



UNIVERSIDAD TECNOLÓGICA DE LA SELVA

REPORTE DE ESTADÍA TITULADO “APLICACIÓN MÓVIL PARA...”, REALIZADO
EN LA CIUDAD DE TUXTLA GUTIÉRREZ, CHIAPAS,

QUE PRESENTA EL

C. XXX XXX XXX XXX.

COMO REQUISITO PARA OBTENER EL TÍTULO DE:

**TÉCNICO SUPERIOR UNIVERSITARIO EN TECNOLOGÍAS DE LA
INFORMACIÓN Y COMUNICACIÓN ÁREA DE SISTEMAS INFORMÁTICOS.**

ASESOR EMPRESARIAL
ING. XXX XXX XXX

ASESOR ACADÉMICO
ING. HÉCTOR HUGO HERRERA ESCOBAR

XXX, CHIAPAS; AGOSTO DE 20XX

ÍNDICE

I.	INTRODUCCIÓN	1
I.1	Conceptualización de la estadía.....	1
I.2	Contextualización	3
I.2.1	Localización geográfica.	3
I.2.2	Giro y tamaño de la empresa.	5
I.2.4	Objetivos de la empresa.....	6
II.	DESARROLLO	7
II.1	Marco referencial	7
II.1.1	Microeconomía de la empresa.....	7
II.1.2	Área motivo de estudio	7
II.1.3	Desarrollo del Objetivo	8
II.2	Estado del Arte	9
II.2.1	Marco histórico	9
II.2.2	Principios de usabilidad.....	10
II.2.3	Patrón a utilizar.	12
II.2.4	Software de desarrollo.....	15
II.3	Materiales y métodos	21
II.3.1	Metodología utilizada.	23
II.4.	Procesamiento de resultados, análisis y discusión.	27
Caso de uso.....		27
Descripción de catalogo.....		28
Modelo relacional.		29
Diccionario de datos.....		30

Script de la base de datos	40
Wireframes de la aplicación.....	49

ÍNDICE DE ILUSTRACIONES

Ilustración 1 Tuxtla Gutiérrez	3
Ilustración 2 Localización de Sistelligent	4
Ilustración 3: Sistelligent Soluciones en Informática S.A. de C.V.	4
Ilustración 4: Modelo, Vista Controlador	12
Ilustración 5: SQLite	15
Ilustración 6: Android Studio	17
Ilustración 7: MySQL	19
Ilustración 8: Modelo en cascada	23
Ilustración 9: Caso de uso	27
Ilustración 10: Modelo relacional	29
Ilustración 11: Pantalla login	49
Ilustración 12: Pantalla principal	49
Ilustración 13: Pantalla agregar usuario	50

ÍNDICE DE TABLAS

Tabla 1.Hardware utilizado	21
Tabla 2 Software utilizado.....	22
Tabla 3: Descripción de catalogo	28
Tabla 4 ccliente.....	30
Tabla 5 cganado	32
Tabla 6 cnfamilia	33
Tabla 7 ctarea	34
Tabla 8 cusuario	35
Tabla 9 cvacunas	35
Tabla 10 halimentacion	36
Tabla 11 hgestacion	37
Tabla 12 htarea.....	38
Tabla 13 hubicacion	39
Tabla 14 hvacuna	39

I. INTRODUCCIÓN

I.1 Conceptualización de la estadía.

La estadía consiste en el período durante el cual el estudiante aplica sus conocimientos en el desarrollo de un proyecto, acorde a las necesidades de una empresa, como parte de su formación profesional y de acuerdo a los planes de estudio vigentes que corresponda. Cabe mencionar que la Estadía es requisito indispensable para la titulación de los universitarios; es de tiempo completo, es decir dedicados únicamente a la realización del proyecto de estadía, desde las instalaciones de las diferentes empresas.

Los alumnos durante su estadía deben realizar un proyecto que satisfaga una necesidad o solucione un problema de la empresa bajo la orientación y supervisión de un profesor asesor y un asesor empresarial.

Beneficios para el alumno:

- ✓ Consolidar y fortalecer la formación profesional de los alumnos a través de una estadía profesional dentro del sector productivo, destacando el dominio y aplicación práctica de los conocimientos que adquirieron en su plan de estudios.
- ✓ El alumno aprenderá a distinguir sus fortalezas y debilidades al poner en práctica sus conocimientos.
- ✓ Reflexionar sobre su propia responsabilidad.
- ✓ Mejorará la capacidad para establecer relaciones interpersonales.
- ✓ Facilitará su incorporación al trabajo a través de su buen desempeño.

Beneficios para la empresa

- ✓ Facilita la solución de problemas a través del apoyo de los alumnos.
- ✓ No existe compromiso laboral durante el período de Estadía Profesional.
- ✓ Cuenta con profesionales que contribuyan a la solución de problemas en diversas áreas.
- ✓ Se obtienen recursos humanos calificados con un cúmulo de conocimientos aplicables en la empresa, con una actitud positiva y de innovación.

I.2 Contextualización

I.2.1 Localización geográfica.

I.2.1.1 Nivel estatal.

Tuxtla Gutiérrez es una ciudad y municipio mexicano, capital y núcleo urbano más grande del estado de Chiapas. Es sede de los poderes públicos del estado de Chiapas y centro de la Zona Metropolitana de Tuxtla que ha sido definida como la integración de los municipios de Chiapa de Corzo, Berriozábal, San Fernando, Suchiapa, Ocozocoautla de Espinosa y Osumacinta.

Este municipio tiene una extensión territorial de 412.40 km². Limita al norte con San Fernando y Osumacinta, al este con Chiapa de Corzo, al sur con Suchiapa y al oeste con Ocozocoautla y Berriozábal.

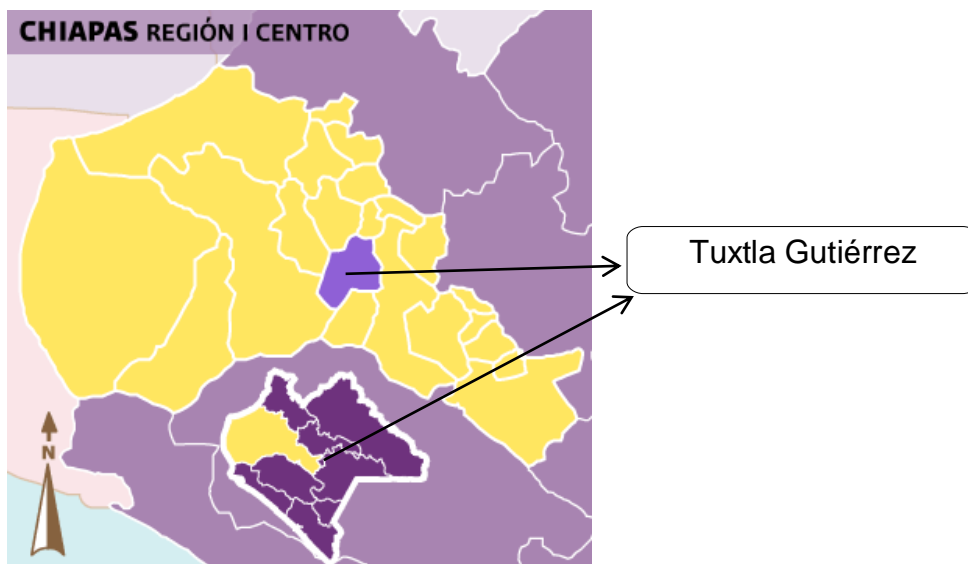


Ilustración 1 Tuxtla Gutiérrez

I.2.2 Giro y tamaño de la empresa.

Sistelligent Soluciones en Informática S.A. de C.V. es una empresa mexicana en el ramo de los servicios tecnológicos para el sector privado y público. Actualmente se ha centrado en el desarrollo de software.

Sistelligent proporciona aplicaciones (móvil, de escritorio y pagina web) a medida de las necesidades del cliente.

SiVent es un software de punto de venta desarrollado por Sistelligent. El software cuenta con los módulos de Ventas, Compras, Clientes, Cuentas por Cobrar, Proveedores, Cuentas por Pagar, que es lo básico con lo debe contar un punto de venta a un precio altamente competitivo. Para el cual se proporcionan cursos a nivel estado

Adicional mente cuenta con venta de equipos de cómputo, impresoras y accesorios además de instalación de sistema de vigilancia.

