jdbc:mysql://url/clinica";  
private static final String user = "root";  
private static final String pwd = "password";
```

- Como ejemplo real de conexión con una base de datos alojada en OpenShift:

```
private static final String url = "jdbc:mysql://127.8.117.129:3306/clinica";  
private static final String user = "adminRhGQFZa";  
private static final String pwd = "AcDgIFBgFbh-";
```

A.2.3. Apache Tomcat 7.0

Una vez instalado JRE el servidor procedemos a la instalación y configuración de nuestro servidor web, que en este caso se describirá para Apache Tomcat 7.0. Se podría haber instalado cualquier otro servidor web como: JBoss, WebSphere, Glassfish, Google App Engine, Red Hat OpenShift Express...

A continuación, realizaremos los siguientes pasos:

1. Descargar el servidor web Apache Tomcat 7.0 de su página web: <http://tomcat.apache.org/download-70.cgi>.
2. Procedemos a la instalación, si el sistema operativo es Windows, ésta se lleva a cabo como cualquiera otra con el asistente de instalación.
3. En el paso *Choose Components*, añadimos los componentes *Service Startup* y *Host Manager*.

A.2. Instalación en el lado del servidor

4. En el paso *Configuration*, en el campo *User Name* introducimos nuestro nombre de administrador y en *Password*, la contraseña asociada a éste.
5. El siguiente paso hace referencia a la ruta en la que se encuentra la máquina virtual Java, que por defecto es: «*C:\Program Files (x86)\Java\jre7*» , aceptamos e instalamos.
6. Accedemos a la dirección <http://localhost:8080> y comprobamos que Tomcat está corriendo, si es así la instalación ha sido satisfactoria.
7. Pulsamos sobre «*Manager App*» y en la sección «*Archivo WAR a desplegar*» seleccionamos el archivo .WAR de nuestra aplicación, disponible en el CD adjunto, y a continuación pulsamos sobre «*Desplegar*».
8. Cuando ya se haya desplegado nuestra aplicación pulsamos sobre él, y ya tendríamos lista nuestra aplicación.

A. Manual de instalación

B. Manual de gestión de copias de seguridad

Las copias de seguridad del sistema, las debe realizar el administrador del sistema, para ello se debe realizar una copia de la base de datos y otra copia de los ficheros generados por el sistema y subidos a éste.

B.1. Copia de seguridad de la base de datos

La copia de la base de datos la podemos realizar utilizando la herramienta phpMyAdmin, es una manera gráfica y sencilla. A continuación explicaremos como realizarla con esta herramienta:

1. Accedemos a la página de phpMyAdmin:
 - a) Si la instalación es en local: <http://localhost/phpmyadmin/>
 - b) Si la instalación es un de servidor de Internet, como ejemplo: <https://clinica-elreydelacasa.rhcloud.com/phpmyadmin/>
2. Introducimos el nombre de usuario y la contraseña.
3. Una vez dentro seleccionamos la base de datos, en el ejemplo es «*joomlclínica*». (Véase Figura B.1).
4. En la pantalla de la Base de Datos, pulsamos sobre «*Exportar*». (Véase Figura B.2).
5. A continuación, en la pantalla «*Exportar*» (Véase Figura B.3), pulsamos sobre «*Continuar*» y se abrirá la ventana para su descargar del fichero *.sql*.
6. Seleccionamos la ubicación donde vayamos a guardar la copia de seguridad y aceptamos.

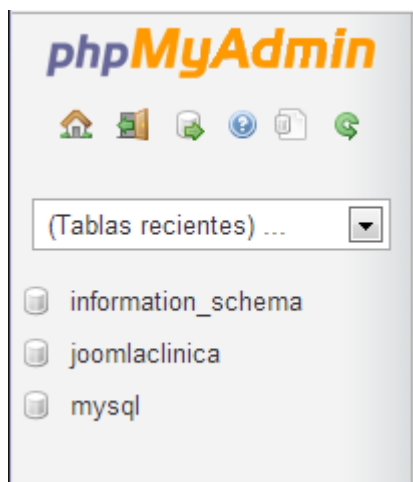


Figura B.1.: Seleccionar Base de Datos

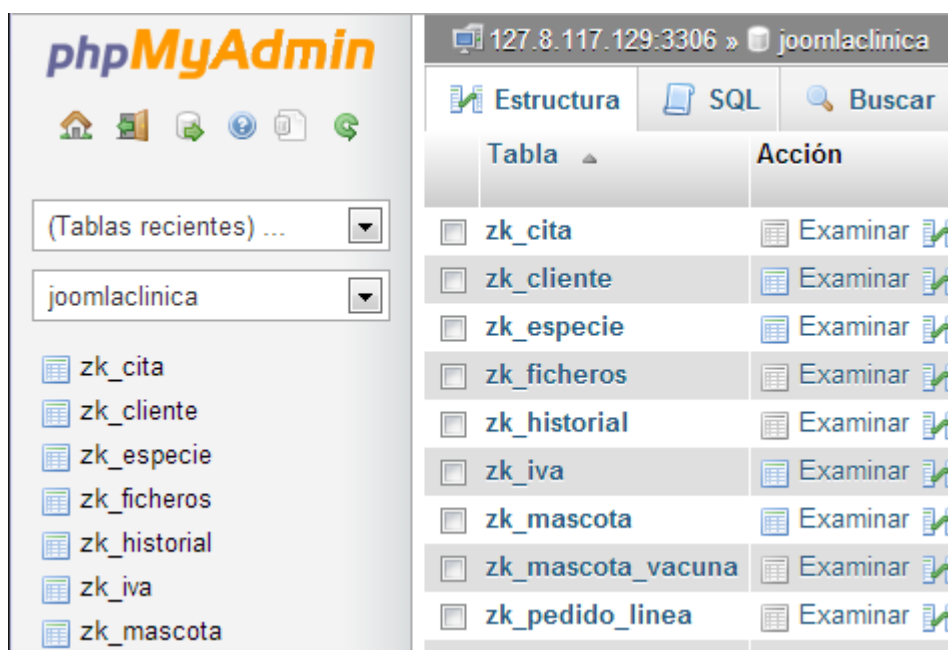


Figura B.2.: Seleccionar Exportar

127.8.117.129:3306 » joomlaclinica

Estructura SQL Buscar Generar una consulta Exportar Importar

Exportando tablas de la base de datos "joomlaclinica"

Método de exportación:

☒ Rápido - mostrar sólo el mínimo de opciones de configuración
☐ Personalizado - mostrar todas las opciones de configuración posibles

Salida:

☒ Guardar en el servidor en el directorio `/tmp/`
☐ Sobrecribir el(los) archivo(s) existente(s)

Formato:

SQL

Continuar

phpMyAdmin

(Tablas recientes) ...

joomlaclinica

- zk_cita
- zk_cliente
- zk_especie
- zk_ficheros
- zk_historial
- zk_iva
- zk_mascota
- zk_mascota_vacuna
- zk_pedido_linea
- zk_peso
- zk_poblacion
- zk_producto
- zk_producto_familia
- zk_proveedor
- zk_provincia

Figura B.3.: Pantalla Exportar

B.2. Restauración de la copia de seguridad de la base de datos

1. Accedemos a la página de phpMyAdmin:
 - a) Si la instalación es en local: <http://localhost/phpmyadmin/>
 - b) Si la instalación es un de servidor de Internet, como ejemplo: <https://clinica-elreydelacasa.rhcloud.com/phpmyadmin/>
2. Introducimos el nombre de usuario y la contraseña.
3. Una vez dentro seleccionamos la base de datos, en el ejemplo es «**joomlaci-nica**». (Véase Figura B.1).
4. En la pantalla de la Base de Datos, pulsamos sobre «*Importar*». (Véase Figura B.2).
5. A continuación, en la pantalla «*Importar*» (Véase Figura B.4), pulsamos sobre «*Seleccionar archivo*» y se abrirá la ventana para su subir el fichero *.sql*.
6. Seleccionamos el fichero que generó la copia de seguridad y aceptamos.
7. Por último pulsamos en «*Continuar*».

B.3. Copia de seguridad de los ficheros

Para realizar una copia de seguridad de los ficheros generados por el sistema: facturas y albaranes, o bien subidos al sistema, aunque los primeros el propio Sistema da la posibilidad de almacenarlos en la máquina cliente cuando éstos se generan; para ello debemos copiar las carpetas: **albaranes**, **facturaspdf** y **uploads**.

La ubicación de estas carpetas se encuentra dentro del directorio donde se desplegó la APP cuando se realizó la instalación:

- Si la instalación fue en local en una máquina con sistema operativo Windows, podemos encontrarlo en la ubicación: «*C:\Program Files (x86)\Apache Software Foundation\Tomcat 7.0\webapps\clinica*».
- Si la instalación es un servidor de Internet, podemos acceder a través del protocolo FTP y descargando las carpetas indicadas del directorio: «*webapps/clinica/*».

phpMyAdmin

(Tablas recientes) ...

joomlaclinica

zk_cita

zk_cliente

zk_especie

zk_ficheros

zk_historial

zk_iva

zk_mascota

zk_mascota_vacuna

zk_pedido_linea

zk_peso

zk_poblacion

zk_producto

zk_producto_familia

zk_proveedor

zk_provincia

zk_raza

zk_servicio

zk_servicio_familia

zk_usuario

zk_vacuna

zk_venta

zk_venta_linea

Crear tabla

127.8.117:129-3306 » joomlaclinica

Estructura

SQL

Buscar

Generar una consulta

Exportar

Importar

Operaciones

Privilegios

Más

Importando en la base de datos "joomlaclinica"

Archivo a importar:

El archivo puede ser comprimido (gzip, bzip2, zip) o descomprimido.
