



Proyecto de Desarrollo de Aplicaciones Multiplataforma

Elia Baladrón Peral

Desarrollo de Aplicaciones Multiplataforma
2020-2021

Tutor: José Alberto Blanco Hernández

Presentación:

0.1. Portada

0.2. Agradecimientos, dedicatorias, ...

0.3. Índice del documento.

Presentación:.....	1
0.1. Portada	1
0.2. Agradecimientos, dedicatorias,	1
0.3. Índice del documento.	1
0.4. Listas de imágenes.	3
0.5. Listas de tablas.	3
1. Estudio del problema y análisis del sistema.....	5
1.1. Introducción.	5
1.2. Descripción de la aplicación. Motivación. Finalidad.	5
1.3. Objetivos	6
Objetivos funcionales.....	6
Objetivos técnicos	7
Objetivos personales.....	7
2. Recursos necesarios para el desarrollo.....	8
2.1. Recursos humanos.	8
2.2. Recursos hardware.....	8
2.3. Recursos software.	8
3. Planificación	9
3.1. Metodología para el diseño del proyecto.	9
3.2. Tabla de actividades, cálculo de tiempos y diagrama de Gantt.....	9
3.3. Secuencia de desarrollo del proyecto	10
4. Desarrollo del proyecto.....	16
4.1. Secuencia real del desarrollo del proyecto	16
4.2. Diagrama relacional. Diagramas de clases.	16

4.2.1.	Diagrama de clases.....	16
4.2.2.	Diagrama BD.....	23
4.3.	Interacción con el usuario	23
4.3.1	Casos de uso	23
4.3.2.	Diseño de la interfaz.....	26
	Áreas.....	27
	Funcionalidades botones.....	34
4.4	Código fuente.	35
	Datos genéricos	35
	Estructura de la aplicación	35
5.	Fase de pruebas.....	39
5.3	Pruebas realizadas.....	39
6.	Conclusiones finales:	41
6.1.	Reflexión de los resultados obtenidos.	41
6.2.	Grado de cumplimiento de los objetivos fijados.....	41
6.3	Propuesta de modificaciones o ampliaciones futuras del sistema	41
7.	Dificultades y problemas fundamentales encontrados.	42
8.	Documentación del sistema desarrollado.....	43
8.3	Manual de instalación.	43
8.4	Manual de uso.....	43
9.	Bibliografía:.....	44
9.3	Incluirá toda la documentación consultada: libros, apuntes, páginas webs, foros, etc. .	44
10.	Anexos.	45
Anexo I.	Diagrama de Gantt.	45
Anexo II.	Casos de uso	50
	Animales	50
	Compraventa	51
	Crotales.....	52
	Datos.....	52
	Rebaño.....	53
	Veterinario.....	53
Anexo III.	Manual de instalación	53
Anexo IV.	Manual de uso	54
	¿Cómo ver los datos guardados?	54
	¿Cómo añadir datos?.....	57
	¿Cómo editar los datos?.....	58

¿Cómo eliminar los datos?	58
---------------------------------	----

0.4. Listas de imágenes.

i. Diagrama de Entidad-Relación.	11
ii. La interfaz contendrá un menú lateral con las áreas generales de la aplicación.....	12
iii. Las áreas de la interfaz estarán subdividas en pestañas.....	12
iv. Diagrama de clases.....	15
v. Diagrama de las clases Activity.	17
vi. Diagrama común clases gestoras de la BD.....	18
vii. Diagrama de las ClasesVO.	19
viii. Diagrama común Fragment.....	20
ix. Diagrama común GridAdapter.	20
x. Diagrama común PagerAdapter.	21
xi. Diagrama MainActivity.....	22
xii. Diagrama BD.....	23
xiii. Casos de uso. General.	24
xiv. Casos de uso. Añadir.	24
xv. Casos de uso. Editar.	25
xvi. Casos de uso. Eliminar.....	26
xvii. Menú lateral con las áreas generales de la aplicación.....	27
xviii. Pestaña Animales	28
xix. Detalle interfaz animales.....	28
i. Detalle interfaz terneros.	29
ii. Detalle interfaz vacas.	30
iii. Pestañas crotales.....	30
iv. Detalle interfaz crotal.....	31
v. Pestaña rebaños.....	31
vi. Detalle interfaz rebaño.	31
vii. Pestaña controles veterinarios.....	32
viii. Detalle interfaz detalles controles veterinarios.....	32
ix. Pestaña visitas veterinarias.....	32
x. Detalle interfaz detalles visitas veterinarias.	33
xi. Pestaña ventas.	33
xii. Detalle interfaz detalles compraventa.....	34

0.5. Listas de tablas.

i. Casos de uso. Añadir.	25
ii. Casos de uso. Editar.	25
iii. Casos de uso. Eliminar.....	26
iv. Pruebas realizadas.....	40

1. Estudio del problema y análisis del sistema

1.1. Introducción.

La utilidad de la aplicación será la gestión y organización de una explotación ganadera, basándose en los datos guardados sobre la misma.

La aplicación será desarrollada en Android Studio, entorno de desarrollo utilizado en el módulo Programación Multimedia y Dispositivos Móviles utilizando lenguaje Java, utilizado en múltiples módulos del ciclo.

1.2. Descripción de la aplicación. Motivación.

Finalidad.

Se guardarán los datos de cada uno de los animales de la explotación, pudiendo añadir posteriormente datos como las fechas de embarazos y partos para facilitar la organización. Se tendrá un registro de diferentes lugares donde podrán estar los animales, para tener un registro de que animal está en cada. También se podrá guardar a que animales les faltan los crotales, para saber cuáles deben ser pedidos, un registro de los tratamientos veterinarios y de las compras y ventas de los animales. También se podrán importar y exportar los datos y ver estadísticas de diferentes datos guardados en la aplicación, como el tiempo de cada animal en las diferentes localizaciones, el tiempo entre partos, el tiempo medio de destete de los animales o el precio medio de los animales.

1.3. Objetivos

Objetivos funcionales

Gestión de los animales

- Añadir animales.
- Modificar animales.
- Dar de baja animales, sin eliminarlos de la base de datos.
- Eliminar animales.

Gestión de la compraventa

- Ventas:
- Preparar un modelo con los datos del animal necesarios para realizar la venta.
- Añadir la venta en el registro de las ventas.
- Dar de baja el animal o eliminarlo de la base de datos.
- Compras:
- Añadir la compra al registro de compras.
- Dar de alta los animales comprados.

Gestión de los rebaños

- Creación de lugares dónde se pueden añadir los animales.
- Relacionar los animales con uno de los lugares, pudiendo cambiarlo con el tiempo.

Gestión de los crotales

- Control de los pendientes que faltan, para saber cuáles deben pedirse.

Gestión del veterinario

- Guardar los datos de:
- Los controles veterinarios, realizados o previstos.
- Las visitas del veterinario por enfermedad de los animales

Gestión de los datos

- Exportar datos de un animal o de todos los animales.

- Importar datos de uno o varios animales.
- Crear estadísticas de:
- Dinero invertido o ganado con la compraventa de animales.
- Media de tiempo hasta el destete de los animales.

Objetivos técnicos

- Fácil uso y aprendizaje del funcionamiento de la aplicación.
- Integración de todas las áreas de la aplicación para poder hacer la misma tarea desde distintas áreas, como realizar las bajas desde las ventas.
- Aplicación desarrollada con Android Studio en lenguaje Java, con una base de datos de SQLite.

Objetivos personales

- Mejorar los conocimientos tanto en el lenguaje Java como en el entorno de Android Studio.
- Aprender la utilización de SQLite.

2. Recursos necesarios para el desarrollo

2.1. Recursos humanos.

2.2. Recursos hardware.

Ordenador con Windows 10 y 16GB de RAM.

2.3. Recursos software.

- Android Studio.
- Draw.io para realizar diagramas.
- Base de datos SQLite.
- Monday.com para la realización del diagrama de Gantt.

3. Planificación

3.1. Metodología para el diseño del proyecto.

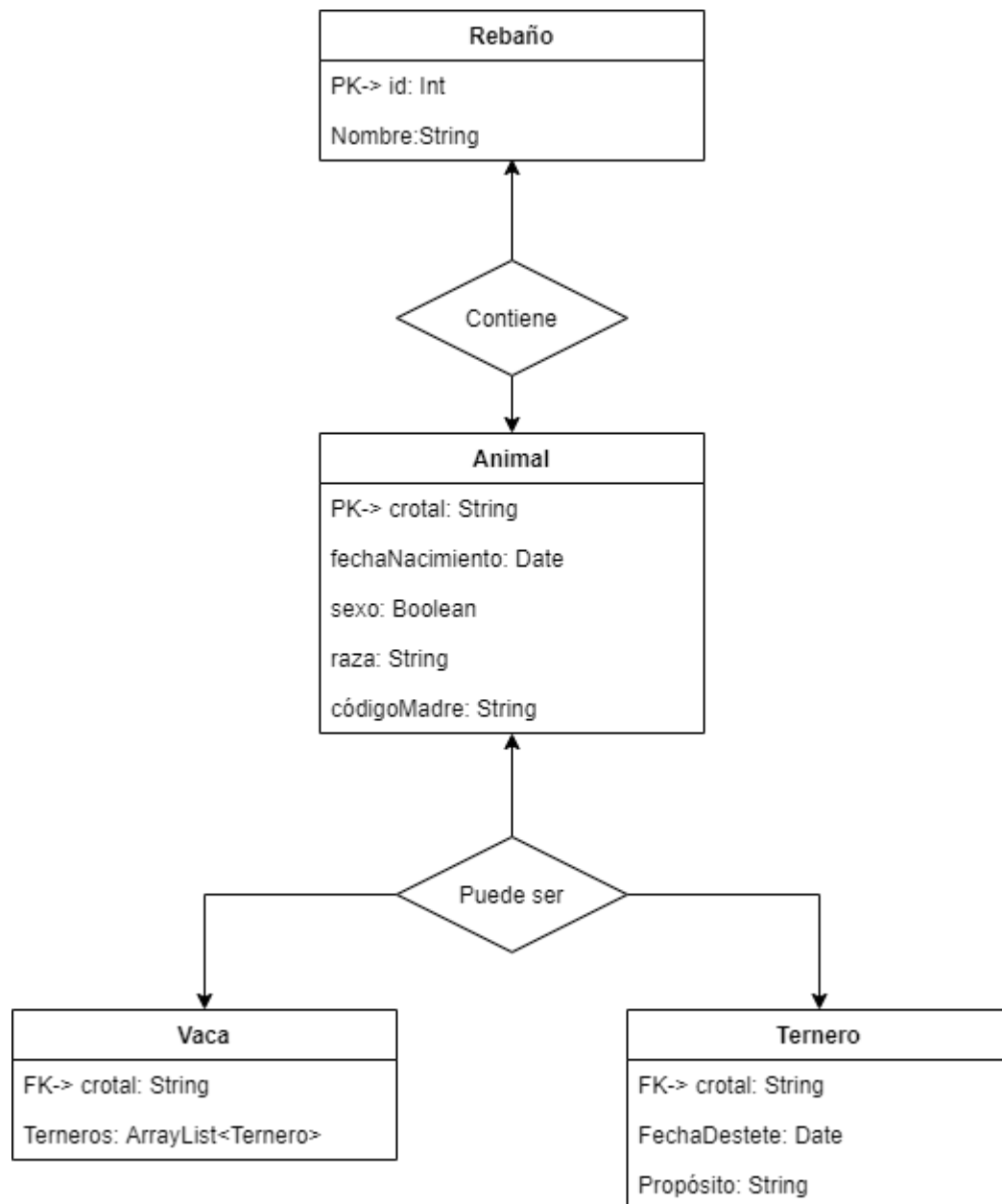
El tipo de metodología a utilizar es el método incremental.