I.2.3 Área de influencia

Sistelligent Soluciones en Informática S.A. de C.V. es una de las pocas empresas de desarrollo de software en la ciudad de Tuxtla Gutiérrez y en el estado de Chiapas por ello muchas personas necesitan suplir sus necesidades con algún tipo de tecnología relacionada al área informática recurran a esta empresa.

El área de influencia de la empresa es a nivel estado ya que esta empresa desarrollo un punto de venta "SiVent" el cual es vendido a nivel estado así como brindar cursos para su utilización en ciudades como Comitán de Domínguez y Tapachula. También se realizan venta de equipos de computa al menudeo y mayoreo.

I.2.4 Objetivos de la empresa.

Sistelligent Soluciones en Informática S.A. de C.V. tiene como propósito ofrecer las soluciones tecnológicas adecuadas a cada cliente mediante la correcta planeación, administración e implementación de sus sistemas de información. Promoviendo soluciones informáticas con metodologías adecuadas a la organización mediante el uso correcto de la tecnología, mejorando la productividad, control, eficiencia y análisis de resultados; mejorando a su vez la interacción y servicio al cliente.

Misión

Integrar la tecnología en las organizaciones para mejorar el control, productividad, análisis y servicio al cliente mediante el uso de equipos y sistemas informáticos de acuerdo a sus necesidades fomentando la productividad, crecimiento y organización de nuestros clientes.

II. DESARROLLO

II.1 Marco referencial

II.1.1 Microeconomía de la empresa

Sistelligent Soluciones en Informática S.A. de C.V. es una empresa dedicada a proponer soluciones y/o resolver problemas de forma tecnológica utilizando las TIC's, por lo cual apoya a pequeñas y medianas empresas implementando soluciones en ellas.

Es una de las pocas empresas dedicadas al desarrollo de software que se encuentran en el estado de Chiapas por ello tiene un amplio campo de influencia, por el cual se da cursos para el uso del Sistema de punto de venta SiVent en varias ciudades del estado.

II.1.2 Área motivo de estudio

En Sistelligent el área de desarrollo de aplicaciones es el encargado de resolver los problemas que son propuestos ante la empresa de una manera tecnológica utilizando aplicaciones que pueden ser páginas web, aplicación móvil o bien de escritorio dependiendo las necesidades y características especificadas por el cliente. Una de estas propuestas es el desarrollar una aplicación móvil para el control de animales de granja así como: caballos, cerdos, vacas, ovejas entre otros. Actualmente no se cuenta con una herramienta utilizando las TIC's que ayuden a resolver o automatizar procesos, tales como llevar un control de los cambios de pastizales de los animales en caso que lo requieran, llevar el control de las vacunas y cada cuanto se deben aplicar, tareas que se realizan y controlar su gestación.

Las principales mejoras que se proporcionaran serán:

- ✓ Administración de altas, bajas y ventas de los animales.
- ✓ Controlar el cambio de los animales a diferentes pastizales
- ✓ Conocer el periodo de gestación
- ✓ Control de las vacunas que se les aplican
- ✓ Control de las tareas a realizar

II.1.3 Desarrollo del Objetivo

Mejora: El área de desarrollo de aplicaciones de la empresa Sistelligent realizara una mejora para el control de animales de granja, dadas las necesidades y la portabilidad se llegó a la conclusión de realizar el proyecto en un ambiente móvil por lo cual se estará trabajando con el entorno de Android Studio en su versión 2.3.2. Se utilizaran una base de datos local con SQLite para guardar la información, así como una base de datos en MySQL montada en un servidor para que se pueda hacer una sincronización de los datos en el momento en que el usuario así lo vea necesario.

Innovación: El área de desarrollo de aplicaciones desarrollara una aplicación móvil para solucionar la problemática de muchos granjeros o ganaderos que deseen llevar el control de sus animales optando por una solución tecnológica.

Solución a un problema: Desarrollar e implementación de una aplicación móvil para llegar el control y registro de animales de granja tales como: vacas, caballos, cerdos, ovejas entre otros animales.

II.2 Estado del Arte

II.2.1 Marco histórico

Las primeras aplicaciones móviles datan de finales de los 90s, estas eran lo que conocemos como la agenda, arcade games, Los editores de ringtone, etc. cumplían funciones muy elementales y su diseño era bastante simple.

La evolución de las apps se dio rápidamente gracias a las innovaciones en tecnología WAP y la transmisión de data (EDGE) esto vino acompañado de un desarrollo muy fuerte de los celulares.

Para entender el resto de la historia se simplifica todo se irá al presente. Apple lanza el iPhone y junto a él llegan muchas más propuestas de Smartphone, entre ellas Android, la competencia más grande del sistema operativo del iPhone.

Es aquí que empieza el boom de las apps, juegos, noticias, diseño, arte, fotografía, medicina todo en tus manos gracias a la revolución de las aplicaciones móviles. La próxima semana entraré en detalle.

II.2.2 Principios de usabilidad.

Visibilidad del estado del sistema: El sistema debe siempre mantener informado a los usuarios de lo que ocurre, con un correcto feedback en un tiempo razonable.

Correspondencia entre el sistema y el mundo real: El sistema debe hablar el lenguaje de los usuarios con palabras, frases y conceptos familiares. Sigue las convenciones del mundo real. Haz que la información aparezca en forma natural y lógica.

Control y libertad del usuario: Los usuarios frecuentemente eligen opciones por error, por eso indica una salida clara a esas situaciones no deseadas sin necesidad de pasar por extensos diálogos.

Consistencia y estándares: Los usuarios no tienen que adivinar que las diferentes palabras, situaciones o acciones significan lo mismo.

Evitar errores: Un diseño cuidado que previene problemas es mejor que unos buenos mensajes de error.

Reconocimiento: Haz objetos, acciones y opciones visibles. El usuario no tiene que recordar información de una parte a otra. Las instrucciones de uso del sistema deben estar visibles o ser fácilmente recuperables.

Flexible y eficiente: Diseña un sistema que pueda ser utilizado por un rango amplio de usuarios. Brinda instrucciones cuando sean necesarias para nuevos usuarios sin dificultar el camino de usuarios avanzados. Permite a los usuarios avanzados ir directamente al contenido que buscan.

Diseño minimalista: No hay que mostrar información irrelevante. Cada pedazo de información extra compite con la importante y disminuye su relativa visibilidad.

Reconocer, diagnosticar y recuperarse de los errores: Para ayudar a los usuarios, los mensajes de error deben estar escritos en lenguaje sencillo, indicar el problema de forma precisa e indicar una solución.

Ayuda y documentación: Facilita siempre una documentación o ayuda. La información debe ser fácil de encontrar, está dirigida a las tareas de los usuarios, lista los pasos concretos para hacer algo y es breve.

II.2.3 Patrón a utilizar.

MVC es una propuesta de diseño de software utilizada para implementar sistemas donde se requiere el uso de interfaces de usuario. Surge de la necesidad de crear software más robusto con un ciclo de vida más adecuado, donde se potencie la facilidad de mantenimiento, reutilización del código y la separación de conceptos.

Su fundamento es la separación del código en tres capas diferentes, acotadas por su responsabilidad, en lo que se llaman Modelos, Vistas y Controladores, o lo que es lo mismo, Model, Views & Controllers, si lo prefieres en inglés. En este artículo estudiaremos con detalle estos conceptos, así como las ventajas de ponerlos en marcha cuando desarrollamos.