Un archivo comprimido tiene que terminar en `[formato].[compresión]`. Por ejemplo, `.sql.zip`

☐

Buscar en su ordenador:

Seleccionar archivo

No se ha seleccionado ningún archivo (Máximo: 200MB)

☐

Seleccionar directorio en el servidor web para subir los archivos `/tmp/`: No hay archivos para subir

Conjunto de caracteres del archivo: utf-8

Importación parcial:

☒ Permitir la interrupción de una importación en caso que el script detecte que se ha acercado al límite de tiempo PHP. (Esto podría ser un buen método para importar archivos grandes; sin embargo, puede dañar las transacciones.)

Número de filas a omitir, iniciando de la primer fila: 0

Formato:

SQL

Opciones específicas al formato:

Modalidad SQL compatible: NONE

☒ No utilizar AUTO_INCREMENT con el valor 0

Continuar

Figura B.4.: Pantalla Importar

B.4. Restauración de la copia de seguridad de ficheros

Si deseamos restaurar la copia de seguridad de los ficheros generados por el sistema, basta con que subamos las carpetas: ***albaranes***, ***facturaspdf*** y ***uploads*** en la ubicación correspondiente:

- Si la instalación fue en local con sistema operativo Windows: «*C:\Program Files (x86)\Apache Software Foundation\Tomcat 7.0\webapps\clinica*».
- Si la instalación es un servidor de Internet, podemos acceder a través del protocolo FTP y subiendo las carpetas indicadas al directorio: «*webapps/clinica/*».

C. Manual de usuario

En el siguiente apartado describiremos el manual de usuario lo más detallado posible para que el usuario final pueda recurrir a él en el caso de tener dudas.

C.1. Pantalla de acceso

Es la primera pantalla que se encontrará el usuario, deberá introducir su nombre de usuario asignado por el Administrador y una clave de acceso. Esta clave debe tener al menos 8 dígitos y tiene que contener caracteres numéricos y alfabéticos, con mayúsculas y minúsculas (Véase Figura C.1). Si se introduce un nombre de usuario o contraseña no válidos, el sistema alertará con un mensaje emergente.

Nota: El nombre de usuario para el administrador es ***admin*** y su contraseña temporal ***admin***, esta contraseña debe ser modificada inmediatamente.

C.2. Pantalla principal

En esta pantalla podemos observar el Menú principal, donde se encuentra cada uno de los apartados de los que se compone la aplicación: Inicio, Clientes, Mascotas, Productos, Servicios, Ventas, Proveedores, Estadísticas, Configuración y Nombre de usuario (Rol).

Además nos encontramos con dos ventanas de Alertas, una ventana de «*Alerta Stock Bajo*» y otra ventana de «*Alerta citas próximas*», ambas ventanas han sido demandadas por el cliente. En la ventana «*Alerta citas próximas*» si hacemos click con el botón derecho del ratón sobre una de las citas, podemos acceder al estado de las citas: Avisado, Acudido, Pendiente, Cancelar. Véase Figura C.2 para más detalles.

C.3. Gestión de clientes

Accedemos a la gestión de clientes a través del menú superior, y nos encontramos de forma predeterminada un listado vacío.

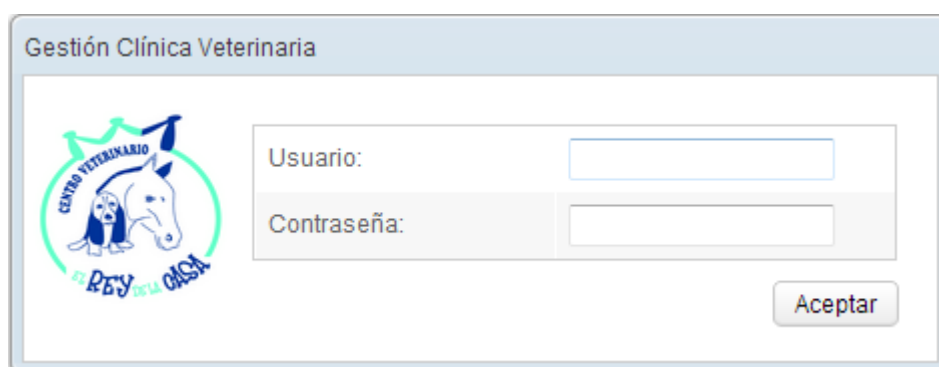


Figura C.1.: Pantalla de acceso

C.3.1. Nuevo cliente

Para añadir un nuevo cliente al sistema debemos pulsar sobre el botón Nuevo, dentro de la ventana Clientes, a continuación se abrirá una ventana con los campos en blanco, a rellenar por el empleado, (Véase Figura C.3).

Los campos con asterisco (*) son obligatorios de rellenar. El NIF/CIF tendrá que ser válido, sino el sistema alertará al usuario con un mensaje de error, este NIF/CIF no puede contener espacios ni guiones, únicamente caracteres alfanuméricos y un máximo de 9 dígitos.

C.3.2. Eliminar cliente

Para eliminar un cliente:

1. Hacemos click sobre el cliente que deseamos eliminar.
2. Se habilita el botón Eliminar.
3. En la ventana emergente que se abre, confirmamos su eliminación. (Ver Figura C.4).

C.3.3. Filtrar clientes

Para filtrar clientes basta con introducir en los campos de las respectivas columnas y el propio sistema se encarga de refrescar la página. (Véase Figura C.5). En el ejemplo podemos observar cómo se ha realizado un filtrado por *Apellidos* con el apellido «García».

Inicio

Inicio

Cientes

Mascotas

Productos

Servicios

Ventas

Proveedores

Estadísticas

Configuración

Agustín (Administrador)

Alerta Stock Bajo

Código	Nombre	Proveedor	PVP	Stock
para100	Paracetamol 500MG	Roche	3,03 €	0
frontline1m	Frontline 1ML	Roche	4,13 €	0
gasa100gr	Gasas de 100 gramos	Bayer	0,55 €	2
collar001	Collares Grandes	Bayer	6,00 €	1
imizol10ml	Imizol 10 ML	MD Veterinaria	3,01 €	1
financiacion300	Financiación 3000€	Banco Santander	3.000,00 €	1
financiacion600	Financiación 6000€	BBVA	6.000,00 €	2
anacarfa002	Carprodyl	Farmavet	1,20 €	2
anacarfa005	Carprodyl 1MG	Farmavet	2,00 €	2
anacimfa001	Cimalgex 8 mg	Farmavet	3,00 €	2
anacimmd001	Cimalgex 10mg	MD Veterinaria	3,00 €	2
anaimmd001	Imizol 5MG	MD Veterinaria	3,00 €	2
anabuba001	Ibuprofeno 0,7MG	Bayer	2,00 €	2
anaimifa001	Imizol 1MG	Farmavet	3,00 €	2

Alerta citas próximas

Estado	Fecha	Hora	Cliente	Teléfono	Mascota
Pendiente	2013-03-15	09:00:00	Venta Rápida	900100100	Pluto
Pendiente	2013-03-18	09:00:00	Laura de los Reyes Sainz	678222585	Miau

Cambiar estado

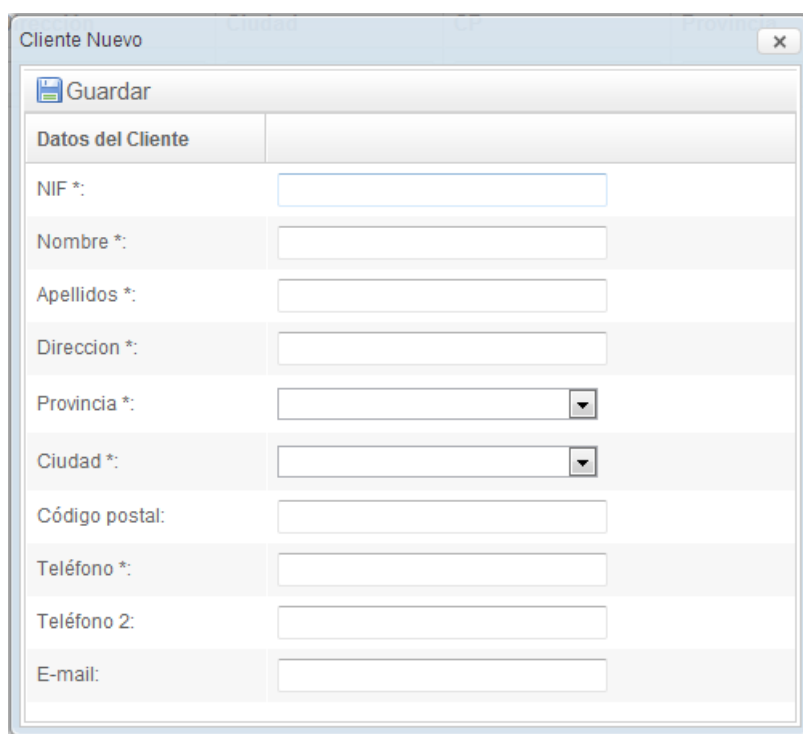
Avisado

Acudido

Pendiente

Cancelar

Figura C.2.: Pantalla principal



Cliente Nuevo

Guardar

Datos del Cliente

NIF *:

Nombre *:

Apellidos *:

Direccion *:

Provincia *:

Ciudad *:

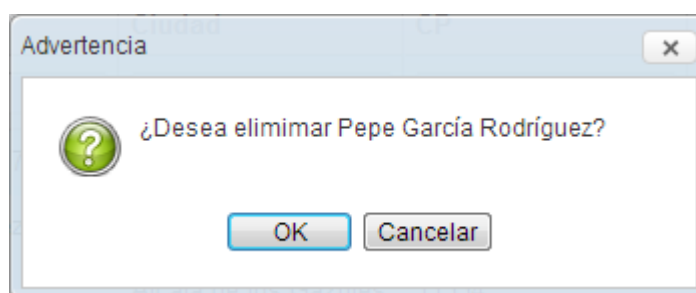
Código postal:

Teléfono *:

Teléfono 2:

E-mail:

Figura C.3.: Cliente nuevo




Advertencia


¿Desea eliminar Pepe García Rodríguez?