3.2. Tabla de actividades, cálculo de tiempos y diagrama de Gantt

Incluido como Anexo I.

3.3. Secuencia de desarrollo del proyecto

Entidad relación



Controles
PK-> id: Int
Título: String
Descripción: String
Fecha: Date
ListaAnimales: String[]
Nº registro: String

Visitas
PK-> id: Int
Título
Fecha: Date
Descripción
ListaAnimales: String[]

Faltan
PK-> crotal: String
unidades: int

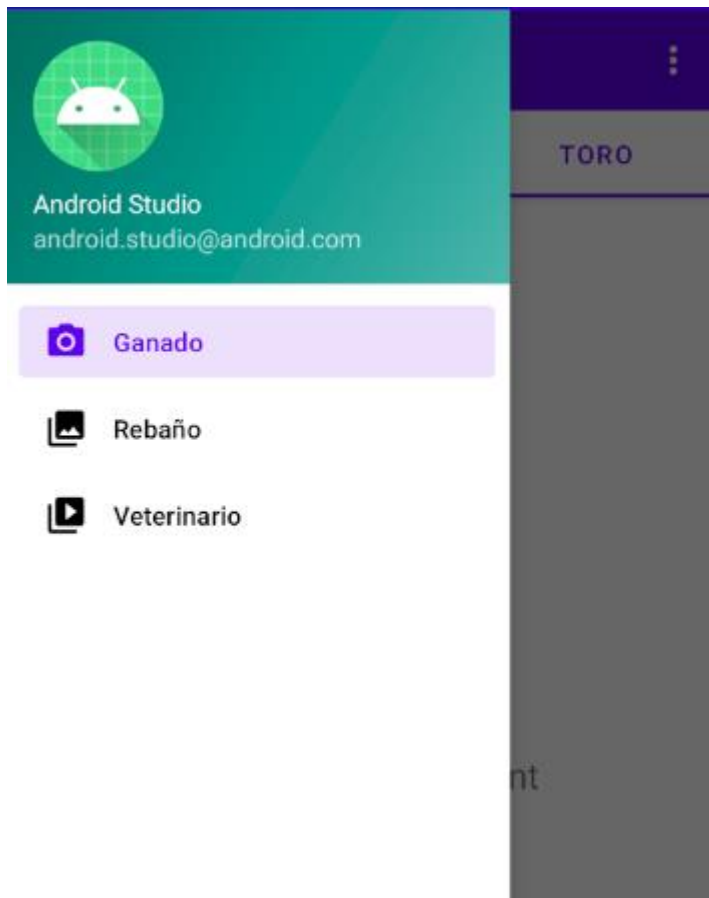
Pedidos
PK-> crotal: String
unidades: int

Recibidos
PK-> crotal: String
unidades: int

SinPoner
PK-> crotal: String
unidades: int

- i. Diagrama de Entidad-Relación.

Prototipo

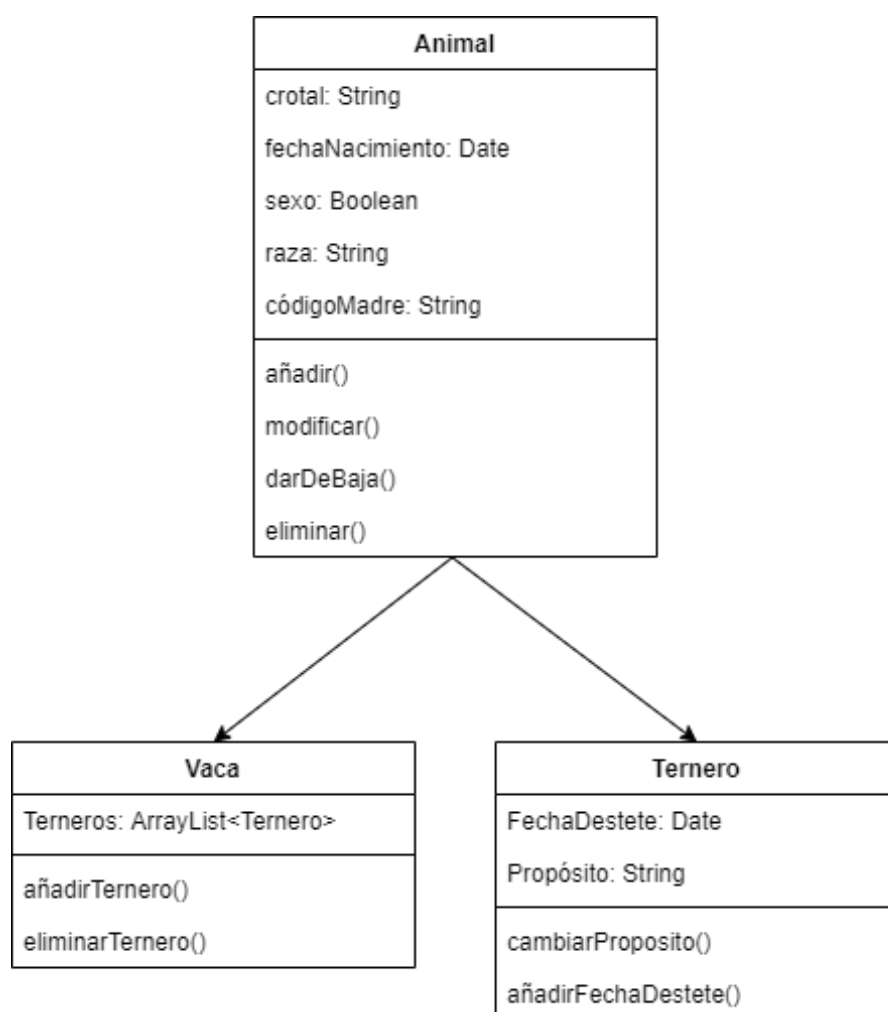


- ii. La interfaz contendrá un menú lateral con las áreas generales de la aplicación.



- iii. Las áreas de la interfaz estarán subdividas en pestañas

Diagrama de clases



Ventas
crotal: String precio: Double fecha: Date
crearModelo () añadirVenta() darDeBajaOEliminar ()

Compras
crotal: String precio: Double fecha: Date
añadirCompra() darDeAlta ()

Veterinario
controles: ArrayList<String> visitas: ArrayList<String>

Controles
Titulo: String Descripción: String Fecha: Date ListaAnimales: String[] Nº registro: String
añadir() cambiar() eliminar()

Visitas
Titulo: String Descripción: String Fecha: Date ListaAnimales: String[]
cambiar() eliminar() añadir()

Rebaño
Nombre:String
Animales: ArrayList<Animal>
añadir() eliminar() cambiar()

Crotales
faltan: ArrayList<String>
pedidos: ArrayList<String>
recibidos: ArrayList<String>
sinPoner: ArrayList<String>
añadirFaltan() eliminarFaltan() añadirPedidos() eliminarPedidos() añadirSinPoner() eliminarSinPoner() cambiarDeArray()

Datos
Titulo
Descripcion
importar() exportar() exportar(idAnimal) crearEstadísticasDinero() crearEstadísticasDestete()

iv. Diagrama de clases.

Casos de usos

Incluido como anexo en el Anexo II.

4. Desarrollo del proyecto

4.1. Secuencia real del desarrollo del proyecto

Incluido como Anexo I.