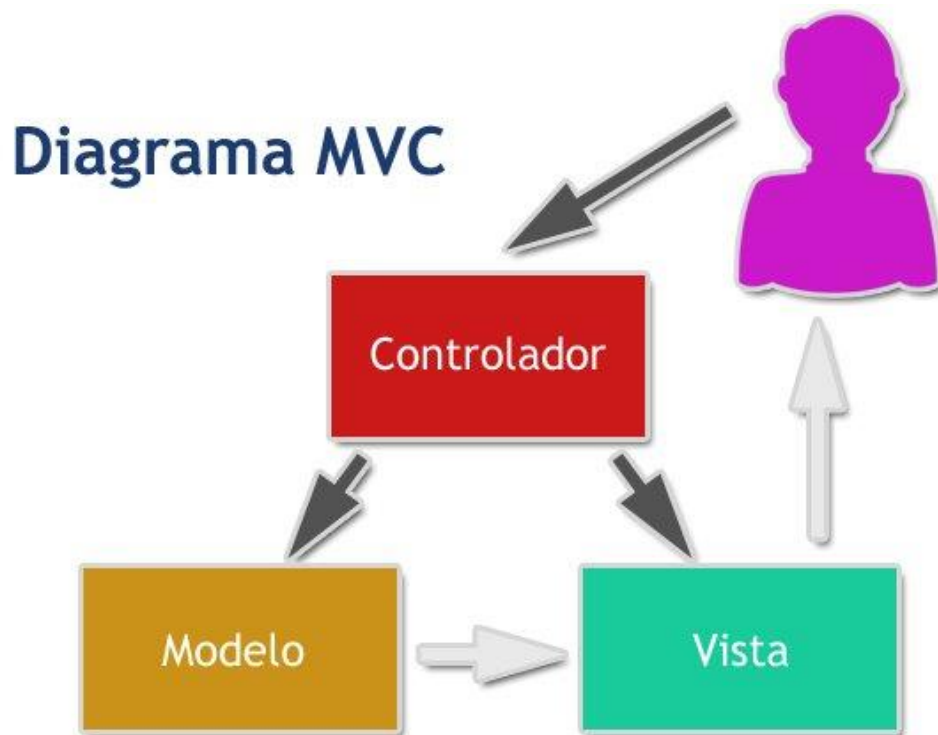


Ilustración 4: Modelo, Vista Controlador

Por qué MVC

La rama de la ingeniería del software se preocupa por crear procesos que aseguren calidad en los programas que se realizan y esa calidad atiende a diversos parámetros que son deseables para todo desarrollo, como la estructuración de los programas o reutilización del código, lo que debe influir positivamente en la facilidad de desarrollo y el mantenimiento.

Los ingenieros del software se dedican a estudiar de qué manera se pueden mejorar los procesos de creación de software y una de las soluciones a las que han llegado es la arquitectura basada en capas que separan el código en función de sus responsabilidades o conceptos. Por tanto, cuando estudiamos MVC lo primero que tenemos que saber es que está ahí para ayudarnos a crear aplicaciones con mayor calidad.

Modelo: Es la capa donde se trabaja con los datos, por tanto contendrá mecanismos para acceder a la información y también para actualizar su estado. Los datos los tendremos habitualmente en una base de datos, por lo que en los modelos tendremos todas las funciones que accederán a las tablas y harán los correspondientes selects, updates, inserts, etc.

No obstante, cabe mencionar que cuando se trabaja con MVC lo habitual también es utilizar otras librerías como PDO o algún ORM como Doctrine, que nos permiten trabajar con abstracción de bases de datos y persistencia en objetos. Por ello, en vez de usar directamente sentencias SQL, que suelen depender del motor de base de datos con el que se esté trabajando, se utiliza un dialecto de acceso a datos basado en clases y objetos.

Vistas: como su nombre nos hace entender, contienen el código de nuestra aplicación que va a producir la visualización de las interfaces de usuario, o sea, el código que nos permitirá renderizar los estados de nuestra aplicación en HTML. En las vistas nada más tenemos los códigos HTML y PHP que nos permite mostrar la salida.

En la vista generalmente trabajamos con los datos, sin embargo, no se realiza un acceso directo a éstos. Las vistas requerirán los datos a los modelos y ellas se generarán la salida, tal como nuestra aplicación requiera.

Controladores: Contiene el código necesario para responder a las acciones que se solicitan en la aplicación, como visualizar un elemento, realizar una compra, una búsqueda de información, etc.

En realidad es una capa que sirve de enlace entre las vistas y los modelos, respondiendo a los mecanismos que puedan requerirse para implementar las necesidades de nuestra aplicación. Sin embargo, su responsabilidad no es manipular directamente datos, ni mostrar ningún tipo de salida, sino servir de enlace entre los modelos y las vistas para implementar las diversas necesidades del desarrollo.

II.2.4 Software de desarrollo

II.2.4.1 SQLite



Ilustración 5: SQLite

El uso de las bases de datos ya se ha extendido de los servidores hacia los dispositivos móviles. El desarrollo constante de la tecnología conjuntamente con los nuevos requerimientos de las empresas ha llevado a crear diversos métodos de almacenamiento de información en dispositivos móviles, embebidos y empotrados.

La demanda de bases de datos para dispositivos móviles como PDAs y teléfonos celulares ha crecido exponencialmente en los últimos años debido a la necesidad de las empresas de tener la información al instante de lo que sucede en el campo y así responder más rápidamente ante la competencia. Esta necesidad ha provocado que el almacenamiento de los datos en estos dispositivos haya mejorado tanto en capacidad como en herramientas. Gracias a esto, actualmente contamos con diversas opciones de manejadores de bases de datos para móviles, y una de mis favoritas es SQLite, que es en la que se enfoca este artículo.

¿Qué es SQLite?

SQLite es una herramienta de software libre, que permite almacenar información en dispositivos empujados de una forma sencilla, eficaz, potente, rápida y en equipos con pocas capacidades de hardware, como puede ser una PDA o un teléfono celular. SQLite implementa el estándar SQL92 y también agrega extensiones que facilitan su uso en cualquier ambiente de desarrollo. Esto permite que SQLite soporte desde las consultas más básicas hasta las más complejas del lenguaje SQL, y lo más importante es que se puede usar tanto en dispositivos móviles como en sistemas de escritorio, sin necesidad de realizar procesos complejos de importación y exportación de datos, ya que existe compatibilidad al 100% entre las diversas plataformas disponibles, haciendo que la portabilidad entre dispositivos y plataformas sea transparente.

SQLite apareció en mayo del año 2000 de la mano de su creador D. Richard Hip, quién ha liberado las diferentes versiones de SQLite en base a la licencia GPL por lo que su código es de dominio público y puede ser modificado por cualquier persona. Gracias a esto, SQLite ha sido mejorada a lo largo de 7 años por un gran número de colaboradores y también ha sido migrada a diversas plataformas.

Características

- Estas son algunas de las características principales de SQLite:
- La base de datos completa se encuentra en un solo archivo.
- Puede funcionar enteramente en memoria, lo que la hace muy rápida.
- Tiene un footprint menor a 230KB.
- Es totalmente auto contenida (sin dependencias externas).
- Cuenta con librerías de acceso para muchos lenguajes de programación.
- Soporta texto en formato UTF-8 y UTF-16, así como datos numéricos de 64 bits.

- Soporta funciones SQL definidas por el usuario (UDF).
- El código fuente es de dominio público y se encuentra muy bien documentado.

II.2.4.2 Android Studio



Ilustración 6: Android Studio

Android Studio fue anunciado en el año 2013 en la conferencia de Google I/O. Fue creado para reemplazar a Eclipse, la plataforma que se usaba para la creación de aplicaciones y que todavía hoy en día es utilizada por muchos programadores. De esta manera, con Android Studio, Google consigue su propio IDE para el desarrollo de aplicaciones, pudiendo instalar todo el SDK para desarrollar apps específicas adaptadas a la mayor parte de versiones.

Nace el 16 de mayo de 2013 en la Google I/O.

Primera versión estable en diciembre de 2014.

Disponible para Windows / Mac / Linux.

Android, según diversas fuentes, tiene una cuota de mercado de más del 80%. Con el lanzamiento de este programa, Google se beneficia de tener su propio creador de aplicaciones para Android y actualmente ya cuenta con la versión Android Studio 2.2

Preview. Obviamente, al ser de Google cuenta con muchas ventajas, como la de tener siempre un software actualizado y con muchas sorpresas.

En la actualidad Android Studio es la plataforma que se postula como el más completo IDE para desarrollar aplicaciones Android con muchas características que destacan de los otros programas usados para este trabajo, está basado en IntelliJ y puede ser descargado de forma gratuita a través de la licencia de Apache 2.0.

Cuenta con una estructura simple que permite organizar los proyectos de manera que facilite su ubicación y su publicación, como también un entorno para desarrollar más potente, fácil e intuitivo. Permite ver el desarrollo a tiempo real de las aplicaciones y las pantallas en las que será usada la aplicación, y a su vez nos ofrece plantillas para diferentes elementos para programar como el uso de mapas.

En resumen Android Studio hace de escritorio de trabajo para los desarrolladores teniendo fácil accesibilidad a sus carpetas, archivos y lo que estén usando para crear una aplicación, este programa es totalmente actual y moderno y aunque las aplicaciones estén escritas en lenguaje Java se pueden compilar y dejarlos como un archivo .apk de una manera muy simple.