OK Cancelar


Figura C.4.: Cliente eliminar


Cientes


 Inicio


 Clientes ▾


 Mascotas

 Productos

 Servicios

 Ventas

 Nuevo

 Eliminar

NIF/CIF	Nombre	Apellidos	Dirección
		García	
12345678P	Pepe	García Rodríguez	Calle Puerto 17
55343453J	Carmen	García Patiño	José Rodriguez

Figura C.5.: Cliente filtrado

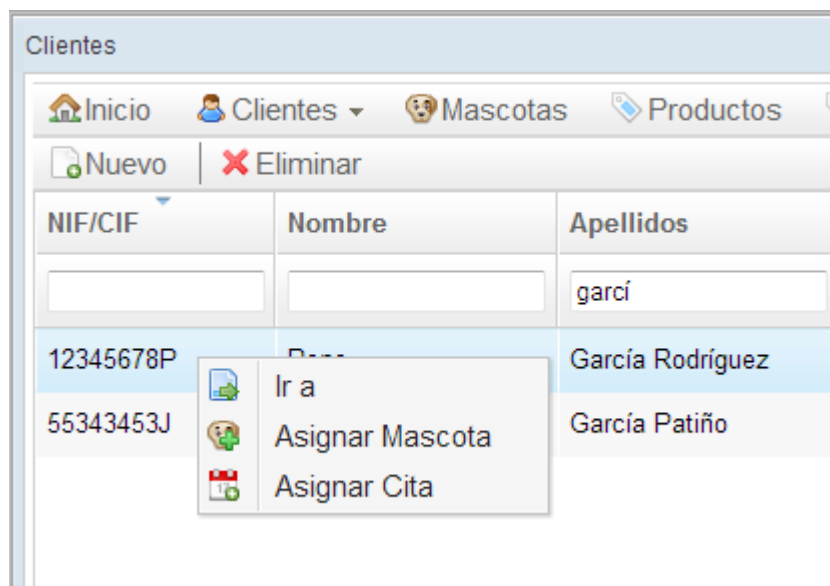


Figura C.6.: Cliente botón derecho

C.3.4. Asignar mascota

En el caso de que queramos añadir una nueva mascota, esta mascota estará asignada a un cliente, para ello hacemos click con el botón derecho del ratón sobre un cliente y se nos abrirá un menú emergente con las siguientes opciones: «*Ir a*», «*Asignar Mascota*», «*Asignar Cita*». (Véase Figura C.6)

1. Pulsamos sobre «*Asignar Mascota*».
2. Se abrirá una venta con los datos de la mascota.
3. Rellenamos los campos y pulsamos en «*Guardar*». Véase Figura C.7.

C.3.5. Asignar cita

En el caso de que queramos añadir una nueva cita, esta cita estará asignada a un cliente, para ello hacemos click con el botón derecho del ratón sobre un cliente y se nos abrirá un menú emergente con las siguientes opciones: «*Ir a*», «*Asignar Mascota*», «*Asignar Cita*». (Véase Figura C.6)

1. Pulsamos sobre «*Asignar Cita*».
2. Se abrirá una venta con los datos de la cita. (Véase Figura C.8).

Asignar mascota Ciudad CP Provincia X

Guardar

Datos Mascota	
N° Chip *	<input type="text"/>
Nombre *	<input type="text"/>
Fecha nacimiento *	<input type="text"/> 31
Sexo *	<input type="text"/> ▼
Peso (kg)	<input type="text"/>
Altura (cm)	<input type="text"/>
Especie	<input type="text"/> ▼
Raza	<input type="text"/> ▼
Pelo:	<input type="text"/> ▼
Observaciones:	<input type="text"/>

Figura C.7.: Cliente asignar mascota

Figura C.8.: Cliente asignar cita

3. Debemos elegir la mascota a la que va dirigida la cita, la fecha, hora disponible, servicio y alguna observación sobre la cita.
4. Por último pulsamos en «*Guardar*».

C.3.6. Ver perfil del cliente

En el perfil del cliente, vamos a tener los datos personales e información relevante sobre este, como las mascotas que tiene, las citas concertadas y las ventas realizadas por éste. En este caso podemos acceder al perfil del cliente haciendo doble click sobre su tupla o con el botón derecho y a continuación «*Ir a*». (Véase Figura C.6).

La ventana de la Figura C.9 hace referencia al perfil del cliente, en la que podemos observar lo siguiente:

- **En la parte superior se encuentra las pestañas:** Datos, Mascotas, Citas y Ventas.
- **Botón Modificar:** si queremos modificar los datos del cliente, pulsamos sobre este botón, entonces se habilitarán los campos para su modificación. Una vez hayamos terminado con los cambios, pulsaremos sobre el botón «*Guardar*».
- **Botón Guardar:** Este botón está deshabilitado de forma predeterminada como seguridad, sólo se activa pulsando antes sobre el botón «*Modificar*». Guarda los cambios realizados.

- **Botón Cancelar:** No es visible de forma predeterminada. Se hace visible cuando pulsamos sobre «*Modificar*». Si hemos realizado cambios y deseamos cancelar estos cambios, pulsamos sobre «*Cancelar*» y volverá a los datos originales. (Véase Figura C.10).

Pestaña Mascota

En esta pestaña se muestra las mascotas que tiene el cliente asignadas, las operaciones disponibles son:

- Añadir una nueva Mascota pulsando sobre «Nuevo».
- Eliminar Mascota, seleccionado previamente la mascota y pulsando sobre «Eliminar».
- Ordenar las mascotas pulsando sobre las columnas, se ordenará ascendentemente o descendentemente.

Pestaña Citas

En esta pestaña se muestra las citas concertadas el cliente, tanto las pasadas como las futuras, las operaciones disponibles son:

- Añadir nueva Cita, pulsando sobre «Nuevo».
- Eliminar Cita, seleccionado previamente la cita y pulsando a continuación sobre «Eliminar».
- Ordenar las citas pulsando sobre las columnas, se ordenará ascendentemente o descendentemente.

Pestaña Ventas

En esta pestaña se muestra las ventas realizadas por el cliente, las operaciones disponibles son:

- Ir a, pulsando con el botón derecho sobre la Venta.
- Ordenar las ventas pulsando sobre las columnas, se ordenará ascendentemente o descendentemente.

C.4. Gestión de mascotas

Accedemos a la gestión de mascotas pulsando sobre el botón «*Mascotas*» en el menú principal.

Perfil Cliente

Inicio

Cientes

Mascotas

Productos

Servicios

Ventas

Proveedores

Estadísticas

Configuración

Agu

Datos

Mascotas

Citas

Ventas

Modificar

Guardar

Dirección

NIF/CIF

12345678P

Nombre

Pepe

Apellidos

García Rodríguez

Dirección

Calle Puerto 17

Provincia

Sevilla

Ciudad

Dos Hermanas

Código postal

14303

Email

E-mail

pepe@gmail.com

Teléfono/Fax

Teléfono 1

954321245

Teléfono 2

0

Figura C.9.: Cliente perfil

210

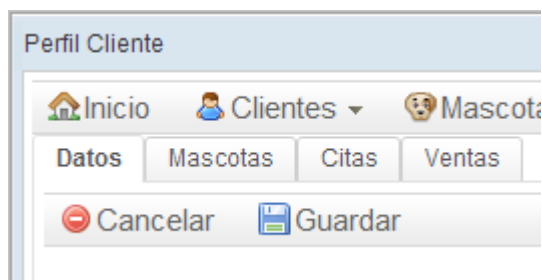


Figura C.10.: Cliente botón cancelar

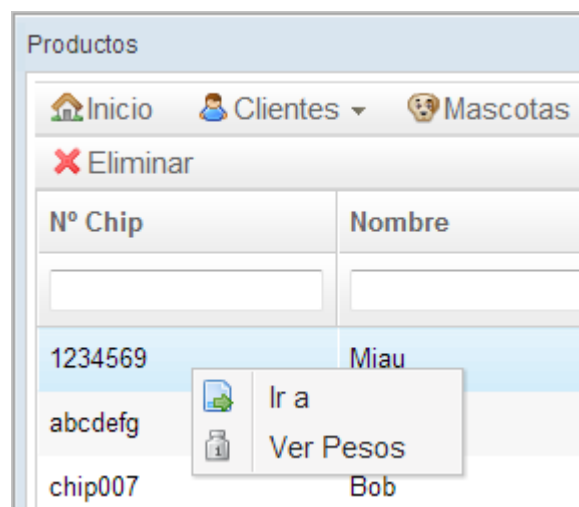


Figura C.11.: Mascota botón derecho

C.4.1. Ver perfil de mascota

En el perfil de la mascota, vamos a tener los datos e información relevante sobre la mascota, como los historiales que tiene, las citas y las vacunas suministradas a éste. En este caso podemos acceder al perfil de la mascota haciendo doble click sobre su tupla o con el botón derecho y a continuación «Ir a». (Véase Figura C.11).