4.2. Diagrama relacional. Diagramas de clases.

4.2.1. Diagrama de clases

Activity

Animal	Animal_Toro	Animal_Ternero	Animal_Vaca
Animal animal	Animal animal	Ternero ternero	Vaca vaca
boolean editar	boolean editar	boolean editar	boolean editar
EditText viewNombre	EditText viewNombre	EditText viewNombre	EditText viewNombre
EditText viewCrotal	EditText viewCrotal	EditText viewCrotal	EditText viewCrotal
EditText viewNacimiento	EditText viewNacimiento	EditText viewNacimiento	EditText viewNacimiento
EditText viewSexo	EditText viewSexo	EditText viewSexo	EditText viewSexo
EditText viewRaza	EditText viewRaza	EditText viewRaza	EditText viewRaza
EditText viewMadre	EditText viewMadre	EditText viewMadre	EditText viewMadre
EditText viewReb	EditText viewReb	EditText viewReb	EditText viewReb
FloatingActionButton fab_editar	FloatingActionButton fab_editar	EditText viewDestete	EditText viewEmbarazo
FloatingActionButton fab_eliminar	FloatingActionButton fab_eliminar	EditText viewProposito	GridView gridView
FloatingActionButton fab_volver	FloatingActionButton fab_volver	FloatingActionButton fab_editar	FloatingActionButton fab_editar
FloatingActionButton fab_aceptar	FloatingActionButton fab_aceptar	FloatingActionButton fab_eliminar	FloatingActionButton fab_eliminar
FloatingActionButton fab_cancelar	FloatingActionButton fab_cancelar	FloatingActionButton fab_volver	FloatingActionButton fab_volver
onCreate()	onCreate()	FloatingActionButton fab_aceptar	FloatingActionButton fab_aceptar
eliminar()	eliminar()	FloatingActionButton fab_cancelar	FloatingActionButton fab_cancelar
volver()	volver()	onCreate()	onCreate()
editar()	editar()	eliminar()	eliminar()
aceptar()	aceptar()	volver()	volver()
cancelar()	cancelar()	editar()	editar()
		aceptar()	aceptar()
		cancelar()	cancelar()

Compraventa_Compras Compra compra boolean editar EditText viewCrotal EditText viewPrecio EditText viewFecha FloatingActionButton fab_editar FloatingActionButton fab_eliminar FloatingActionButton fab_volver FloatingActionButton fab_aceptar FloatingActionButton fab_cancelar onCreate() eliminar() volver() editar() aceptar() cancelar()	Compraventa_Ventas Venta venta boolean editar EditText viewCrotal EditText viewPrecio EditText viewFecha FloatingActionButton fab_editar FloatingActionButton fab_eliminar FloatingActionButton fab_volver FloatingActionButton fab_aceptar FloatingActionButton fab_cancelar onCreate() eliminar() volver() editar() aceptar() cancelar()	Compraventa_Ventas_Realizadas Venta venta boolean editar EditText viewCrotal EditText viewPrecio EditText viewFecha FloatingActionButton fab_editar FloatingActionButton fab_eliminar FloatingActionButton fab_volver FloatingActionButton fab_aceptar FloatingActionButton fab_cancelar onCreate() eliminar() volver() editar() aceptar() cancelar()	Compraventa_Ventas_Planificadas Venta venta boolean editar EditText viewCrotal EditText viewPrecio EditText viewFecha FloatingActionButton fab_editar FloatingActionButton fab_eliminar FloatingActionButton fab_volver FloatingActionButton fab_aceptar FloatingActionButton fab_cancelar onCreate() eliminar() volver() editar() aceptar() cancelar()
Crotal_Faltan Crotal crotal boolean editar EditText viewCrotal EditText viewUnidades FloatingActionButton fab_editar FloatingActionButton fab_eliminar FloatingActionButton fab_volver FloatingActionButton fab_aceptar FloatingActionButton fab_cancelar onCreate() eliminar() volver() editar() aceptar() cancelar()	Crotal_Pedidos Crotal crotal boolean editar EditText viewCrotal EditText viewUnidades FloatingActionButton fab_editar FloatingActionButton fab_eliminar FloatingActionButton fab_volver FloatingActionButton fab_aceptar FloatingActionButton fab_cancelar onCreate() eliminar() volver() editar() aceptar() cancelar()	Crotal_Recibidos Crotal crotal boolean editar EditText viewCrotal EditText viewUnidades FloatingActionButton fab_editar FloatingActionButton fab_eliminar FloatingActionButton fab_volver FloatingActionButton fab_aceptar FloatingActionButton fab_cancelar onCreate() eliminar() volver() editar() aceptar() cancelar()	Crotal_SinPoner Crotal crotal boolean editar EditText viewCrotal EditText viewUnidades FloatingActionButton fab_editar FloatingActionButton fab_eliminar FloatingActionButton fab_volver FloatingActionButton fab_aceptar FloatingActionButton fab_cancelar onCreate() eliminar() volver() editar() aceptar() cancelar()
Rebaño Rebaño rebaño boolean editar EditText viewId EditText viewNombre FloatingActionButton fab_editar FloatingActionButton fab_eliminar FloatingActionButton fab_volver FloatingActionButton fab_aceptar FloatingActionButton fab_cancelar onCreate() eliminar() volver() editar() aceptar() cancelar()	Veterinario_Visitas Visitas visitas boolean editar EditText viewTitulo EditText viewDescripcion EditText viewFecha GridView gridView FloatingActionButton fab_editar FloatingActionButton fab_eliminar FloatingActionButton fab_volver FloatingActionButton fab_aceptar FloatingActionButton fab_cancelar onCreate() eliminar() volver() editar() aceptar() cancelar()	Veterinario_Controlos Controles controles boolean editar EditText viewTitulo EditText viewDescripcion EditText viewFecha GridView gridView FloatingActionButton fab_editar FloatingActionButton fab_eliminar FloatingActionButton fab_volver FloatingActionButton fab_aceptar FloatingActionButton fab_cancelar onCreate() eliminar() volver() editar() aceptar() cancelar()	

v. Diagrama de las clases Activity.

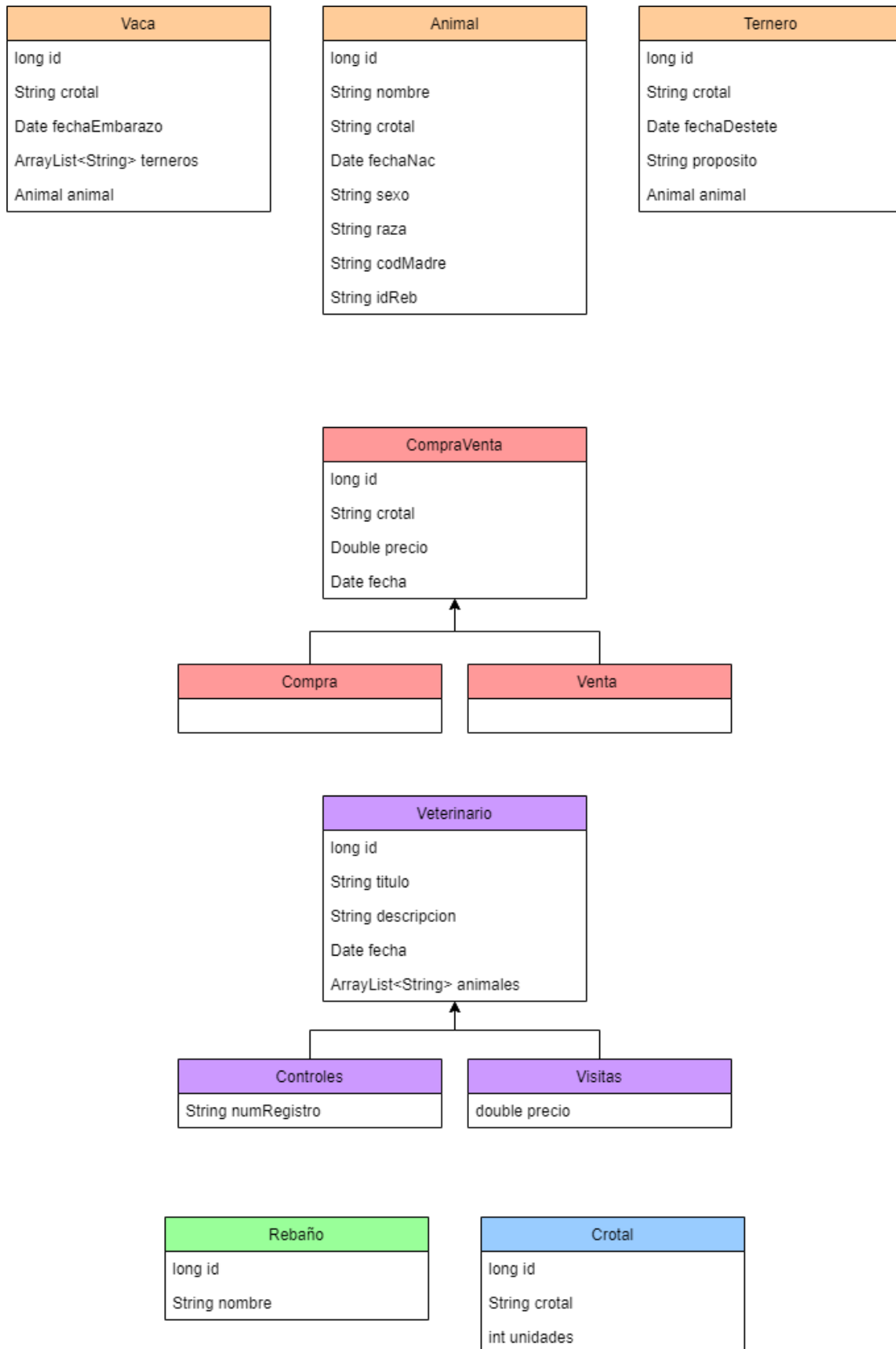
BD

Las clases que gestionan la base de datos tienen todas ellas la misma estructura.

BD
FeedReaderDbHelper_VacApp dbHelper
onCreate() onDestroy() insertarDatosPrueba() insertarDatos(Objeto objeto) getDatosCrotales() getDatosObjetos() borrarDatos(Objeto objeto) actualizar(Objeto objeto)

vi. Diagrama común clases gestoras de la BD.

ClasesVO



vii. Diagrama de las ClasesVO.

Fragment

Estructura común de los Fragment.

Fragment
GridView gridView FloatingActionButton fab_add MainActivity main ArrayList<Objeto> objetos
newInstance() onCreate() onCreateView() onStart() iniciarVista() actualizar(Objeto objeto) eliminar(Objeto objeto) añadir(Objeto objeto)

viii. Diagrama común Fragment.

GridAdapter

Estructura común de los GridAdapter.

GridAdapter
Context context ArrayList<Objeto> objetos
GridAdapter(Context c, ArrayList o) getCount() getItem(int pos) getItemId(int pos) getView()

ix. Diagrama común GridAdapter.

PagerAdapter

Estructura común de los PagerAdapter.