Aunque muchos expertos siguen inclinando su preferencia por Eclipse, la gran mayoría apuesta por esta nueva opción que brinda Google mucho más estable, actual y que dio como resultado que los asistentes para la adaptación y configuración de los dispositivos móviles, smartwatch y Google Glass, TV o autos destaquen de otros por su compatibilidad.

II.2.4.3 MySQL



Ilustración 7: MySQL

MySQL es un caso particular, pues se trata de un programa de licencia open-source y gratuito pero que, sin embargo, está mantenido por una empresa, MySQL AB, con sede en Suecia. El código fuente de MySQL está sólo relativamente abierto y disponible para modificaciones, puesto que es la empresa MySQL AB la que contrata y coordina los trabajos de mantenimiento del producto. No obstante, los trabajadores contratados, procedentes de todo el mundo, son usuarios del producto que realizan sus encargos a través de Internet.

Orígenes

El origen de MySQL se remonta a la década de los ochenta. Michael Widenius, también conocido como Monty, un joven programador que realizaba complejas aplicaciones en lenguaje BASIC, al no encontrar un sistema de almacenamiento de archivos que le resultara satisfactorio, pensó en construir el suyo propio. Años después, en 1995, y en colaboración con David Axmark, Widenius desarrolló un producto que básicamente era el resultado de sus investigaciones, más dos aportaciones nuevas: el uso del

lenguaje SQL y la accesibilidad a través de Internet. Así nació MySQL y también la empresa MySQL AB.

Evolución

La evolución de MySQL se produce con las sugerencias de los usuarios, canalizadas por la empresa MySQL AB, que contrata a programadores de todo el mundo para que, a través de Internet, vayan ampliando y mejorando el producto. Las versiones, como es costumbre en este tipo de software libre, se van colgando en Internet para que los usuarios puedan disponer de ellas.

Sin embargo, también como es habitual, hay que distinguir entre versiones de prueba o beta y versiones estables o de producción. Las versiones de prueba o beta están a disposición de los usuarios para ser testadas en busca de posibles mejoras. Por su parte, las versiones estables o de producción son las que ya están preparadas para ser utilizadas a pleno rendimiento.

En la actualidad, 4.1. 14 es la última versión estable. Esta versión dispone de características como gestión de transacciones, integridad referencial (bajo ciertas condiciones), uso de índices, incorporación de librerías, etcétera. La versión de prueba actualmente disponible es la 5.0, que empieza a incorporar elementos hasta ahora no incluidos en MySQL, como el uso de procedimientos, cursores, triggers o vistas. Existe una versión incluso más reciente, la 5.1, pero que aún es un proyecto. Esta versión promete mejoras en la seguridad y en la gestión de claves ajenas, aunque todavía no se conoce cómo se plasmarán dichas características en posteriores versiones estables.

II.3 Materiales y métodos

Para el desarrollo de la Aplicación Móvil para Control de Animales de Granja “Si-Gan” es necesario tener herramientas adecuadas para realizar el proyecto.

En las siguientes tablas se presentaran las herramientas necesarias tanto de software como hardware.

Hardware	
Lenovo Z40	8 GB de memoria PC3-12800 DDR3L SDRAM 1600 MHz
Disco duro de 1 TB	2 puertos USB 2.0, 1 puerto USB 3.0, 1 puerto HDMI, 1 puerto de red.
Sistema Operativo Ubuntu 17.04	Procesador Intel® Core™ i5- 4200U 4º generación (2,60 GHz 1600 MHz 3MB)
Bluetooth® 4.0, Lenovo BGN Wireless, LAN 10 / 100 / 1000M	Cámara HD 720p (1M píxel)

Tabla 1. Hardware utilizado

Software	Descripción
Android Studio 2.3.2	Es el entorno de desarrollo integrado oficial para la plataforma Android. Fue anunciado el 16 de mayo de 2013 en la conferencia Google I/O, y reemplazó a Eclipse como el IDE oficial para el desarrollo de aplicaciones para Android.
SQLite versión 3.19.3	Sistema de gestión de bases de datos relacional compatible con ACID, contenida en una relativamente pequeña (~275 kiB) ² biblioteca escrita en C. SQLite es un proyecto de dominio público ¹ creado por D. Richard Hipp
MySQL 10.1.10-MariaDB	Sistema de gestión de bases de datos relacional desarrollado bajo licencia dual GPL/Licencia comercial por Oracle Corporation y

	está considerada como la base datos open source más popular del mundo, ¹ ² y una de las más populares en general junto a Oracle y Microsoft SQL Server, sobre todo para entornos de desarrollo web.
Genymotion 2.9.0	Emulador de Android que aprovecha la arquitectura x86 para ejecutar de forma fluida y rápida distintos dispositivos Android.
SQLyog v8.05	Es una excelente interfaz gráfica diseñada especialmente para trabajar de forma más rápida y cómoda con el servidor de base de datos MySQL. El programa está orientado a usuarios que ya tienen ciertos conocimientos de SQL y necesitan un intérprete gráfico sin excesivas florituras, ágil y funcional.
WPS Office 10.1.0.5707	Es una suite ofimática para Microsoft Windows, Linux, Android OS y iOS. Creada por el desarrollador de software Kingsoft Office en Zhuhai, China. Incluye los siguientes componentes: Writer, Presentation y Spreadsheet.
Umbrello UML modeller Versión 2.21.3	Es una herramienta libre para crear y editar diagramas UML, que ayuda en el proceso del desarrollo de software. Fue desarrollada por Paul Hensgen, y está diseñado principalmente para KDE, aunque funciona en otros entornos de escritorio.

Tabla 2 Software utilizado

II.3.1 Metodología utilizada.

Modelo en cascada

Modelo en Cascada, también llamado Lineal secuencial, es el enfoque metodológico que ordena rigurosamente las etapas del proceso para el desarrollo de software, de tal forma que el inicio de cada etapa debe esperar a la finalización de la etapa anterior.

Fases del Modelo en Cascada

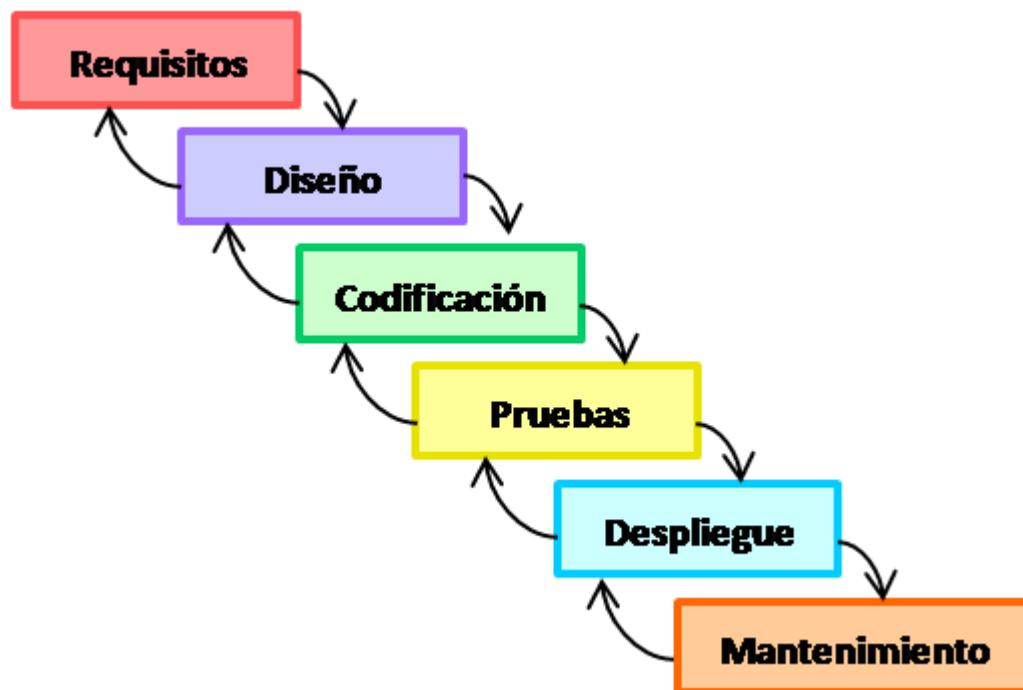


Ilustración 8: Modelo en cascada

Análisis de requisitos: En esta fase se analizan las necesidades de los usuarios finales del software para determinar qué objetivos debe cubrir. De esta fase surge una memoria llamada SRD (documento de especificación de requisitos), que contiene la especificación completa de lo que debe hacer el sistema sin entrar en detalles internos. Es importante señalar que en esta etapa se debe consensuar todo lo que se requiere

del sistema y será aquello lo que seguirá en las siguientes etapas, no pudiéndose requerir nuevos resultados a mitad del proceso de elaboración del software.