La ventana de la Figura C.12 hace referencia al perfil del cliente, en la que podemos observar lo siguiente:

- **En la parte superior se encuentra las pestañas:** Datos, Historial, Citas y Vacunas.
- **Botón Modificar:** Si queremos modificar los datos de la mascota, pulsamos sobre este botón, entonces se habilitarán los campos para su modificación. Una

vez hayamos terminado con los cambios, pulsaremos sobre el botón «*Guardar*».

- **Botón Guardar:** Este botón está deshabilitado de forma predeterminada como seguridad, sólo se activa pulsando previamente sobre el botón «*Modificar*». Guarda los cambios realizados.
- **Botón Cancelar:** No es visible de forma predeterminada. Se hace visible cuando pulsamos sobre «*Modificar*». Si hemos realizado cambios y deseamos cancelar estos cambios, pulsamos sobre «*Cancelar*» y volverá a los datos originales. (Véase Figura C.10).
- **Modificar propietario:**
 1. Para ello debemos previamente pulsar sobre el botón «*Modificar*».
 2. A continuación pulsamos sobre el nombre del propietario y se abrirá una ventana emergente con el listado de clientes.
 3. Seleccionamos el cliente y pulsamos «*Enviar*».
 4. Por último pulsamos en «*Guardar*».

Pestaña Historial

En esta pestaña se muestra los historiales que tiene la mascota, las operaciones disponibles son:

- Añadir un nuevo Historial pulsando sobre «Nuevo»:
 - Se abrirá la siguiente ventana, con los siguientes campos: Nombre de la mascota, Peso, Tipo de Visita [Normal, Urgencias, Vacunas], Anamnesis, Diagnóstico y Tratamiento. (Véase Figura C.13).
 - Si se desea adjuntar documentos, pulsamos en la pestaña «Documentos» y pulsamos en «*Adjuntar*».
 - Si se desea descargar un documento, pulsamos sobre el documento y pulsamos en «*Descargar*».
 - Si se desea eliminar un documento, pulsamos sobre el documento y a continuación en «*Eliminar*».
 - Para visualizar un documento, en el caso de que sea del tipo Imagen, si pulsamos sobre éste en la parte derecha se mostrará. (Véase Figura C.14).
- Eliminar Historial, seleccionado previamente la mascota y pulsando sobre «*Eliminar*».
- Ordenar los historiales pulsando sobre las columnas, se ordenará ascendentemente o descendentemente.

Perfil Mascota

[Inicio](#) [Clientes](#) [Mascotas](#) [Productos](#) [Servicios](#) [Ventas](#) [Proveedores](#) [Estadísticas](#) [Configuración](#) [Agustín \(Administrador\)](#)

[Datos](#) [Historial](#) [Citas](#) [Vacunas](#)

[Modificar](#) [Guardar](#)

Datos

N° Chip	1234569
Nombre	Miau
Sexo	Hembra
Fecha nacimiento	02-Jul-2012
Fecha defunción	
Estado	<input checked="" type="radio"/> Activo <input type="radio"/> Inactivo

Datos de contacto

Nombre	Laura de los Reyes Sainz
Dirección	Calle Bolsa 84, 1ºB
Ciudad	Sanlúcar de Barrameda
Teléfono 1	678222585
Teléfono 2	
Email	risan7.pete@gmail.com

Características

Especie	Canina
Raza	Bulldog
Peso (kg)	3,85 kg
Altura (cm)	20 cm
Pelo	Corto

Observaciones

Observaciones

Figura C.12.: Mascota perfil

Historial Nuevo

Guardar Documentos

Datos

Mascota **Miau**

Peso **0**

Tipo de Visita **▼**

Anamnesis

Diagnóstico

Tratamiento

Figura C.13.: Mascota historial

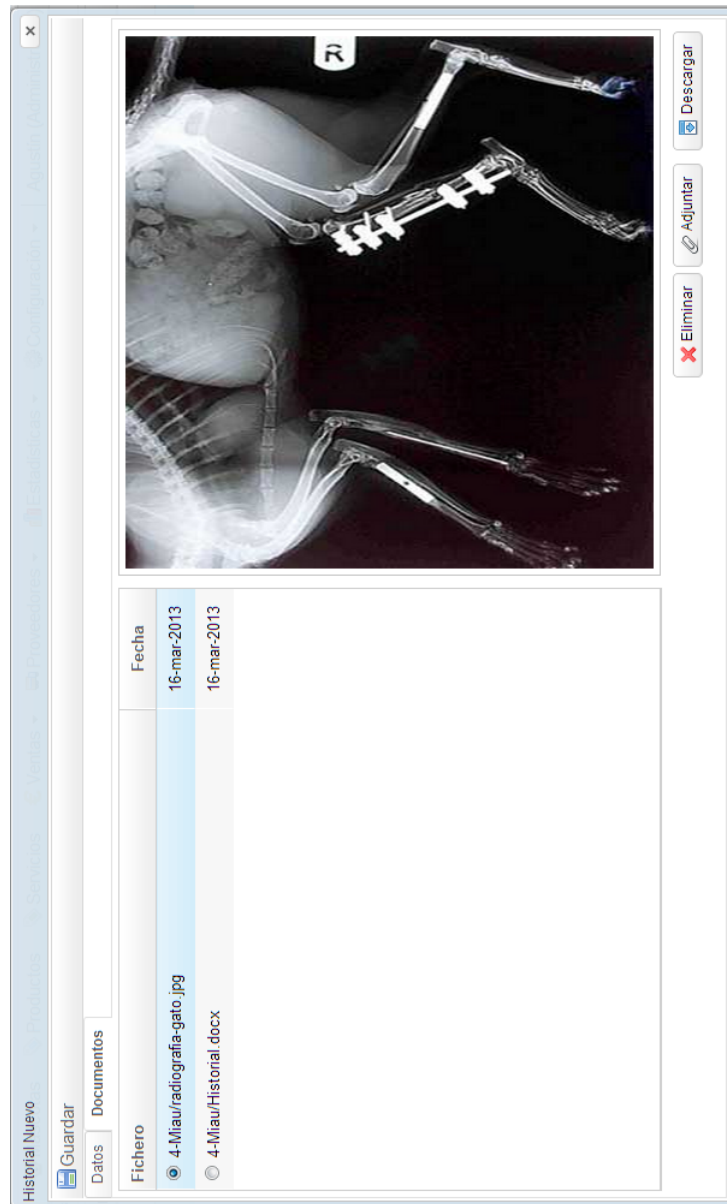


Figura C.14.: Mascota historial imagen

Pestaña Citas

El comportamiento de esta pestaña es idéntico al de la Sección C.3.5, salvo que el campo Mascota tendrá el valor de la mascota en curso.

Pestaña Vacunas

En esta pestaña se muestra las vacunas suministradas a la mascota, las operaciones disponibles son:

- Nueva vacuna, pulsando sobre el botón «*Nuevo*» y se abrirá la ventana con las vacunas disponibles para la especie de la mascota.
- Eliminar vacuna, seleccionando la que se desee eliminar y a continuación pulsar sobre el botón «*Eliminar*».
- Ordenar las vacunas pulsando sobre las columnas, se ordenará ascendentemente o descendentemente.