PagerAdapter
int tabsNUmber
MainActivity main
PagerAdapter(FragmentManager fm, int behavior, int tabs, MainActivity main)
getItem(int pos)
getCount()

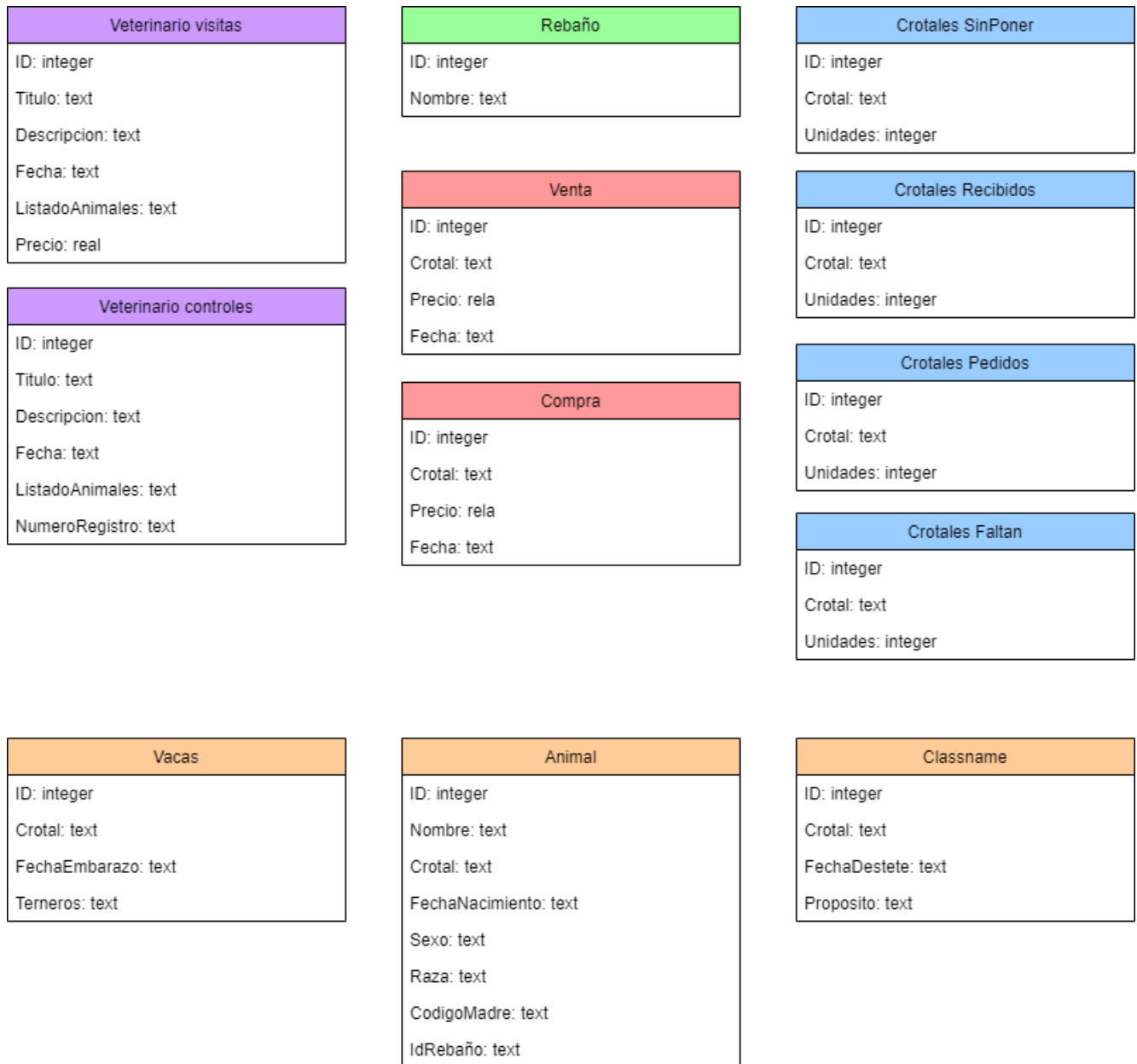
- x. Diagrama común PagerAdapter.

MainActivity

MainActivity
FeedReaderDbHelper_VacApp dbHelper_vacApp BD_Animales bdAnimales BD_Animales_Terneros bdAnimalesTerneros BD_Animales_Vacas bdAnimalesVacas BD_Crotales_Faltan bdCrotalesFaltan BD_Crotales_Pedidos bdCrotalesPedidos BD_Crotales_Recibidos bdCrotalesRecibidos BD_Crotales_SinPoner bdCrotalesSinPoner BD_Rebaños bdRebaños BD_Veterinario_Controles bdVeterinarioControles BD_Veterinario_Visitas bdVeterinarioVisitas BD_Compraventa_Compras bdCompraventaComp BD_Compraventa_Ventas bdCompraventaVentas DrawerLayout drawerLayout ActionBarDrawerToggle toggle NavigationView navigationView ViewPager viewPager TabLayout tabLayout PagerAdapter_Animales adapter PagerAdapter_Crotales adapter_crotales PagerAdapter_Rebano adapter_rebano PagerAdapter_Veterinario adapter_veterinario PagerAdapter_Datos adapter_datos PagerAdapter_Compras adapter_compras PagerAdapter_Ventas adapter_ventas TabItem itemAnimales, itemVacas, itemTerneros, itemToros TabItem itemFaltan, itemPedidos, itemRecibidos, itemSinPoner TabItem itemRebano TabItem itemControles, itemVisitas TabItem itemExportar, itemImportar, itemDestete, itemDinero TabItem itemCompras TabItem itemVentas, itemPlanificadas, itemRealizadas
onCreate(Bundle savedInstanceState) iniciar() insertarDatosPrueba() iniciarAnimales() iniciarCompras() iniciarVentas() iniciarCrotales() iniciarVeterinario() iniciarDatos() iniciarRebaño() onNavigationItemSelected(Menuitem item)

xi. Diagrama MainActivity.

4.2.2. Diagrama BD

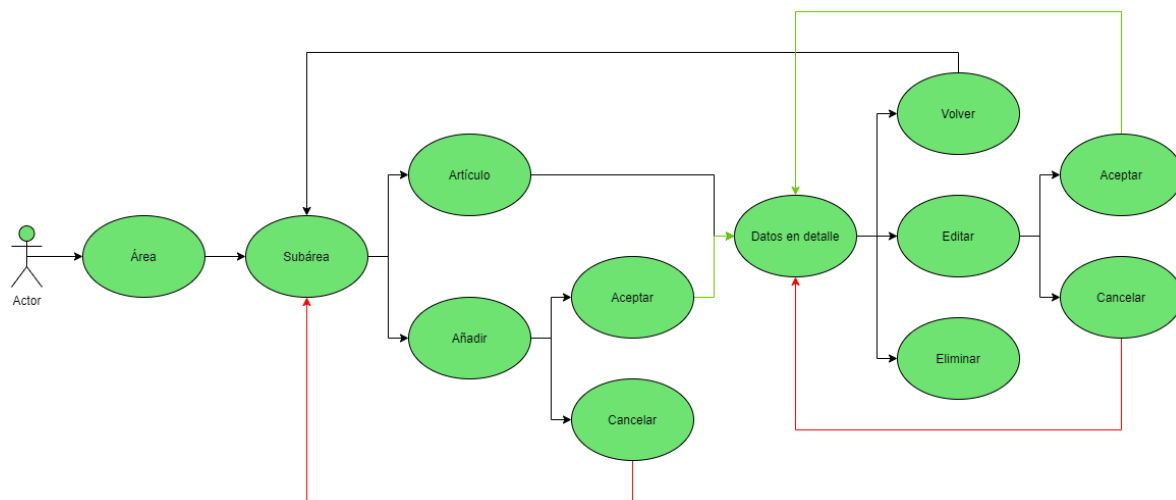


xii. Diagrama BD.

4.3. Interacción con el usuario

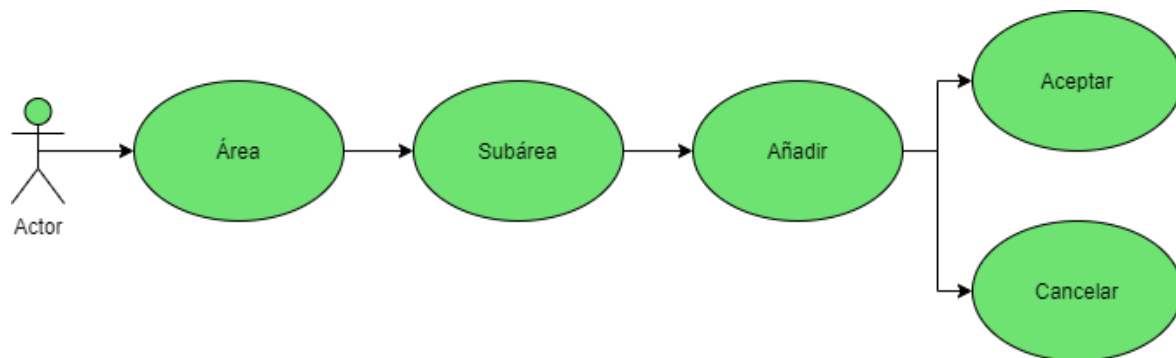
4.3.1 Casos de uso

Estructura del diagrama general de los casos de uso de las diferentes áreas y subáreas.



xiii. Casos de uso. General.

Añadir datos



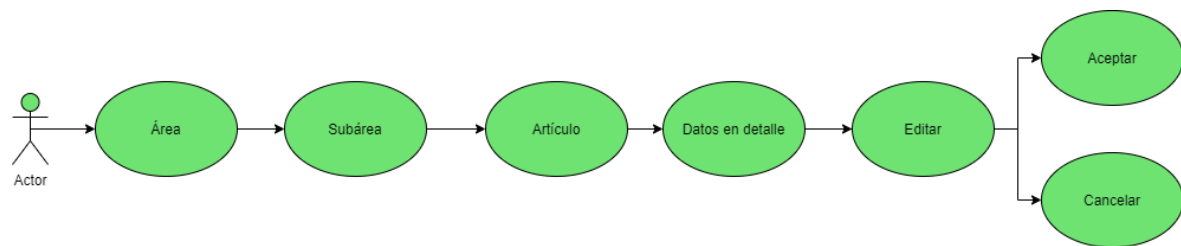
xiv. Casos de uso. Añadir.