Diseño del Sistema: Se descompone y organiza el sistema en elementos que puedan elaborarse por separado, aprovechando las ventajas del desarrollo en equipo. Como resultado surge el SDD (Documento de Diseño del Software), que contiene la descripción de la estructura relacional global del sistema y la especificación de lo que debe hacer cada una de sus partes, así como la manera en que se combinan unas con otras. Es conveniente distinguir entre diseño de alto nivel o arquitectónico y diseño detallado. El primero de ellos tiene como objetivo definir la estructura de la solución (una vez que la fase de análisis ha descrito el problema) identificando grandes módulos (conjuntos de funciones que van a estar asociadas) y sus relaciones. Con ello se define la arquitectura de la solución elegida. El segundo define los algoritmos empleados y la organización del código para comenzar la implementación.

Diseño del Programa: Es la fase en donde se realizan los algoritmos necesarios para el cumplimiento de los requerimientos del usuario así como también los análisis necesarios para saber que herramientas usar en la etapa de Codificación.

Codificación: Es la fase en donde se implementa el código fuente, haciendo uso de prototipos así como de pruebas y ensayos para corregir errores. Dependiendo del lenguaje de programación y su versión se crean las bibliotecas y componentes reutilizables dentro del mismo proyecto para hacer que la programación sea un proceso mucho más rápido.

Pruebas: Los elementos, ya programados, se ensamblan para componer el sistema y se comprueba que funciona correctamente y que cumple con los requisitos, antes de ser entregado al usuario final.

Verificación: Es la fase en donde el usuario final ejecuta el sistema, para ello el o los programadores ya realizaron exhaustivas pruebas para comprobar que el sistema no falle.

Mantenimiento: Una de las etapas más críticas, ya que se destina un 75% de los recursos, es el mantenimiento del Software ya que al utilizarlo como usuario final puede ser que no cumpla con todas nuestras expectativas.

Ventajas

- El horario se establece con los plazos normalmente adecuados para cada etapa de desarrollo.
- Este proceso conduce a entregar el proyecto a tiempo.
- Es sencilla y facilita la gestión de proyectos.
- Permite tener bajo control el proyecto.
- Limita la cantidad de interacción entre equipos que se produce durante el desarrollo.

Desventajas

- No conocer si la solución es correcta hasta estar cerca de su lanzamiento.
- Poco tiempo para corregir fallas.
- Depuración complicada.
- Los cambios introducidos durante el desarrollo pueden confundir al equipo profesional en las etapas tempranas del proyecto.

- No es frecuente que el cliente o usuario final explicita clara y completamente los requisitos.
- Es necesaria la paciencia del cliente.
- El cliente podría detectar un error.
- El proceso es lento y pesado.

II.4. Procesamiento de resultados, análisis y discusión.

Caso de uso

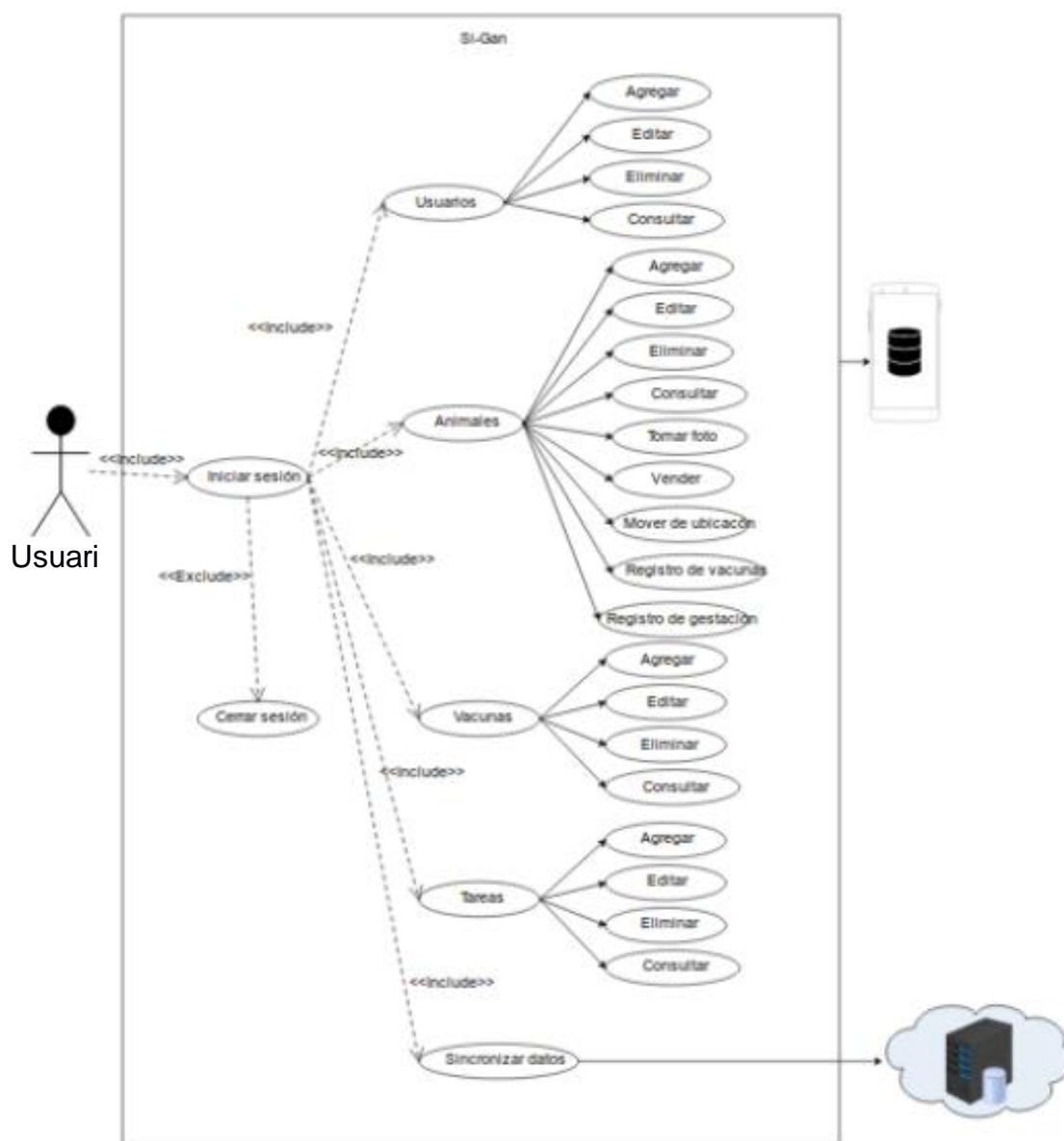


Ilustración 9: Caso de uso

Descripción de catalogo

Nombre de caso de uso Usuario	
Actor	Usuario
Objetivo:	<ul style="list-style-type: none"> ✓ Consultar usuarios, animales, tareas, vacunas, gestaciones, alimentación. ✓ Altas de usuarios, animales, tareas, vacunas, gestaciones, alimentación. ✓ Bajas de usuarios, animales, tareas, vacunas, gestaciones, alimentación. ✓ Modificaciones de usuarios, animales, tareas, vacunas, gestaciones, alimentación.
Precondiciones:	❖ El usuario deberá iniciar sesión
Flujo básico:	
Alta:	1. Agregar datos
Baja:	2. Eliminar datos
Modificar:	4. Modificar los datos
Consulta:	5. Ver los datos de los catálogos
Flujo alternativo:	1. Iniciar sesión como administrador. 2. Realizar consulta.

Tabla 3: Descripción de catalogo

Modelo relacional.