C.4.2. Eliminar cliente

Para eliminar un cliente:

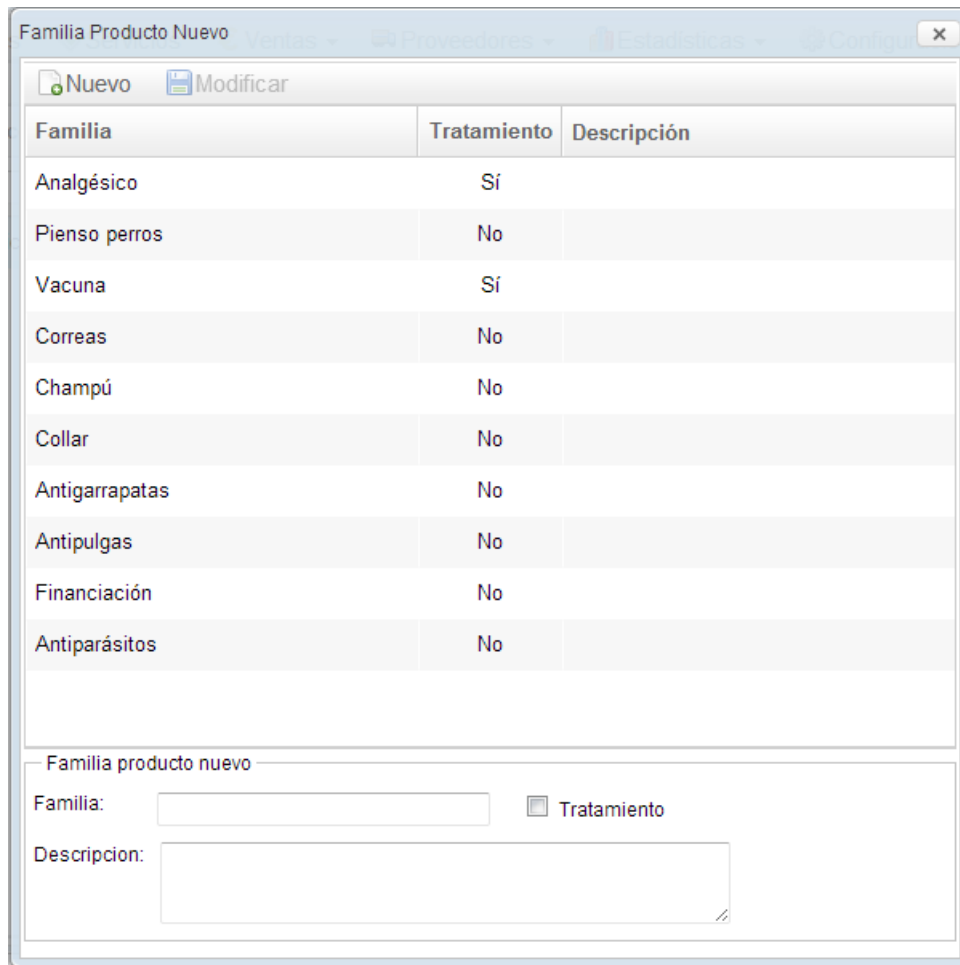
1. Hacemos click sobre el cliente que deseamos eliminar.
2. Se habilita el botón Eliminar.
3. En la ventana emergente que se abre, confirmamos su eliminación. (Ver Figura C.4).

C.5. Gestión de productos

Antes de añadir un nuevo producto debemos tener el proveedor al que pertenece el producto, la familia a la que pertenece el producto y el tipo de IVA que tendrá éste.

C.5.1. Nueva familia de productos

La ventana de nueva familia de productos, gestionamos las diferentes familias de productos que existirán en el sistema. (Véase Figura C.15).



The screenshot shows a window titled 'Familia Producto Nuevo' with a menu bar containing 'Ventas', 'Proveedores', 'Estadísticas', and 'Configuración'. Below the menu bar are two buttons: 'Nuevo' (with a green plus icon) and 'Modificar' (with a blue document icon). The main area contains a table with three columns: 'Familia', 'Tratamiento', and 'Descripción'. The table lists ten product families with their corresponding treatments. Below the table, there is a section titled 'Familia producto nuevo' with a 'Familia:' label and a text input field, a 'Tratamiento' checkbox, and a 'Descripción:' label with a larger text input area.

Familia	Tratamiento	Descripción
Analgésico	Sí	
Pienso perros	No	
Vacuna	Sí	
Correas	No	
Champú	No	
Collar	No	
Antigarrapatas	No	
Antipulgas	No	
Financiación	No	
Antiparásitos	No	

Familia producto nuevo

Familia: ☐ Tratamiento

Descripción:

Figura C.15.: Nueva familia de productos

Añadir nueva familia de productos

Introducimos el nombre de la familia de producto, seleccionamos en «*Tratamiento*» si es un medicamento y una descripción si se desea.

Para añadirlo pulsamos en el botón «*Nuevo*».

Modificar familia de productos

Para modificar una familia de productos, seleccionamos la familia de producto, modificamos el campo que se requiere y pulsamos en «*Modificar*».

Datos del Producto	
Código	anacarfa001
Nombre	Carprodyl
Familia	Analgésico
Proveedor	Farmavet
IVA	21
Precio	3
PVP	4,2
Stock	3
Imagen	

Figura C.16.: Producto modificar

C.5.2. Nuevo producto

Para añadir un nuevo producto en el sistema, el proceso a seguir es el mismo que el proceso para añadir un nuevo cliente, (Véase Sección C.3.1).

C.5.3. Modificar producto

Para modificar los datos de un producto, los pasos a seguir son los siguientes:

1. Doble click sobre la tupla de este en el listado de productos.
2. Se abre una ventana con los datos del producto. (Véase Figura C.16).
3. Los campos «Código», «Nombre», «Familia» y «Proveedor» no son modificables debido a restricciones de integridad del sistema. Modificamos los campos necesarios. Si el tipo de IVA asociado no está dado de alta en el sistema se deberá dar de alta a éste con anterioridad.
4. Por último, pulsamos en el botón «*Guardar*» para almacenar los cambios.

C.5.4. Filtrar productos

Para filtrar un producto, el proceso a seguir es el mismo que el proceso para filtrar un cliente, (Véase Sección C.3.3).

C.5.5. Eliminar producto

Para eliminar un producto:

1. Hacemos click sobre el producto que deseamos eliminar.
2. Se habilita el botón Eliminar.
3. En la ventana emergente que se abre, confirmamos su eliminación. (Ver Figura C.4).

C.6. Gestión de servicios

Antes de añadir un nuevo servicio debemos tener la familia a la que pertenece el servicio y el tipo de IVA que tendrá éste.

C.6.1. Nueva familia de servicios

Añadir nueva familia de servicios

Introducimos el nombre de la familia de servicios, y una descripción si se desea. Para añadirlo pulsamos sobre el botón «Nuevo».

Modificar familia de servicios

Para modificar una familia de servicios, seleccionamos la familia de servicio, modificamos el campo que se requiere y pulsamos en «Modificar».

C.6.2. Nuevo servicio

Para añadir un nuevo servicio en el sistema, el proceso a seguir es el mismo que el proceso para añadir un nuevo cliente, (Véase Sección C.3.1).

C.6.3. Filtrar servicios

Para filtrar un servicio, el proceso a seguir es el mismo que el proceso para filtrar un cliente, (Véase Sección C.3.3).

The screenshot shows a software window titled "Familia Servicio Nuevo". At the top, there are navigation buttons: "Nuevo" (with a document icon) and "Modificar" (with a floppy disk icon). Below these is a table with two columns: "Familia" and "Descripción". The table contains two rows of data: "Vacuna" and "Peluquería". Below the table is a large empty rectangular area. At the bottom of the window, there is a section titled "Familia servicio nuevo" which contains two input fields: "Familia:" and "Descripcion:". The "Descripcion:" field is a larger text area with a small icon in the bottom right corner.

Familia	Descripción
Vacuna	Vacuna
Peluquería	Peluquería

Familia servicio nuevo

Familia:

Descripcion:

Figura C.17.: Nueva familia de servicios

C.6.4. Modificar servicio

Para modificar los datos de un servicio, los pasos a seguir son los mismos que para «Modificar producto», (Véase Sección C.5.3).

C.6.5. Eliminar servicio

Para eliminar un servicio:

1. Hacemos click sobre el cliente que deseamos eliminar.
2. Se habilita el botón Eliminar.
3. En la ventana emergente que se abre, confirmamos su eliminación.

C.7. Gestión de proveedores

La gestión de proveedores se accede pulsando sobre el botón «*Proveedores*» del menú principal.

C.7.1. Nuevo proveedor

Para añadir un nuevo proveedor en el sistema, el proceso a seguir es el mismo que el proceso para añadir un nuevo cliente, (Véase Sección C.3.1).

C.7.2. Filtrar proveedores

Para filtrar un proveedor en el sistema, el proceso a seguir es el mismo que el proceso para filtrar un cliente, (Véase Sección C.3.3).

C.7.3. Modificar proveedor

Para modificar los datos de un proveedor, debemos hacer:

1. Doble click sobre la tupla de este en el listado de proveedores.
2. Se abre una ventana con los datos del proveedor. (Véase Figura C.18).
3. Modificamos los campos necesarios. Si uno de los campos es Provincia o Población, hacemos un click sobre el campo y entonces éste se convertirá en un seleccionable.
4. Elegimos la provincia y la población.
5. Por último, pulsamos en el botón «*Guardar*» para almacenar los cambios.

Proveedor Modificar

Guardar

Datos del Proveedor	
CIF *	A22222256
Nombre *	Bayer
Dirección *	Medicamentos
Provincia *	Cádiz
Población *	Cádiz
Teléfono *	956325117
Teléfono 2	
Fax	
Email *	bayer@bayer.es
Contacto	

Figura C.18.: Proveedor modificar

C.7.4. Eliminar proveedor





Para eliminar un proveedor del sistema, el proceso a seguir es el mismo que el proceso para eliminar un cliente, (Véase Sección C.3.2).

C.8. Gestión de pedidos

Para realizar un pedido, accedemos pulsando en «*Proveedores>Pedidos*» y además previamente debe existir en el sistema el proveedor al que vayamos a hacer el pedido (Véase Sección C.7.1) y por supuesto los productos asociados al proveedor (Véase Sección C.5.2).

C.8.1. Nuevo pedido

Para realizar un pedido, accedemos pulsando en «*Proveedores>Pedidos*» y además previamente debe existir en el sistema el proveedor al que vayamos a hacer el pedido (Véase Sección C.7.1) y por supuesto los productos asociados al proveedor (Véase Sección C.5.2).

1. Un nuevo pedido lo iniciamos buscando el proveedor al que vayamos a realizar el pedido, para ello pulsamos sobre el botón  del campo «*Proveedor*».