ESCENARIO "Inserción de un nuevo dato"	
Numeración	1
Precondiciones	El usuario debe situarse en el subárea de la aplicación dónde desea editar los datos.
Postcondiciones	
Quién lo comienza	Usuario.
Quién lo finaliza	Usuario.
Excepciones	Datos introducidos de manera incorrecta o campos obligatorios vacíos.
Descripción	El usuario trata de introducir un nuevo dato en la aplicación clicando en el botón añadir de la subárea. Se muestra una nueva ventana donde se introducen los datos. El usuario clicará

	<p>en el botón aceptar tras introducir los datos para guardarlos en la aplicación, o cancelar para desecharlos.</p> <p>Si se guardan los datos, se mostrará la ventana con los datos introducidos, pero sin ser editables.</p>
--	--

i. Casos de uso. Añadir.

Editar datos

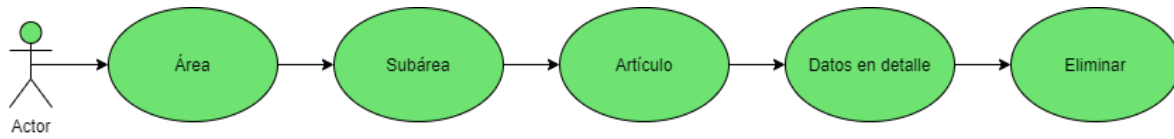


xv. Casos de uso. Editar.

ESCENARIO “Edición de datos”	
Numeración	2
Precondiciones	El usuario debe situarse en la subárea de la aplicación dónde desea editar los datos.
Postcondiciones	
Quién lo comienza	Usuario.
Quién lo finaliza	Usuario.
Excepciones	Datos introducidos de manera incorrecta o campos obligatorios vacíos.
Descripción	El usuario trata de editar los datos clicando en el artículo que muestra un resumen de los datos. Se muestra una nueva ventana con los datos no editables. El usuario debe clicar en el botón de editar. Tras cambiar los datos clicará en aceptar para actualizarlos o cancelar para dejarlos como estaban anteriormente.

ii. Casos de uso. Editar.

Eliminar datos



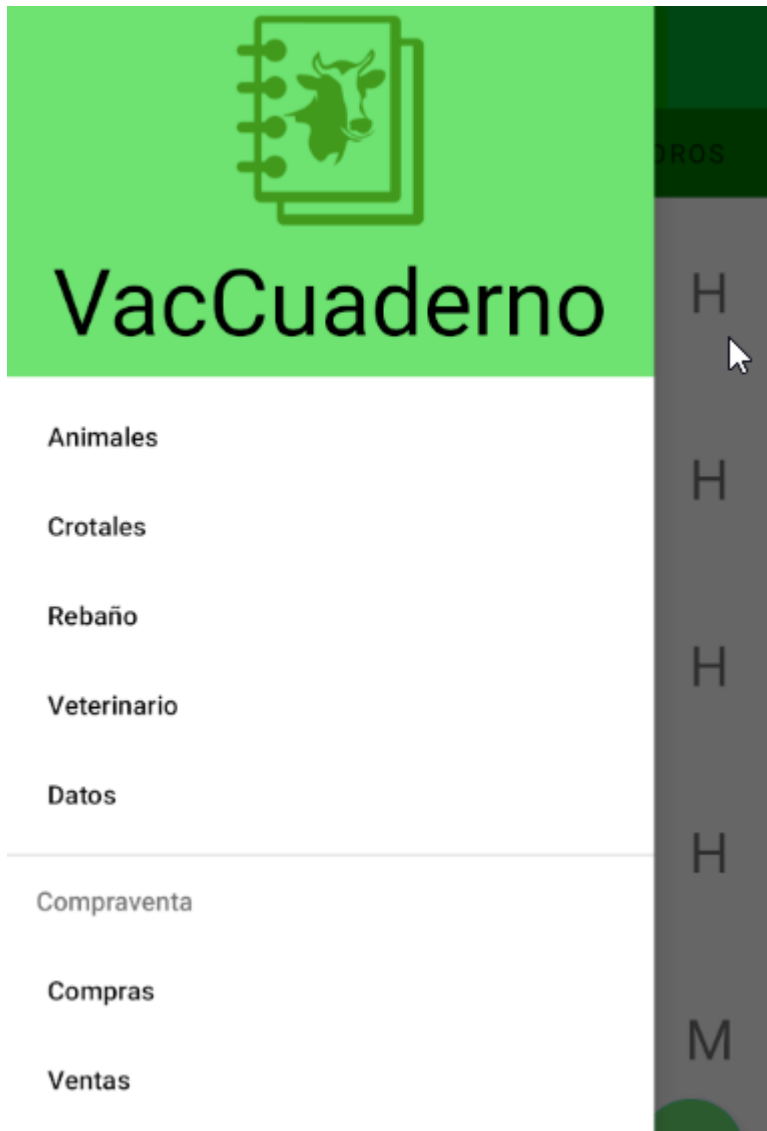
xvi. Casos de uso. Eliminar.

ESCENARIO “Eliminación de datos”	
Numeración	3
Precondiciones	El usuario debe situarse en la subárea de la aplicación dónde desea editar los datos.
Postcondiciones	
Quién lo comienza	Usuario.
Quién lo finaliza	Usuario.
Excepciones	
Descripción	El usuario clicará en el artículo que muestra un resumen de los datos que desea eliminar. Se muestra una vista en detalle de los datos no editables. El usuario debe clicar en el botón eliminar para eliminar los datos mostrados.

iii. Casos de uso. Eliminar.

4.3.2. Diseño de la interfaz.

La interfaz de la aplicación está formada por un menú lateral con las áreas generales de la aplicación.



xvii. Menú lateral con las áreas generales de la aplicación.

Áreas

Área de animales

El área de animales está dividido en las pestañas de Animales, Vacas, Terneros y Toros. En cada pestaña se muestra un listado de los animales con los datos del rebaño, el nombre, el crotal y el sexo.

VacCuaderno			
ANIMALES	VACAS	TERNEROS	TOROS
1	Nombre1		H
	ES 1234 1234 1231		
2	Nombre2		H
	ES 1234 1234 1232		
3	Nombre3		H
	ES 1234 1234 1233		

xviii. Pestaña Animales

Al hacer clic en cada animal se mostrarán sus detalles.

Nombre1	
ES 1234 1234 1231	
Nacimiento	2010-04-11
Sexo	H
Raza	Raza1
Cod Madre	ES 4321 4321 4321
Rebaño	1

xix. Detalle interfaz animales.

En el caso de los terneros se mostrarán también la fecha de destete y el propósito.

Nombre4
ES 1234 1234 1234

Nacimiento	2020-04-14
Sexo	H
Raza	Raza1
Cod Madre	ES 1234 1234 1231
Rebaño	4
Destete	2020-08-14
Proposito	Proposito4

i. Detalle interfaz terneros.

Y en el caso de las vacas se mostrará la fecha del último embarazo y sus terneros.

Nombre3
ES 1234 1234 1233

Nacimiento	2010-04-13
------------	------------

Sexo	H
------	---

Raza	Raza3
------	-------

Cod Madre	
-----------	--

Rebaño	3
--------	---

Embarazo	2019-07-05
----------	------------

Terneros

ii. Detalle interfaz vacas.

Área crotales

El área de crotales está dividido en las pestañas Faltan, Pedidos, Recibidos y Sin poner.

En cada pestaña se muestra un listado con los crotales que pertenecen a cada subárea, mostrando el número de crotal y las unidades.

FALTAN	PEDIDOS	RECIBIDOS	SIN PONER
ES 1234 1234 1231			1
ES 1234 1234 1232			2

iii. Pestañas crotales.

Al hacer clic en uno de los artículos se muestran sus detalles.

Crotal	ES 1234 1234 1231
Unidades	1

iv. Detalle interfaz crotal.

Área rebaños

El área de rebaños está formado por una única pestaña que muestra los diferentes rebaños, mostrando un nombre y un ID.

REBAÑO	
Campo	2
Granja	1

v. Pestaña rebaños.

Al hacer clic en uno de los artículos se muestran sus detalles.

ID	1
Nombre	Campo

vi. Detalle interfaz rebaño.

Área veterinaria

El área de veterinario está formada por las pestañas Controles y Visitas.

En la pestaña de controles se muestra un listado de los controles con un título, la fecha en la que se realizó y un número identificativo.

CONTROLES	VISITAS
Control2	1A78 1B78
2019-04-15	
Control1	1A34 1B34
2020-04-15	

vii. Pestaña controles veterinarios.

Al hacer clic en uno de los artículos se muestran sus detalles.

Control2	
Descripción	
Fecha	2019-04-15
Animales	ES 1234 1234 1238

- viii. Detalle interfaz detalles controles veterinarios

En la pestaña de visitas se muestra un listado de las visitas con un título, la fecha en la que se realizó y el precio.

CONTROLES	VISITAS
Visita1	
2020-04-15	120.0
Visita2	
2020-05-05	50.0

ix. Pestaña visitas veterinarias

Al hacer clic en uno de los artículos se muestran sus detalles.

Visita2
Descripción
Fecha 2020-05-05
Animales ES 1234 1234 1238

x. Detalle interfaz detalles visitas veterinarias.

Área compraventa

El área de compraventa se divide en las áreas de compras y ventas. El área de compras solo contiene la pestaña Compras y el área de ventas se divide en Ventas, Planificadas y Realizadas. Ambas muestran un listado con los crotales de los animales, la fecha y el precio.

VENTAS	PLANIFICADAS	REALIZADAS
ES 1234 1234		1200.0
2018-04-11		
ES 1234 1234		1500.0
2018-04-13		

xi. Pestaña ventas.

Al hacer clic en uno de los artículos se muestran sus detalles, común en ventas y compras.

ES 1234 1234 1231
Precio 1200.0
Fecha 2018-04-11

xii. Detalle interfaz detalles compraventa.

Funcionalidades botones



Añadir



Añade nuevos datos dependiendo desde la subárea desde dónde se clicca. Tras clicarlo muestra una nueva vista de los detalles de los datos a introducir.

Volver



Aparece desde la vista de detalles de los datos de las diferentes subáreas, y retrocede la aplicación a la anterior vista, que es el listado de los datos de la subárea.

Editar



Aparece en la vista de detalles y sirve para editar los datos de la vista.

Eliminar



Aparece en la vista de detalles y sirve para eliminar los datos de la vista.

Aceptar



Aparece cuando se están añadiendo o editando datos en las vistas de detalles. En la vista de edición guarda los datos modificados y en la de añadir guarda los nuevos datos.