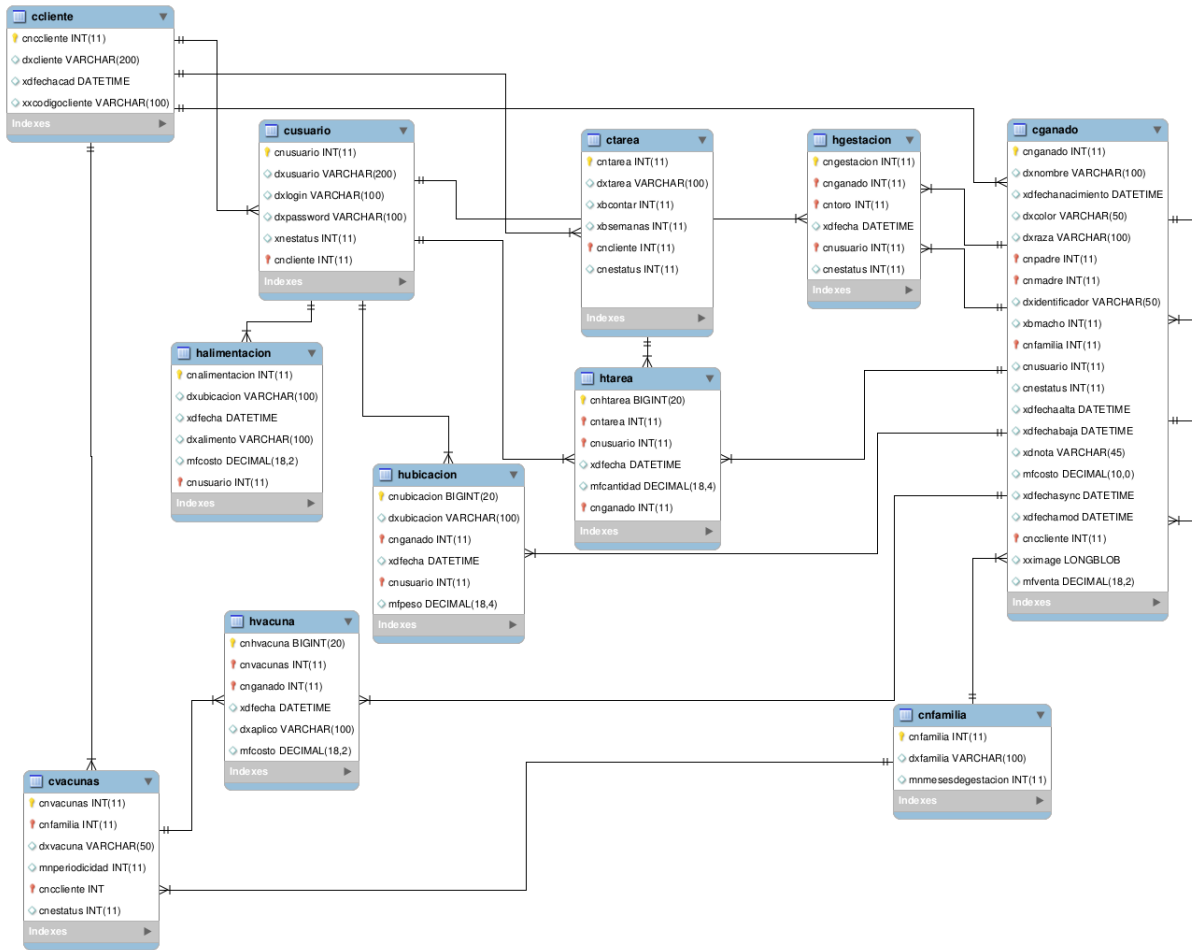


Ilustración 10: Modelo relacional

Diccionario de datos.

Nombre de la base de datos: si_gan

Nombre de la tabla: **ccliente**

Descripción: almacena información del cliente

Campo	Tipo de dato	Tamaño	PK	FK	Nulo	Descripción
cnccliente	Int	11	Si	--	No	Llave primaria de la tabla ccliente
dxcliente	Varchar	200	--	--	Si	Nombre completo del cliente
xdfechacad	Datetime		--	--	Si	Fecha en que caduca la licencia
Xxcodigocliente	Varchar	100	--	--	Si	Código del cliente

Tabla 4 ccliente

Nombre de la tabla: **cganado**

Descripción: almacena información de los animales

Campo	Tipo de dato	Tamaño	PK	FK	Nulo	Descripción
cnganado	Int	11	Si	--	No	Llave primaria de la tabla cganado
dxnombre	Varchar	100	--	--	Si	Nombre del animal
Xdfechana	Datetime		--	--	Si	Fecha de nacimiento del

cimiento						animal
dxcolor	Varchar	50	--	--	Si	Color del animal
Dxraza	Varchar	100	--	--	Si	Raza del animal
cnpadre	Int	11	Si	Si	Si	Padre del animal Llave foránea a la tabla cganado --> campo cnganado
cnmadre	Int	11	Si	Si	Si	Madre del animal Llave foránea a la tabla cganado --> campo cnganado
Dxidentifica dor	Varchar	50	--	--	Si	Identificador del animal
xbmacho	Int	11	--	--	Si	Sexo del animal 0 es macho 1 es hembra
cnfamilia	Int	11	Si	Si	No	Llave foránea a la tabla cnfamilia --> campo cnfamilia
cnusuario	Int	11	Si	Si	No	Llave foránea a la tabla cusuario --> campo cusuario
cnestatus	Int	11	--	--	Si	Estatus del animal 0 es eliminado

						1 es vivo 2 es muerto 3 es vendido
xdfechaalta	Datetime		--	--	Si	Fecha en que fue dado de alta el animal
Xdfechabaj a	Datetime		--	--	Si	Fecha en que se le dio de baja
Xdnota	Varchar	50	--	--	Si	Nota para el animal
mfcosto	Decimal	10,0	--	--	Si	Valor del animal
Xdfechamo d	Datetime		--	--	Si	Fecha de modificación del animal
Xdfechasy nc	Datetime		--	---	Si	Fecha en que fue sincronizado
cnccliente	Int	11	Si	Si	No	Llave foránea a la tabla ccliente --> campo cnccliente
xximage	Longblob		--	---	Si	Fotografía del animal
mfventa	Decimal	18,2	--	--	Si	Precio en el que fue vendido el animal

Tabla 5 cganado

Nombre de la tabla: **cnfamilia**

Descripción: almacena información de la familia

Campo	Tipo de dato	Tamaño	PK	FK	Nulo	Descripción
cnfamilia	Int	11	Si	--	No	Llave primaria de la tabla cnfamilia
dxfamilia	Varchar	100	--	--	Si	Nombre de la familia
mnmesesdegestacion	Int	11	--	--	Si	Meses de gestación

Tabla 6 cnfamilia

Nombre de la tabla: **ctarea**

Descripción: almacena información de las tareas

Campo	Tipo de dato	Tamaño	PK	FK	Nulo	Descripción
Cntarea	Int	11	Si	--	No	Llave primaria de la tabla ctarea
Dxtarea	Varchar	100	--	--	Si	Tarea
xbcontar	Int	11	--	--	Si	Si es contable la tarea 0 no es contable 1 si es contable
xbsemanas	Int	11	--	--	Si	Cada cuantas semanas se realiza la tarea

cnccliente	Int	11	Si	Si	No	Llave foránea a la tabla ccliente --> campo cnccliente
cnestatus	Int	11	--	--	Si	Estatus de la tarea 0 es eliminada 1 es activo 2 es inactivo

Tabla 7 ctarea

Nombre de la tabla: **cusuario**

Descripción: almacena información de los usuarios

Campo	Tipo de dato	Tamaño	PK	FK	Nulo	Descripción
cusuario	Int	11	Si	--	No	Llave primaria de la tabla cusuario
dxusuario	Varchar	200	--	--	Si	Nombre completo del usuario
Dxlogin	Varchar	100	--	--	Si	Usuario para el login
Dxpassword	Varchar	100	--	--	Si	Contraseña para el login
xnestatus	Int	11	--	--	Si	Estatus del usuario 0 es eliminado 1 es activo 2 es inactivo
cnccliente	Int	11	Si	Si	No	Llave foránea a la tabla

						ccliente --> campo cnccliente
--	--	--	--	--	--	-------------------------------

Tabla 8 cusuario

Nombre de la tabla: **cvacunas**

Descripción: almacena información de la familia

Campo	Tipo de dato	Tamaño	PK	FK	Nulo	Descripción
cnvacunas	Int	11	Si	--	No	Llave primaria de la tabla cvacunas
cnfamilia	Int	11	Si	Si	No	Llave foránea a la tabla cnfamilia --> campo cnfamilia
dxvacuna	Varchar	50	--	--	Si	Nombre de la vacuna
Mnperiodicidad	Int	11	--	--	Si	Cada cuantos días se aplica la vacuna
cnccliente	Int	11	Si	Si	No	Llave foránea a la tabla ccliente --> campo cnccliente
cnestatus	Int	11	--	--	Si	Estatus de la vacuna 0 es eliminada 1 es activa 2 es inactiva