2. En la venta emergente que se abra, buscamos el proveedor y pulsamos a continuación sobre el botón «*Enviar*».
3. Observamos como la ventana se cierra automáticamente y ahora en el campo «*Proveedor*» aparece el nombre de éste.
4. Pulsamos sobre el botón «*Nuevo*» y nos llevará a una nueva ventana, que será la del pedido.
5. Para añadir un nuevo producto al pedido, debemos pulsar sobre el botón  del campo Código.
6. Se abre una ventana con los productos asociados al Proveedor.
7. Elegimos los productos que llevará el pedido, pulsando sobre el checkbox  y vemos como se activa .
8. Cuando ya hayamos seleccionado todos los productos, pulsamos en «*Confirmar*».

Si deseamos modificar la cantidad de un producto, ya que de forma predeterminada su cantidad es 1:

1. Seleccionamos el producto que deseamos modificar su cantidad.
2. En el campo Cantidad, indicamos el nuevo valor.
3. Pulsamos [INTRO] en el teclado o seleccionando en cualquier otro sitio de la página y se guardarán los cambios.

C.8.2. Modificar pedido

Para modificar un pedido, debemos estar dentro del pedido, haciendo doble click en el listado de pedidos.

Una vez que estemos dentro, si el cambio es en la parte superior, que corresponde a la información general del pedido, no es necesario hacer ninguna interacción, ya que el sistema guarda automáticamente el cambio.

Si deseamos modificar una línea del pedido, es decir, un producto, basta con seleccionarlo y realizar los cambios oportunos.

C.8.3. Filtrar pedidos

Para filtrar un pedido en el sistema, el proceso a seguir es el mismo que el proceso para filtrar un cliente, (Véase Sección C.3.3).

C.8.4. Eliminar pedido

Para eliminar un pedido del sistema, el proceso a seguir es el mismo que el proceso para eliminar un cliente, (Véase Sección C.3.2).





C.9. Gestión de ventas

Para realizar una nueva venta, previamente debe existir en el sistema el producto que vayamos a vender (Véase Sección C.7.1) y por supuesto los productos asociados al proveedor (Véase Sección C.5.2).

C.9.1. Nueva venta

Para realizar una venta, accedemos pulsando en «*Ventas>Nueva Venta*» y además previamente debe existir en el sistema el producto o servicio que vayamos a vender.


1. La venta comienza con la siguiente ventana Figura C.19.

2. Si el cliente está registrado en el sistema, pulsamos en el botón  del campo «Cliente» y seleccionamos el cliente, si no es el caso se considerará como *Venta Rápida*.
3. Si el empleado tuviera el rol de Veterinario, en el campo *Veterinario* se mostraría su nombre, si no es así, el empleado puede seleccionar al veterinario que realizó el servicio.
4. Para añadir un nuevo producto o servicio a la venta, debemos pulsar sobre el botón  del campo Código respectivo.
5. Se abre una ventana con los productos o servicios disponibles.
6. Elegimos los artículos que se van a vender, pulsando sobre el checkbox  y vemos como se activa .
7. Cuando ya hayamos seleccionado todos los productos, pulsamos en «*Confirmar*».
8. Se añaden los artículos a la venta.

Si deseamos modificar la cantidad de un artículo, ya que de forma predeterminada su cantidad es 1:

1. Seleccionamos el artículo que deseamos modificar.
2. Modificamos el campo correspondiente indicando el nuevo valor.
3. Pulsamos [INTRO] en el teclado o seleccionando en cualquier otro sitio de la página y se guardarán los cambios.

C.9.2. Generar albarán

Si deseamos obtener el albarán de la venta, debemos pulsar sobre el botón *Generar Albarán*  **Generar Albarán** y se abrirá una ventana de descarga.

C.9.3. Modificar venta

Para modificar una venta, debemos estar dentro de la venta, haciendo doble click en el listado de ventas.

Una vez que estemos dentro, si el cambio es en la parte superior, que corresponde a la información general de la venta, no es necesario hacer ninguna interacción, ya que el sistema guarda automáticamente el cambio.

Si deseamos modificar una línea de la venta, basta con seleccionarlo y realizar los cambios oportunos.

Inicio

Clientes

Mascotas

Productos

Servicios

Ventas

Proveedores

Estadísticas

Configuración

Agustín (Administrador)

Atrás

Generar Albarán

Eliminar

Venta Descripción

Número: 62

Cliente: Venta Rápida

Fecha: 16-mar-2013

Veterinario:

Código	Nombre	Tipo	Descripción	Cantidad	PVP	IVA	Precio total

Servicio

Código:

Nombre:

Producto

Código:

Nombre:

Cantidad: 0

Precio: 0

IVA: 0

Total:

0 €

Figura C.19.: Venta nueva

226

C.9.4. Filtrar venta

Para filtrar las ventas en el sistema, el proceso a seguir es el mismo que el proceso para filtrar un cliente, (Véase Sección C.3.3).


C.9.5. Eliminar venta

Para eliminar una venta del sistema, el proceso a seguir es el mismo que el proceso para eliminar un cliente, (Véase Sección C.3.2).

C.10. Gestión de facturas


C.10.1. Nueva factura

Para crear una nueva factura, accedemos pulsando en «Ventas>Facturas», previamente debe existir en el sistema el cliente sobre el que vayamos a hacer la factura.

1. Una nueva factura la iniciamos buscando el cliente al que vayamos a realizar la factura, para ello pulsamos sobre el botón  del campo «Cliente».
2. En la venta emergente que se abra, buscamos el cliente y pulsamos a continuación sobre el botón «Enviar».
3. Observamos como la ventana se cierra automáticamente y ahora en el campo «Cliente» aparece el nombre de éste.
4. Pulsamos sobre el botón «Nuevo» y nos llevará a una nueva ventana, que será la de la factura. (Véase Figura C.20).

Total en el campo Total aparece la suma total de la factura, incluido el importe del IVA.

IVA en el campo IVA aparece el valor del IVA asociado a la factura.

5. Para añadir una nueva venta a la factura, debemos pulsar sobre el botón  del campo Venta.
6. Se abre una ventana con las ventas no facturadas asociadas al cliente en curso.
7. Elegimos las ventas que llevará la factura, pulsando sobre el checkbox ☐ y vemos como se activa ☒.
8. Cuando ya hayamos seleccionado todas las ventas, pulsamos en «Confirmar».

Inicio

Clientes

Mascotas

Productos

Servicios

Ventas

Proveedores

Estadísticas

Configuración

Agustín (Administrador)

Altas

Generar Factura

Descargar Factura

Eliminar

Venta Descripción

Número: 15

Cliente: Pepe García Rodríguez

Fecha: 16-mar-2013

Código	Nombre	Fecha	Tipo	Descripción	Cantidad	PVP	IVA %	Precio total
frontline 1m	Frontline 1ML	2012-11-05	Producto		2	4.13 €	21	9.99 €
ibu0001	ibuprofeno 1MG	2012-11-05	Producto		2	1.45 €	21	3.51 €
gasas 100gr	Gasas de 100 gramos	2012-12-20	Producto		2	0.55 €	21	1.33 €
collar001	Collares Grandes	2012-12-20	Producto		1	6.00 €	21	7.26 €
inizol10ml	Imizol 10 ML	2012-12-21	Producto		3	3.01 €	21	10.94 €
vacrab001	Rabia	2013-02-20	Servicio	Vacuna de la Rabia	1	10.00 €	21	12.10 €

Venta

9

Total: 45.13 €

IVA: 7.83 €

Figura C.20.: Factura línea

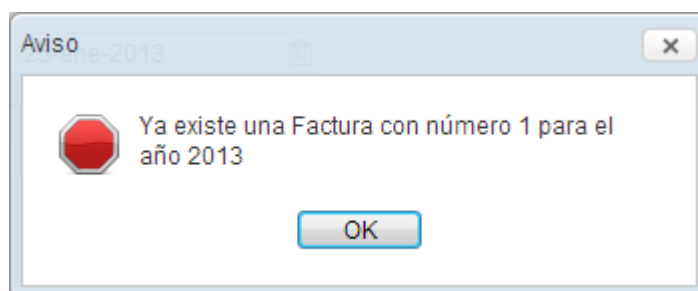


Figura C.21.: Factura mensaje alerta

C.10.2. Generar factura

Si deseamos obtener la factura en formato PDF, debemos pulsar sobre el botón *Generar Factura* y se abrirá una ventana de descarga.

C.10.3. Descargar factura

Si ya la factura está generada, pulsamos sobre el botón *Descargar Factura* y se abrirá una ventana de descarga.

C.10.4. Modificar factura

Para modificar una factura, debemos estar dentro de la factura, haciendo doble click en el listado de facturas.

- **Modificación de la fecha de la factura**, seleccionamos la fecha correcta. El sistema guarda los cambios automáticamente.
- **Modificación del número de la factura**, indicamos el número de la factura correcta. El sistema puede alertar con un mensaje emergente de error si el número indicado ya existe en el sistema para ese año fiscal. (Véase Figura C.21).

C.10.5. Filtrar factura

Para filtrar las facturas en el sistema, el proceso a seguir es el mismo que el proceso para filtrar un cliente, (Véase Sección C.3.3).