Cancelar



Aparece cuando se están añadiendo o editando datos en las vistas de detalles. En la vista de edición cancela la edición de los datos y vuelve al estado antes de comenzar la edición, y en la de añadir cancela la inserción de los nuevos datos.

4.4 Código fuente.

Datos genéricos

El `GridAdapter_Compraventa` utiliza datos genéricos para funcionar con la clase `Compra y Venta`.

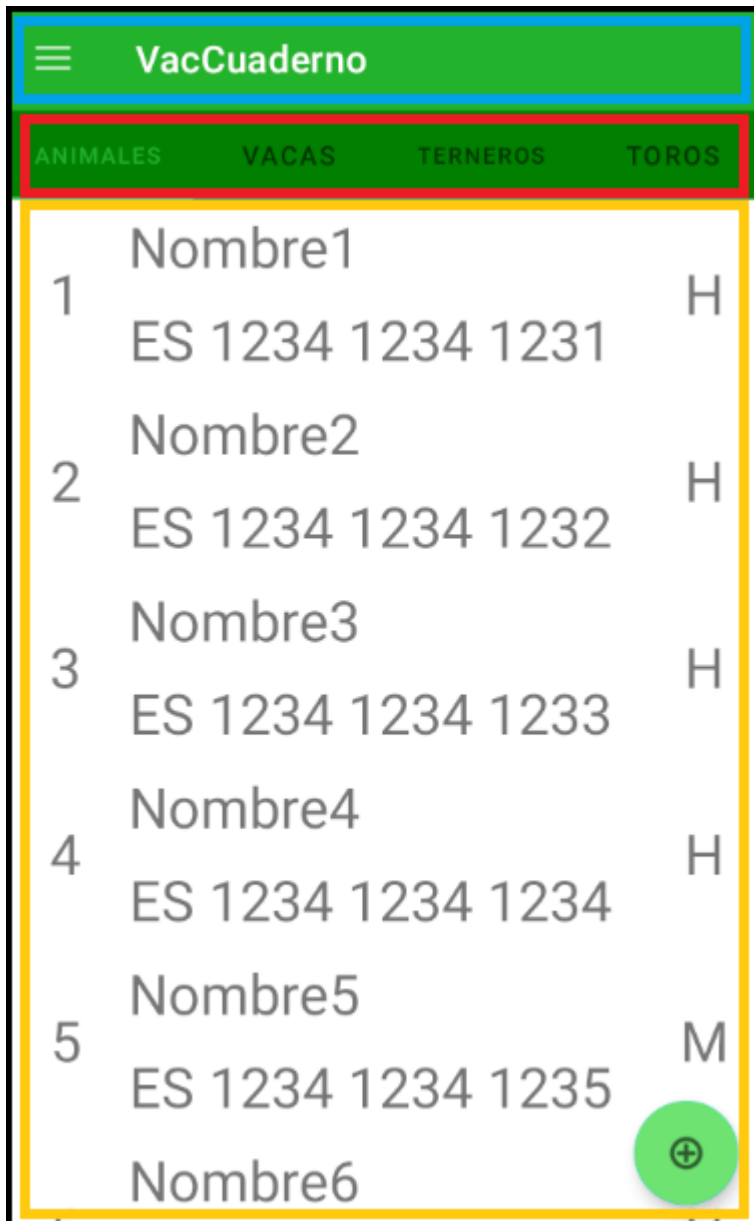
```
public<C extends CompraVenta>GridAdapter_Compraventa(Contextcont, ArrayList<C>
al){
    this.context= cont;
    this.arrayList= al;
}
```

Estructura de la aplicación

La aplicación está formada por un menú lateral formado por un `nav_header.xml`, que contiene los datos de la aplicación, y varios `ítems`.



Cada uno de los ítems lleva a un área de la aplicación. Estas áreas están formadas por un archivo `content_main_nombreArea.xml`. Este archivo contiene una `ToolBar`, un `TabLayout` y un `fragment_subarea.xml`.



El fragment contiene un GridView que se inicia con un GridAdpater_Subarea.java, que añade item_objeto con los datos obtenidos de la base de datos.

Al seleccionar en cada item se inicia una Activity_Subarea que muestra los detalles de los datos.

El detalle contiene uno o varios **layouts** con los datos de la BD y varios **botones flotantes**.

Cada botón tiene un listener que realizan las diferentes acciones.

Según si se están editando o introduciendo datos se pueden editar los datos y aparecen diferentes botones.

Nombre6	
ES 1234 1234 1236	
Nacimiento	2020-04-16
Sexo	H
Raza	Raza3
Cod Madre	ES 1234 1234 1233
Rebaño	6
Destete 2020-08-16	
Proposito	Propisto6
	

5. Fase de pruebas

5.3 Pruebas realizadas

	Descripción	Parámetros	Resultado	Solución
01	Correcta definición de las vistas de la estructura general de la aplicación		CORRECTO	
02	Mostrar las vistas correctas según la pestaña	Pestaña seleccionada	ERROR	Corregir el “grid” y su adaptador
03	Mostrar las vistas de detalle correcta según la pestaña	Pestaña desde la que se accede a los detalles	ERROR	Corregir clase pasada al crear la actividad
04	Obtener e insertar datos en las vistas	Datos introducidos o a mostra	ERROR	Iniciar los objetos que referencian a la vista antes de tratar de mostrar los datos
05	Añadir datos	Datos a introducir	ERROR	Añadir un constructor vacío con datos de prueba y si es una fecha comprobar si está vacía antes de tratar de convertirla
06	Editar datos	Datos cambiados	ERROR	Guardar los datos directamente en el objeto de la actividad para evitar perder el id

07	Eliminar datos		ERROR	Guardar el id autogenerado en el objeto al guardarlo en la BD
		El objeto eliminado es una Vaca o un Ternero desde la vista de Animal	ERROR	Eliminar los datos de vaca o ternero si hay un objeto con el mismo crotal guardado
08	Exportar datos		ERROR	

iv. Pruebas realizadas

6. Conclusiones finales:

6.1. Reflexión de los resultados obtenidos.

6.2. Grado de cumplimiento de los objetivos fijados.

Se han cumplido los objetivos a excepción del área de gestión de los datos y el dar de alta o baja de forma automática a los animales en la compraventa. A pesar de ello y de que la interfaz podría estar más elaborada, la aplicación es muy similar a la planificada inicialmente.

6.3 Propuesta de modificaciones o ampliaciones futuras del sistema

Algunas de las modificaciones a realizar sería la comprobación de los datos antes de guardarlos y la modificación de las interfaces, especialmente al mostrar los listados, para mostrar los datos de manera más clara.

Algunas de las ampliaciones a realizar sería la introducción de imágenes en los animales. También poder cambiar el tipo de animal, por ejemplo, de Ternero a Vaca, sin tener que borrar los datos y crearlos de nuevo.

7. Dificultades y problemas fundamentales encontrados.

- Manejo de la BD SQLite, debido al desconocimiento de este gestor de base de datos.
- Desequilibrio entre el código realizado y el código documentado a lo largo del desarrollo.
- Manejo de las vistas. Encontrando errores al insertar u obtener datos debido a utilizar objetos que referencian a la vista antes de iniciarlos o tratar de insertar u obtener datos nulos.

8. Documentación del sistema desarrollado

8.3 Manual de instalación.

Incluido como Anexo III.

8.4 Manual de uso.

Incluido como Anexo IV.

9. Bibliografía:

9.3 Incluirá toda la documentación consultada: libros, apuntes, páginas webs, foros, etc.

- Apuntes recibidos y actividades realizadas durante el ciclo superior de Desarrollo de Aplicaciones Multiplataforma, especialmente de el módulo de Programación Multimedia y de Dispositivos Multimedia.
- Las siguientes páginas web:
- <https://developer.android.com/>
- <https://es.stackoverflow.com/>

10. Anexos.

Anexo I. Diagrama de Gantt.

Proyecto

Powered by  **monday.com**

[Click here to start your free trial](#)

Proyecto Desarrollo de Aplicaciones Multiplataforma 2020-2021

1º Fase

Nombre	Subelementos	Estado	Fecha programada	Fecha realización	Esfuerzo planificado	Esfuerzo utilizado	Déficit de esfuerzo
Planificación de los objetivos		Listo	2021-03-08	2021-03-08	2,5		2,5
Planificación de la estructura de la aplicación		Listo	2021-03-10	2021-03-10	5		5
Realización de los principales casos de uso		Listo	2021-03-12	2021-03-12	5		5
Planificación de los medios hardware y software		Listo	2021-03-14	2021-03-08	2,5		2,5
Realización de los diagramas generales de la BD		Listo	2021-03-16	2021-03-21	5		5
Realización del primer prototipo		Listo	2021-04-18	2021-03-21	10		10
Documentación		Listo	2021-03-20	2021-03-21	5		5

Desde el 2021-03-08 hasta el 2021-04-18	Desde el 2021-03-08 hasta el 2021-03-21	35	0	35
---	---	----	---	----

2º Fase

Nombre	Subelementos	Estado	Fecha programada	Fecha realización	Esfuerzo planificado	Esfuerzo utilizado	Déficit de esfuerzo
Realización del diagrama UML		Listo	2021-03-23	2021-03-21	5	5	0
Realización de los diagramas específicos de la BD		Listo	2021-03-25	2021-03-26	5	3	2
Diseño de la base de datos		Listo	2021-03-27	2021-04-03	5	3	2
Realización de pruebas en la BD		Listo	2021-03-29	2021-04-03	5	3	2
Primera versión de la aplicación		Listo	2021-04-03	2021-04-03	25	18	7
Subitems	Name	Estado	Fecha programada	Fecha realización	Esfuerzo planificado	Esfuerzo utilizado	Déficit de esfuerzo
	Diseño general de la interfaz de la aplicación móvil	Listo	2021-03-30	2021-04-03	20	15	5
	Realización de pruebas	Listo	2021-04-02	2021-04-03	5	3	2
Documentación		Listo	2021-04-04	2021-04-04	5	3	2
			Desde el 2021-03-23 hasta el 2021-04-04	2021-03-22	50	35	15