Tabla 9 cvacunas

Nombre de la tabla: **halimentacion**

Descripción: almacena información de los alimentos que se le da al animal que está en una ubicación de estabulación

Campo	Tipo de dato	Tamaño	PK	FK	Nulo	Descripción
Cnalimentacion	Int	11	Si	--	No	Llave primaria de la tabla halimentacion
Dxubicacion	Varchar	100	--	--	Si	Nombre de la ubicación
xdfecha	Datetime		--	--	Si	Fecha en que se dio el alimento
dxalimento	Varchar	100	--	--	Si	Nombre del alimento
mfcosto	Decimal	18,2	--	--	Si	Costo total del alimento
cnusuario	Int	11	Si	Si	No	Llave foránea a la tabla cusuario --> campo cusuario
mfcantidad	Decimal	18,2	--	--	Si	Almacena la cantidad de alimento

Tabla 10 halimentacion

Nombre de la tabla: **hgestacion**

Descripción: almacena información de los alimentos que se le da al animal que está en una ubicación de estabulación

Campo	Tipo de dato	Tamaño	PK	FK	Nulo	Descripción
-------	--------------	--------	----	----	------	-------------

Cngestacion	Int	11	Si	--	No	Llave primaria de la tabla hgestacion
cnganado	Int	11	Si	Si	No	Llave foránea a la tabla cganado --> campo cnganado
Cntoro	Int	11	Si	Si		Llave foránea a la tabla cganado --> campo cnganado
xdfecha	Datetime		--	--	Si	Fecha en que reviso
cnusuario	Int	11	Si	Si	No	Llave foránea a la tabla cusuario --> campo cnusuario
cnestatus	Int	11	--	--	Si	0 no preñada, 1 estimulada 2 preñada 3 aborto

Tabla 11 hgestacion

Nombre de la tabla: **htarea**

Descripción: almacena información de las tareas que se hacen

Campo	Tipo de dato	Tamaño	PK	FK	Nulo	Descripción
cnhtarea	Bigint	20	Si	--	No	Llave primaria de la tabla htarea

cntarea	Int	11	Si	Si	No	Llave foránea a la tabla cntarea --> campo cntarea
dxfecha	Datetime		--	--	Si	Fecha en que se realizó la tarea
mfcantidad	Decimal	18,4	--	--	Si	Se guarda la cantidad
cnusuario	Int	11	Si	Si	No	Llave foránea a la tabla cusuario --> campo cnusuario
cnganado	Int	11	Si	Si	No	Llave foránea a la tabla cganado --> campo cnganado

Tabla 12 htarea

Nombre de la tabla: **hubicacion**

Descripción: almacena información de donde se encuentra el animal

Campo	Tipo de dato	Tamaño	PK	FK	Nulo	Descripción
Cnubicacion	Bigint	20	Si	--	No	Llave primaria de la tabla hubicacion
Dxubicacion	Varchar	100	--	--	Si	Nombre de la ubicación
cnganado	Int	11	Si	Si	No	Llave foránea a la tabla cganado --> campo cnganado.
xdfecha	Int	11	--	--	Si	Fecha en que se movió el

						animal a esa ubicación
cnusuario	Int	11	Si	Si	No	Llave foránea a la tabla cnusuario --> campo cnusuario
mfpeso	Int	11	--	--	Si	Peso del animal pero solo es utilizado cuando está en una ubicación de estabulación

Tabla 13 hubicacion

Nombre de la tabla: **hvacuna**

Descripción: almacena información de las tareas que se hacen

Campo	Tipo de dato	Tamaño	PK	FK	Nulo	Descripción
cnhvacuna	Bigint	20	Si	--	No	Llave primaria de la tabla hvacuna
cnvacunas	Int	11	Si	Si	No	Llave foránea a la tabla cvacunas --> campo cnvacunas
xdfecha	Datetime		--	--	Si	Fecha en que se puso la vacuna
dxaplico	Varchar	100	--	--	Si	Nombre de la persona que aplico la vacuna
mfcosto	Decimal	18,2	--	--	Si	Costo de la vacuna

Tabla 14 hvacuna

Script de la base de datos

```
CREATE DATABASE si_gan;

USE si_gan;

CREATE TABLE `ccliente` (
  `cnccliente` INT(11) NOT NULL,
  `dxcliente` VARCHAR(200) NULL,
  `xdfechacad` DATETIME NULL,
  `xxcodigocliente` VARCHAR(100) NULL,
  PRIMARY KEY (`cnccliente`))
ENGINE = InnoDB;

CREATE TABLE `cnfamilia` (
  `cnfamilia` INT(11) NOT NULL,
  `dxfamilia` VARCHAR(100) NULL,
  `mnmesesdegestacion` INT(11) NULL,
  PRIMARY KEY (`cnfamilia`))
ENGINE = InnoDB;

CREATE TABLE `cganado` (
```

```
`cnganado` INT(11) NOT NULL,  
`dxnombre` VARCHAR(100) NULL,  
`xdfechanacimiento` DATETIME NULL,  
`dxcolor` VARCHAR(50) NULL,  
`dxraza` VARCHAR(100) NULL,  
`cnpadre` INT(11) NOT NULL,  
`cnmadre` INT(11) NOT NULL,  
`dxidentificador` VARCHAR(50) NULL,  
`xbmacho` INT(11) NULL,  
`cnfamilia` INT(11) NOT NULL,  
`cnusuario` INT(11) NULL,  
`cnestatus` INT(11) NULL,  
`xdfechaalta` DATETIME NULL,  
`xdfechabaja` DATETIME NULL,  
`xdnota` VARCHAR(45) NULL,  
`mfcosto` DECIMAL(10,0) NULL,  
`xdfechasync` DATETIME NULL,  
`xdfechamod` DATETIME NULL,  
`cnccliente` INT(11) NOT NULL,  
`xximage` LONGBLOB NULL,  
`mfventa` DECIMAL(18,2) NULL,
```

```

PRIMARY KEY (`cnganado`, `cnfamilia`, `cnpadre`, `cnccliente`, `cnmadre`),
CONSTRAINT `fk_cganado_cnfamilia1`
  FOREIGN KEY (`cnfamilia`)
    REFERENCES `cnfamilia` (`cnfamilia`),
CONSTRAINT `fk_cganado_cganado1`
  FOREIGN KEY (`cnpadre`)
    REFERENCES `cganado` (`cnganado`),
CONSTRAINT `fk_cganado_cganado2`
  FOREIGN KEY (`cnmadre`)
    REFERENCES `cganado` (`cnganado`),
CONSTRAINT `fk_cganado_ccliente1`
  FOREIGN KEY (`cnccliente`)
    REFERENCES `ccliente` (`cnccliente`))
ENGINE = InnoDB;

CREATE TABLE `ctarea` (
  `cntarea` INT(11) NOT NULL,
  `dxtarea` VARCHAR(100) NULL,
  `xbcontar` INT(11) NULL,
  `xbsemanas` INT(11) NULL,
  `cncliente` INT(11) NOT NULL,

```

```

`cnestatus` INT(11) NULL,

PRIMARY KEY (`cntarea`, `cncliente`),

CONSTRAINT `fk_ctarea_ccliente1`

FOREIGN KEY (`cncliente`)

REFERENCES `ccliente` (`cnccliente`))

ENGINE = InnoDB;

```

```

CREATE TABLE `cusuario` (

`cnusuario` INT(11) NOT NULL,

`dxusuario` VARCHAR(200) NULL,

`dxlogin` VARCHAR(100) NULL,

`dxpassword` VARCHAR(100) NULL,

`xnestatus` INT(11) NULL,

`cncliente` INT(11) NOT NULL,

PRIMARY KEY (`cnusuario`, `cncliente`),

CONSTRAINT `fk_cusuario_ccliente`

FOREIGN KEY (`cncliente`)

REFERENCES `ccliente` (`cnccliente`))

ENGINE = InnoDB;