C.10.6. Eliminar factura

Para eliminar una factura del sistema, el proceso a seguir es el mismo que el proceso para eliminar un cliente, (Véase Sección C.3.2).



C.11. Consultar estadísticas

Pulsando en «Estadísticas» en el menú principal tenemos dos tipos de estadísticas: estadísticas sobre las mascotas y estadísticas de ingresos/gastos.

C.11.1. Estadísticas Mascota

Las estadísticas que nos encontramos serán las Defunciones, Nuevas Altas, Vacunas, Citas. Los datos están divididos por meses, dentro del año indicado en la parte superior, que de forma predeterminada será el año en curso. (Véase Figura C.22).


Operaciones disponibles:

- Consultar año: en el campo «*Año*» podemos indicar el año y a continuación pulsar en el botón «*Buscar*» o bien pulsar [INTRO] en el teclado.
- Imprimir gráfica: si deseamos imprimir la gráfica, debemos hacer click en . Tras pulsar se nos abrirá la ventana para imprimir.
- Descargar gráfica: si deseamos descargar la gráfica, debemos hacer click en . Tras pulsar se abrirá un menú con los distintos formatos de ficheros disponible: PNG (imagen), JPEG (imagen), PDF, SVG (imagen vectorial).

C.11.2. Estadísticas Ingresos/Gastos

Las estadísticas que nos encontramos serán los Ingresos, Gastos, Ingresos por Venta de artículos, Ingresos por Honorarios. Los datos están divididos por meses, dentro del año indicado en la parte superior, que de forma predeterminada será el año en curso. (Véase Figura C.23).

Operaciones disponibles:

- Consultar año: en el campo «*Año*» podemos indicar el año y a continuación pulsar en el botón «*Buscar*» o bien pulsar [INTRO] en el teclado.
- Imprimir gráfica: si deseamos imprimir la gráfica, debemos hacer click en . Tras pulsar se nos abrirá la ventana para imprimir.

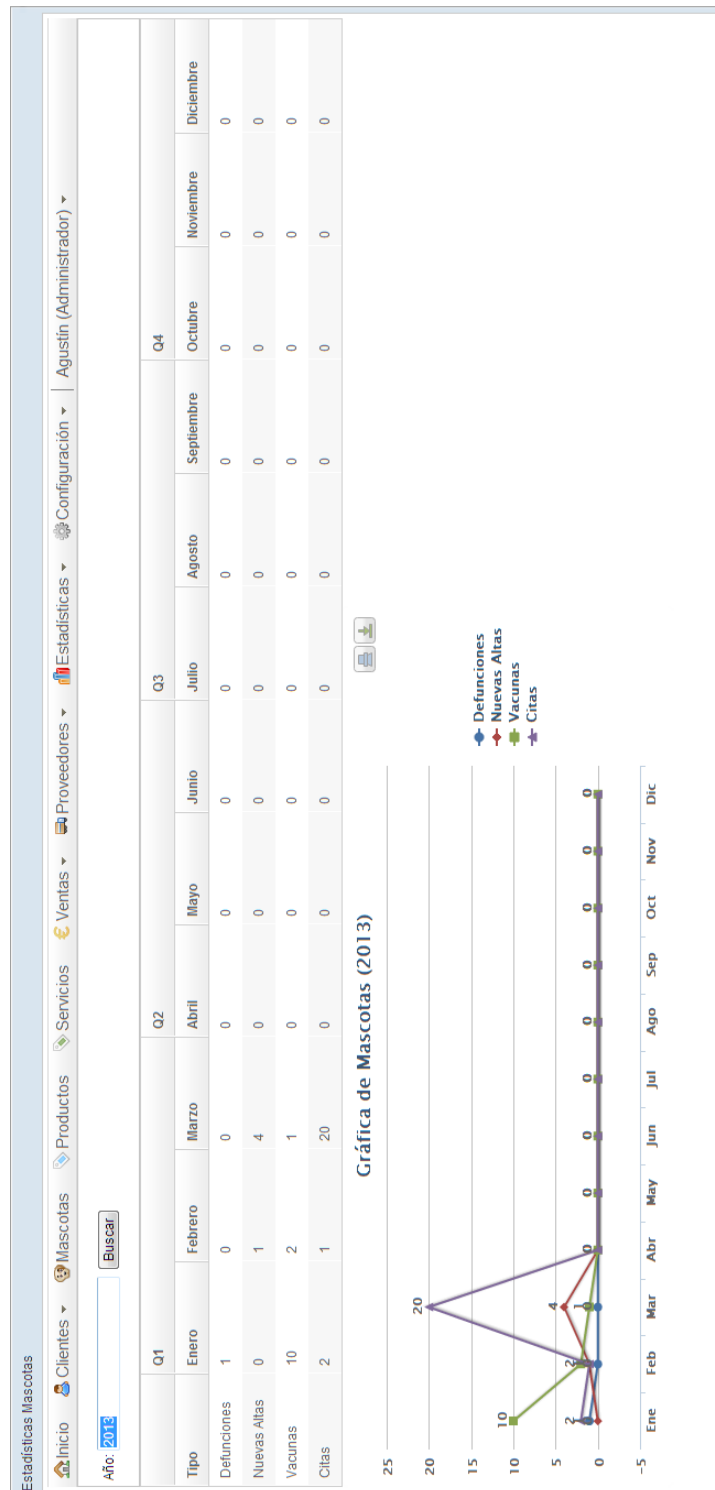



Figura C.22.: Estadísticas de Mascota

- Descargar gráfica: si deseamos descargar la gráfica, debemos hacer click en . Tras pulsar se abrirá un menú con las distintos formatos de ficheros disponible: PNG (imagen), JPEG (imagen), PDF, SVG (imagen vectorial).

C.12. Gestión de usuarios

Esta sección sólo está accesible para un usuario con el rol «*Administrador*». Para acceder pulsamos en «Configuración > Usuarios».

C.12.1. Nuevo usuario

1. Para añadir un nuevo usuario pulsamos sobre le botón «*Nuevo*».
2. Se abrirá una venta emergente. (Véase Figura C.24).
3. Rellenamos los campos obligatorios. Algunas consideraciones:
 - a) El Alias será el nombre de usuario para acceder al sistema.
 - b) La contraseña será temporal y el usuario cuando acceda al sistema deberá modificar la contraseña. (Véase Sección).
 - c) El Rol, existen 3 tipos: Administrador, Veterinario y Empleado.
4. Para guardar la información se pulsa sobre el botón «*Guardar*».

C.12.2. Modificar usuario

Para modificar a un usuario, basta con hacer doble click sobre él y entonces se abrirá una ventana con los datos del usuario.

Una vez se haya modificado se pulsa en el botón «*Guardar*» para almacenar los cambios en el sistema.

C.12.3. Eliminar usuario

Para eliminar un usuario y como medida de seguridad ampliada:

1. Seleccionamos el usuario que deseemos eliminar.
2. Hacemos click sobre el botón «*Modificar*» y a continuación se activa el botón «*Eliminar*».

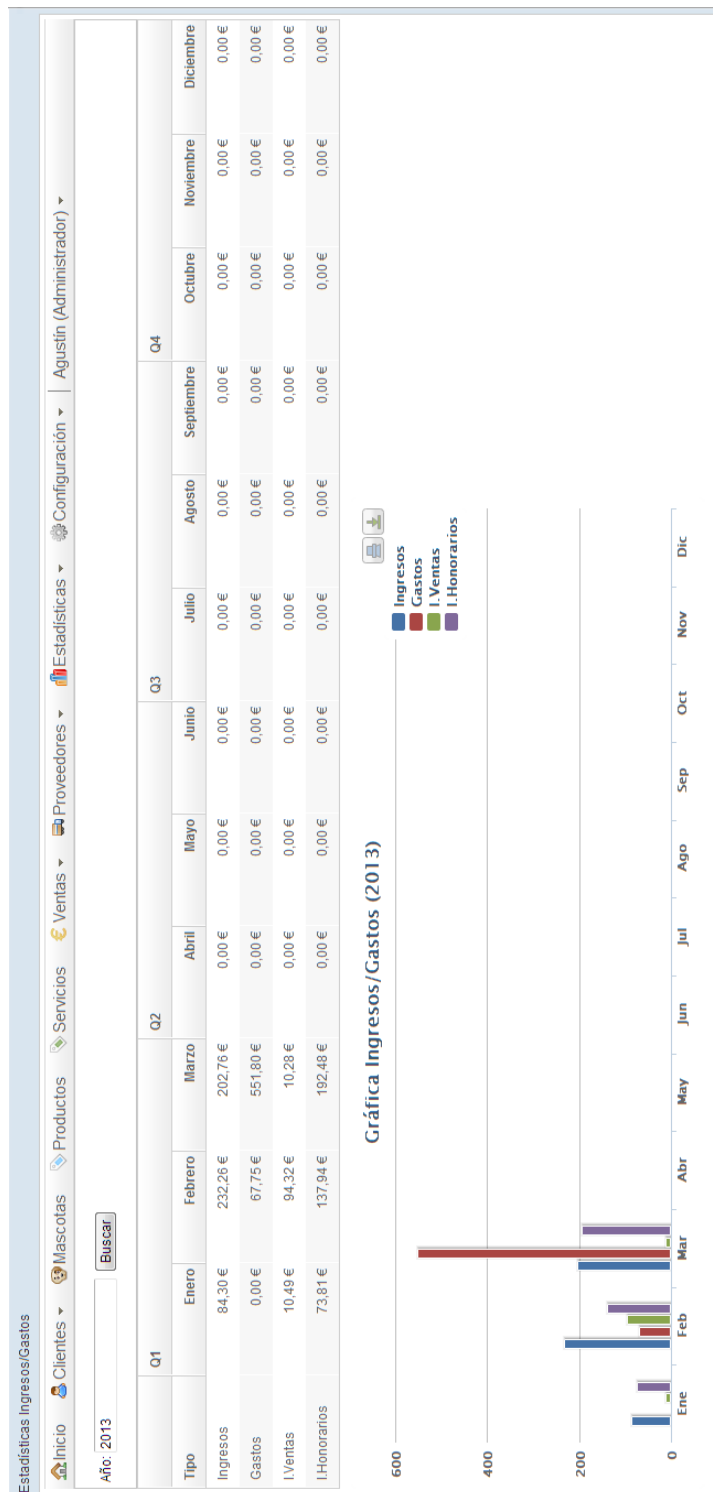
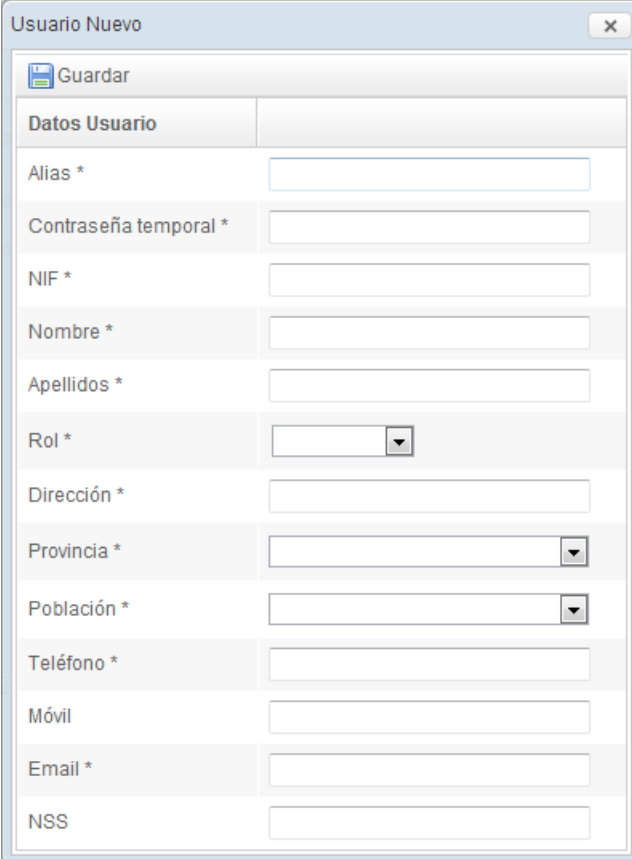


Figura C.23.: Estadísticas de Ingresos/Gastos




Usuario Nuevo	
 Guardar	
Datos Usuario	
Alias *	<input type="text"/>
Contraseña temporal *	<input type="text"/>
NIF *	<input type="text"/>
Nombre *	<input type="text"/>
Apellidos *	<input type="text"/>
Rol *	<input type="text"/>
Dirección *	<input type="text"/>
Provincia *	<input type="text"/>
Población *	<input type="text"/>
Teléfono *	<input type="text"/>
Móvil	<input type="text"/>
Email *	<input type="text"/>
NSS	<input type="text"/>

Figura C.24.: Usuario nuevo

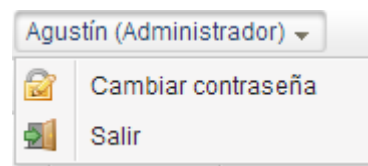


Figura C.25.: Usuario submenú

3. Pulsamos sobre el botón «*Eliminar*» y se abrirá una ventana de confirmación. Si el usuario tiene el rol *Administrador* no se podrá borrar, para ello se tiene que modificar el rol.
4. El usuario confirma y el sistema lo elimina del sistema.

C.12.4. Modificar contraseña

Si deseamos modificar la contraseña pulsamos sobre nuestro nombre en el menú principal y se abrirá un submenú con las opciones de «*Modificar contraseña*» y «*Salir*». (Véase Figura C.25).

1. Pulsamos sobre «*Modificar contraseña*» y se abrirá la ventana correspondiente. (Véase Figura C.26).
2. Debemos introducir la «Antigua contraseña», si no fuera correcta se deshabilitará el resto de campos.
3. La nueva contraseña a introducir debe contener de 8-20 caracteres con dígitos 0-9, caracteres en mayúsculas A-Z y minúsculas de a-z.
4. Se debe repetir la nueva contraseña y si coinciden se podrá modificar pulsando en «Guardar».

Cambiar contraseña

Guardar

Datos usuario

Usuario	Agustín
Antigua contraseña	
Nueva contraseña	
Repetir nueva contraseña	

Figura C.26.: Usuario modificar contraseña

D. Documentos generados por el sistema

A continuación se presenta un ejemplo de albarán generado por nuestra aplicación, el cual sigue el formato establecido e impuesto por el cliente (Véase Figura D.1).

En la Figura D.2 podemos observar una factura generada por nuestra aplicación, en la que se muestra el logo original de la clínica veterinaria, datos de contacto y los datos generales de las facturas: número, fecha, datos del cliente, artículos, coste del IVA, coste total, etc.

ALBARÁN N° 40		16 / 12 / 2012		
SERVICIOS MÉDICOS-VETERINARIOS				
Cliente: Pepe García Rodríguez				
Teléfono: 954321245				
CANT.	CONCEPTO	DESCRIPCIÓN	PRECIO	IMPORTE
1	Collares Grandes		6,00 €	6,00 €
3	Imizol 10 ML		3,00 €	9,00 €
2	Gasas de 100 gramos		0,55 €	1,10 €
			SUMA	16,10 €
			IVA 21%	3,38 €
			TOTAL	19,48 €

Figura D.1.: Ejemplo de albarán



Centro Veterinario "El Rey de la Casa"

Luis Miguel Puerto del Canto

NIF: 25.594.942-J

C/ Comandante Vázquez Ramos, 10

Montellano (Sevilla)

Teléfono: 626 31 06 64

FACTURA

15

FECHA

16 / 03 / 2013

CLIENTE

Nombre: Pepe García Rodríguez

NIF/CIF: 12345678P

Dirección: Calle Puerto 17

Ciudad: Dos Hermanas, 41700, (Sevilla)

Teléfono: 954321245

CANT.	FECHA	CONCEPTO	DESCRIPCIÓN	PRECIO	IMPORTE
1	20-12-2012	Collares Grandes		6,00 €	6,00 €
3	21-12-2012	Tratamiento		3,00 €	9,00 €
2	20-12-2012	Gasas de 100 gramos		0,55 €	1,10 €
1	20-02-2013	Rabia	Vacuna de la Rabia	10,00 €	10,00 €
2	05-11-2012	Tratamiento		1,45 €	2,90 €
2	05-11-2012	Frontline IML		4,00 €	8,00 €
				SUMA	37,00 €
				IVA 21%	7,77 €
				TOTAL	44,77 €

Figura D.2.: Ejemplo de factura

D. Documentos generados por el sistema

Bibliografía

- [1] Microsoft: Introduction to Model-View-ViewModel pattern for building WPF apps (Oct 2005), <http://blogs.msdn.com/b/johngossman/archive/2005/10/08/478683.aspx>
- [2] Highcharts - Interactive JavaScript charts for your webpage (Oct 2012), <http://www.highcharts.com/>
- [3] iText - Free / Open Source PDF Library for Java (Sep 2012), <http://itextpdf.com/>
- [4] MySQL 5.5 Reference Manual (May 2012), <http://dev.mysql.com/doc/refman/5.5/en/index.html>
- [5] Página oficial de Apache Flex (May 2012), <http://flex.apache.org/>
- [6] Página oficial de Dia (Jul 2012), <http://projects.gnome.org/dia/>
- [7] Página oficial de GIMP (May 2012), <http://www.gimp.org/>
- [8] Página oficial de Google Web Toolkit (May 2012), <https://developers.google.com/web-toolkit/>
- [9] Página oficial de LyX (Jul 2012), <http://www.lyx.org/>
- [10] Página oficial de Microsoft Silverlight (May 2012), <http://www.microsoft.com/silverlight/>
- [11] Página oficial de NetBeans (May 2012), <http://netbeans.org/>
- [12] REM - A NetBeans Module for ZK (Apr 2012), <http://sourceforge.net/projects/rem1/>
- [13] ZHighCharts Demo (Oct 2012), <http://zhighcharts.appspot.com/demo.zul>
- [14] ZK - Leading Enterprise Java Web Framework (May 2012), <http://www.zkoss.org>
- [15] TIOBE Programming Community Index for November 2012 (Mar 2013), <http://www.tiobe.com/index.php/content/paperinfo/tpci/index.html>

Bibliografía

- [16] Booch: El Lenguaje Unificado De Modelado. Pearson (2006)
- [17] Goodwill, A.V.J.: Apache Tomcat 7
- [18] IEEE: IEEE 829 (Dec 2012), http://en.wikipedia.org/wiki/IEEE_829
- [19] O’Sullivan, B.: Mercurial: The Definitive Guide (Jul 2009), <http://hgbook.red-bean.com/>
- [20] P.J. Deitel, H.D.: Java: cómo programar. Prentice Hall (2008)