3º Fase

Nombre	Subelementos	Estado	Fecha programada	Fecha realización	Esfuerzo planificado	Esfuerzo utilizado	Déficit de esfuerzo
Segunda versión de la aplicación		Listo	2021-04-13		20	17	3
Subitems	Name	Estado	Fecha programada	Fecha realización	Esfuerzo planificado	Esfuerzo utilizado	Déficit de esfuerzo
	Realización de la clase animal en java y de sus subclases vaca y ternero	Listo	2021-04-06	2021-04-14	5	2	3
	Diseño de la interfaz del área de animales	Listo	2021-04-08	2021-04-17	10	12	-2
	Crear las clases Rebaño, Veterinario y sus subclases Controles y Visitas, Compraventa y sus subclases Compra y Venta, y Crotales	Listo	2021-04-12	2021-04-16	5	3	2
Realización de pruebas y corrección de errores		Listo	2021-04-14	2021-04-18	5	3	2
Documentación		Listo	2021-04-18	2021-04-18	5	3	2
			Desde el 2021-04-13 hasta el 2021-04-18		30	23	7

4º Fase

Nombre	Subelementos	Estado	Fecha programada	Fecha realización	Esfuerzo planificado	Esfuerzo utilizado	Déficit de esfuerzo
Tercera versión de la		Listo	2021-04-27		30	22	8

aplicación

Subitems	Name	Estado	Fecha programada	Fecha realización	Esfuerzo planificado	Esfuerzo utilizado	Déficit de esfuerzo
	Crear las interfaces de las áreas Rebaño, Veterinario, Compraventa y Crotales	Listo	2021-04-20		10	10	0
	Crear las clases Exportar y Estadísticas	Listo	2021-04-24		8,5	9	-0,5
	Crear las clase Importar	No realizado	2021-04-24		4	0	4
	Crear las interfaces de las áreas Exportar, Importar y Estadísticas	Listo	2021-04-26		7,5	3	4,5
	Realización de pruebas y corrección de errores	Listo	2021-04-30		5	5	0
Documentación		Listo	2021-05-02		5	5	0
			Desde el 2021-04-27 hasta el 2021-05-02		40	32	8

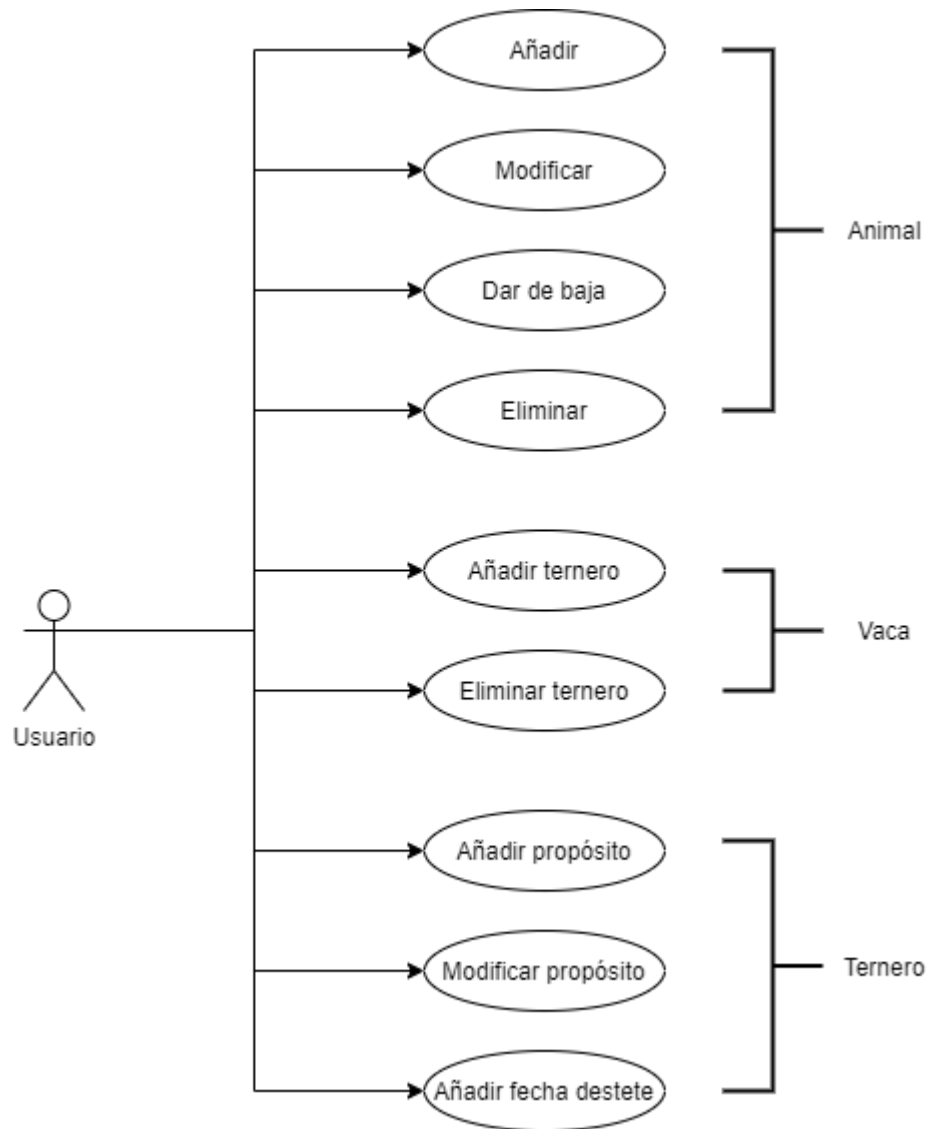
5º Fase

Nombre	Subelementos	Estado	Fecha programada	Fecha realización	Esfuerzo planificado	Esfuerzo utilizado	Déficit de esfuerzo
Última versión de la aplicación		Listo	2021-05-07		10		10
Subitems	Name	Estado	Fecha programada	Fecha realización	Esfuerzo planificado	Esfuerzo utilizado	Déficit de esfuerzo
	Retoques finales de las	Listo	2021-05-04		5	5	0

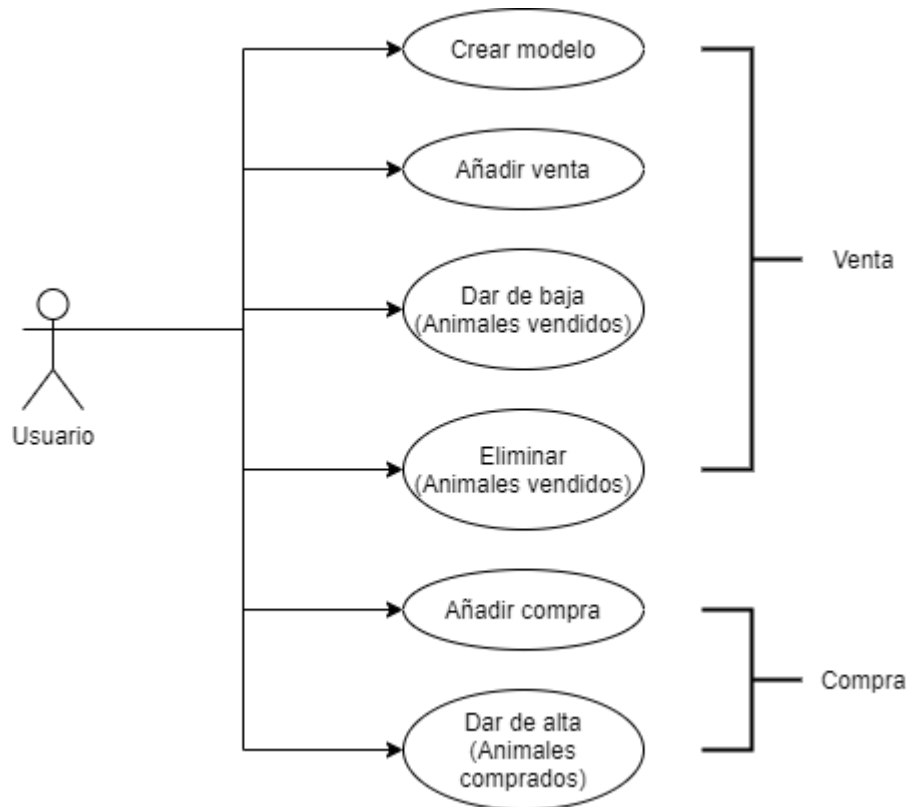
	interfaces					
	Gestión de las ayudas	Listo	2021-05-06	5	0	5
Análisis de los objetivos		Listo	2021-05-08	5	1	4
Realización de pruebas en cada una de las áreas comprobando su integración con el resto de las áreas y corrección de errores		Listo	2021-05-10	10	5	5
Depuración del código		Listo	2021-05-14	5	10	-5
Elaboración del manual		Listo	2021-05-16	5	3	2
Documentación		Listo	2021-05-17	5	10	-5
			Desde el 2021-05-07 hasta el 2021-05-17	50	34	40

Anexo II. Casos de uso

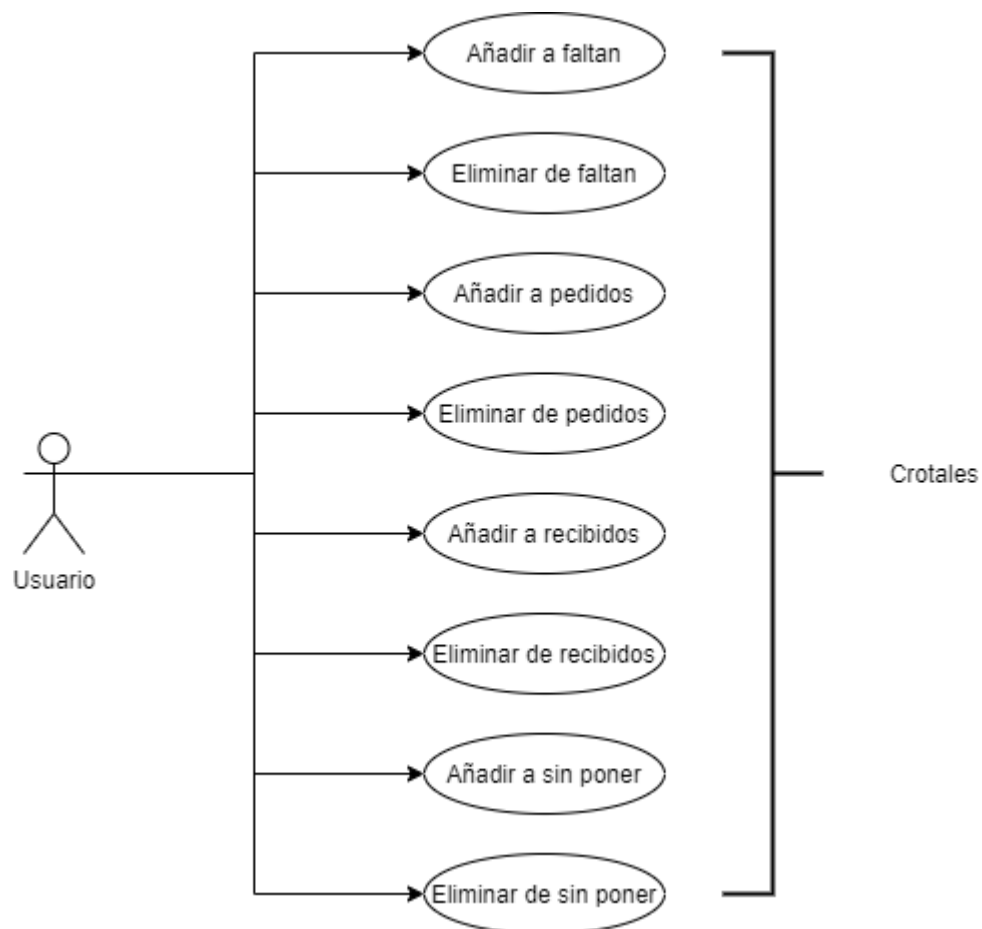
Animales



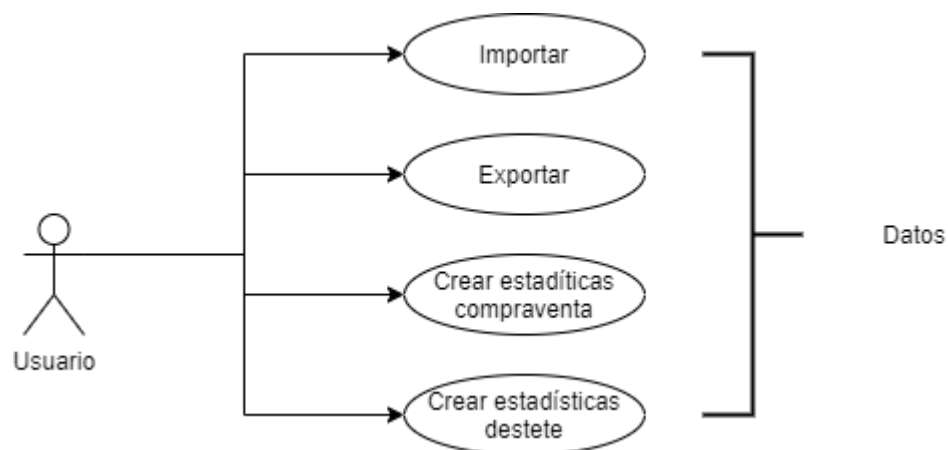
Compraventa



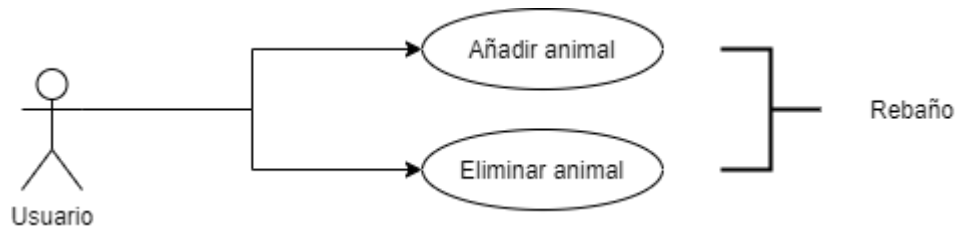
Crotales



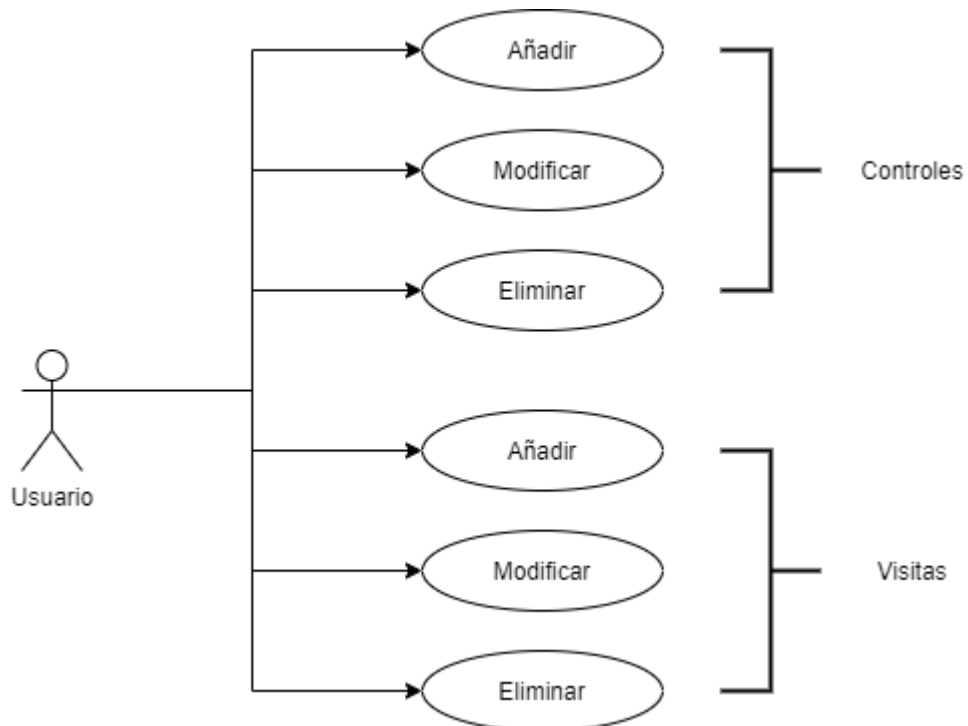
Datos



Rebaño

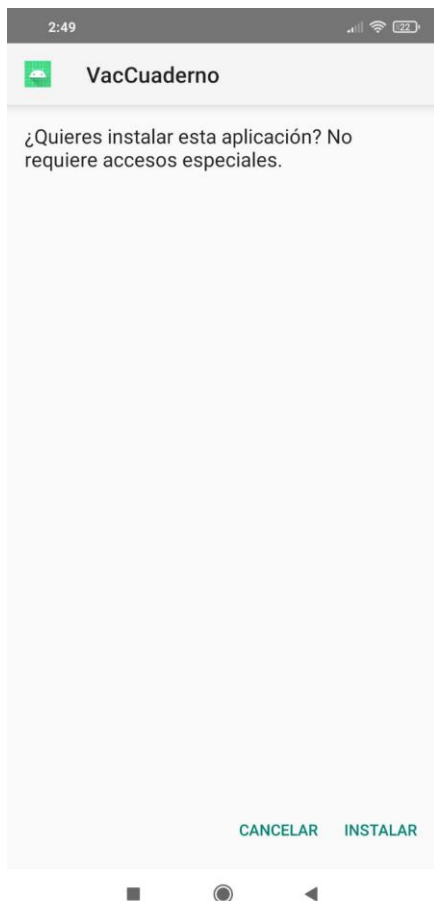


Veterinario



Anexo III. Manual de instalación

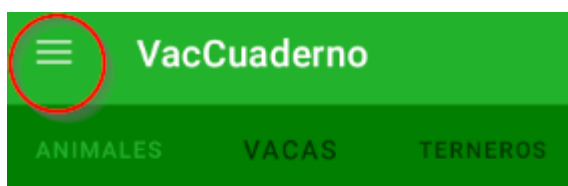
Tras descargar el apk y aceptar los permisos del dispositivo para ejecutar aplicaciones externas, clicar en el botón instalar. Tras ello se iniciará la aplicación en el área de animales, donde se podrá comenzar a insertar nuevos datos.



Anexo IV. Manual de uso

¿Cómo ver los datos guardados?

Para ver los datos guardados debe dirigirse al área que quiere consultar desde el menú desplegable.



- 1 Nombre1
ES 1234 1234 123
- 2 Nombre2

Seleccionar el área.



Animales

Crotales

Rebaño

Veterinario

Y seleccionar la subárea en las pestañas superiores.



4	Nombre4	H
	ES 1234 1234 1234	
5	Nombre5	M
	ES 1234 1234 1235	

Para ver los datos en mayor detalle debe seleccionar en el artículo que desea ver.

ANIMALES	VACAS	TERNEROS	TOROS
4	Nombre4		H
	ES 1234 1234 1234		
5	Nombre5		M
	ES 1234 1234 1235		
6	Nombre6		H
	ES 1234 1234 1236		

Y podrá ver los datos en detalle.

Nombre5	
ES 1234 1234 1235	
Nacimiento	2020-04-15
Sexo	M
Raza	Raza2
Cod Madre	ES 1234 1234 1232
Rebaño	5
Destete	2020-08-15
Proposito	Propisto5

¿Cómo añadir datos?

Tras acceder a la pestaña de la subárea correspondiente como se indica anteriormente, debe clicar en el botón añadir.



4 Nombre4 H
ES 1234 1234 1234

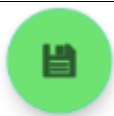
5 Nombre5 M
ES 1234 1234 1235

6 Nombre6 H
ES 1234 1234 1236

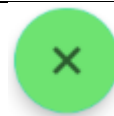


Tras darle al botón se inicia una vista de detalles. Tras introducir los datos debe darle a guardar, o a cancelar si no desea guardarlos.

Aceptar



Cancelar



¿Cómo editar los datos?

Tras seleccionar el artículo que se desea editar se selecciona el botón editar en la vista de detalles.



Tras cambiar los datos se selecciona aceptar para guardar los cambios o cancelar para desecharlos.

¿Cómo eliminar los datos?

Tras seleccionar el artículo que se desea eliminar se selecciona el botón eliminar en la vista de detalles.