```

```

CREATE TABLE `cvacunas` (

```

```

`cnvacunas` INT(11) NOT NULL,

`cnfamilia` INT(11) NOT NULL,

`dxvacuna` VARCHAR(50) NULL,

`mnperiodicidad` INT(11) NULL,

`cnccliente` INT NOT NULL,

`cnestatus` INT(11) NULL,

PRIMARY KEY (`cnvacunas`, `cnfamilia`, `cnccliente`),

CONSTRAINT `fk_cvacunas_ccliente1`

FOREIGN KEY (`cnccliente`)

REFERENCES `ccliente` (`cnccliente`),

CONSTRAINT `fk_cvacunas_cnfamilia1`

FOREIGN KEY (`cnfamilia`)

REFERENCES `cnfamilia` (`cnfamilia`))

ENGINE = InnoDB;

CREATE TABLE `halimentacion` (

`cnalimentacion` INT(11) NOT NULL,

`dxubicacion` VARCHAR(100) NULL,

`xdfecha` DATETIME NULL,

`dxalimento` VARCHAR(100) NULL,

`mfcosto` DECIMAL(18,2) NULL,

```



```

`cnusuario` INT(11) NOT NULL,

`mfcantidad` DECIMAL(18,2) NULL,

PRIMARY KEY (`cnalimentacion`, `cnusuario`),

CONSTRAINT `fk_halimentacion_cusuario1`

FOREIGN KEY (`cnusuario`)

REFERENCES `cusuario` (`cnusuario`))

ENGINE = InnoDB;

CREATE TABLE `hgestacion` (

`cngestacion` INT(11) NOT NULL,

`cnganado` INT(11) NOT NULL,

`cntoro` INT(11) NOT NULL,

`xdfecha` DATETIME NULL,

`cnusuario` INT(11) NOT NULL,

`cnestatus` INT(11) NULL,

PRIMARY KEY (`cngestacion`, `cnganado`, `cntoro`, `cnusuario`),

CONSTRAINT `fk_hgestacion_cganado1`

FOREIGN KEY (`cnganado`)

REFERENCES `cganado` (`cnganado`),

CONSTRAINT `fk_hgestacion_1`

FOREIGN KEY (`cntoro`)

```

```

REFERENCES `cganado` (`cnganado`),

CONSTRAINT `fk_hgestacion_cusuario1`

FOREIGN KEY (`cusuario`)

REFERENCES `cusuario` (`cusuario`))

ENGINE = InnoDB;

CREATE TABLE `htarea` (

`cnhtarea` BIGINT(20) NOT NULL,

`cntarea` INT(11) NOT NULL,

`cusuario` INT(11) NOT NULL,

`xdfecha` DATETIME NULL,

`mfcantidad` DECIMAL(18,4) NULL,

`cnganado` INT(11) NOT NULL,

PRIMARY KEY (`cnhtarea`, `cntarea`, `cusuario`, `cnganado`),

CONSTRAINT `fk_htarea_ctarea1`

FOREIGN KEY (`cntarea`)

REFERENCES `ctarea` (`cntarea`),

CONSTRAINT `fk_htarea_cusuario1`

FOREIGN KEY (`cusuario`)

REFERENCES `cusuario` (`cusuario`),

CONSTRAINT `fk_htarea_cganado1`

```

```

FOREIGN KEY (`cnganado`)
REFERENCES `cnganado` (`cnganado`))

ENGINE = InnoDB;

CREATE TABLE `hubicacion` (
  `cnubicacion` BIGINT(20) NOT NULL,
  `dxubicacion` VARCHAR(100) NULL,
  `cnganado` INT(11) NOT NULL,
  `xdfecha` DATETIME NULL,
  `cnusuario` INT(11) NOT NULL,
  `mfpeso` DECIMAL(18,4) NULL,
  PRIMARY KEY (`cnubicacion`, `cnganado`, `cnusuario`),
  CONSTRAINT `fk_hubicacion_cusuario1`
  FOREIGN KEY (`cnusuario`)
  REFERENCES `cusuario` (`cnusuario`),
  CONSTRAINT `fk_hubicacion_cganado1`
  FOREIGN KEY (`cnganado`)
  REFERENCES `cnganado` (`cnganado`))

ENGINE = InnoDB;

CREATE TABLE `hvacuna` (

```

```
`cnhvacuna` BIGINT(20) NOT NULL,  
  
`cnvacunas` INT(11) NOT NULL,  
  
`cnganado` INT(11) NOT NULL,  
  
`xdfecha` DATETIME NULL,  
  
`dxaplico` VARCHAR(100) NULL,  
  
`mfcosto` DECIMAL(18,2) NULL,  
  
PRIMARY KEY (`cnhvacuna`, `cnganado`, `cnvacunas`),  
  
CONSTRAINT `fk_hvacuna_cvacunas1`  
  
FOREIGN KEY (`cnvacunas`)  
  
REFERENCES `cvacunas` (`cnvacunas`),  
  
CONSTRAINT `fk_hvacuna_cganado1`  
  
FOREIGN KEY (`cnganado`)  
  
REFERENCES `cganado` (`cnganado`))  
  
ENGINE = InnoDB;
```

Wireframes de la aplicación

En la siguiente imagen se puede ver lo que es el login donde se pide el usuario y contraseña del mismo, así como una opción para recordar ambos datos.

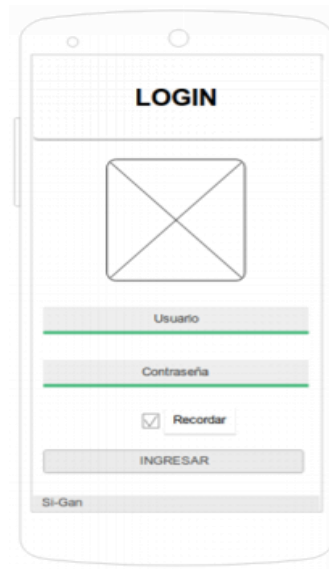


Ilustración 11: Pantalla login

La siguiente pantalla es la ventana principal donde se encuentra un resumen de los catálogos de la base de datos.



Ilustración 12: Pantalla principal

La pantalla mostrada es donde agregaran nuevos usuarios, se puede apreciar los campos donde se ingresaran todos los datos correspondientes



The image shows a mobile application interface for adding a new user. At the top, there is a header bar with a back arrow on the left, the text "Si-Gan" in the center, and a plus sign on the right. Below the header, the title "Agregar Usuario" is displayed. The form consists of several input fields: "Nombre" with the placeholder "Ingrese su nombre completo", "Usuario" with the placeholder "Ingrese el usuario", and "Password" with a masked input (dots). Below these is a dropdown menu for "Estatus" with "Activo" selected. Each input field has a green underline.

Ilustración 13: Pantalla agregar usuario

Anexo 1.

MANUAL DE INSTALACIÓN.

APLICACIÓN MÓVIL PARA EL CONTROL DE ANIMALES DE GRANJA.



Autor:
C. Gabriel Moreno Reyes

Índice “manual de instalación”

REQUERIMIENTOS DE HARDWARE Y SOFTWARE.	2
Requerimientos mínimos.	2
Hardware.	2
Software	2
INSTALACIÓN.....	3
Instalación de la aplicación móvil.....	3

Índice ilustraciones

Ilustración 1 Aplicación Si-Gan.....	3
Ilustración 2 Permisos de la aplicación	4
Ilustración 3 Instalando	5
Ilustración 4 instalación finalizada	5

REQUERIMIENTOS DE HARDWARE Y SOFTWARE.

Requerimientos mínimos.

Para el funcionamiento correcto y óptimo de la aplicación Si-Gan es necesario tener los requerimientos siguientes mínimos.

Hardware.

- ✓ Memoria RAM: 512 MG
- ✓ Memoria Interna: 1 GB como mínimo
- ✓ Conexión a Internet mediante datos móviles o Wifi
- ✓ Cámara de 1 Megapixel.
- ✓ Pantalla táctil

Software

- ✓ Versión de Android mínimo 4.1.2 JELLY BEAN.
- ✓ Controlador de cámara.

INSTALACIÓN

Instalación de la aplicación móvil.

Teniendo en cuenta las características que se debe tener su teléfono celular se procederá a la instalación, para ello es necesario tener el archivo llamado Si-Gan.apk.

Nota.

¿Qué significa APK?

Empecemos por lo más básico, para aquellos usuarios poco relacionados con tecnicismos. Se le llama APK a los archivos con extensión que corresponde a .apk y cuyo significado completo es Application Package File, el cual fue pensado para el sistema operativo Android.

¿Para qué sirve?

A través de estos archivos es posible instalar programas que han sido comprimidos (similar a un ZIP), funcionan tanto en teléfonos como en tablets en general en cualquier plataforma con estructura similar.

1. Trasládese hacia la carpeta contenedora del archivo Si-Gan.apk que le fue proporcionado. Una vez estando en la carpeta contenedora del archivo antes mencionado es necesario dar clic sobre él.

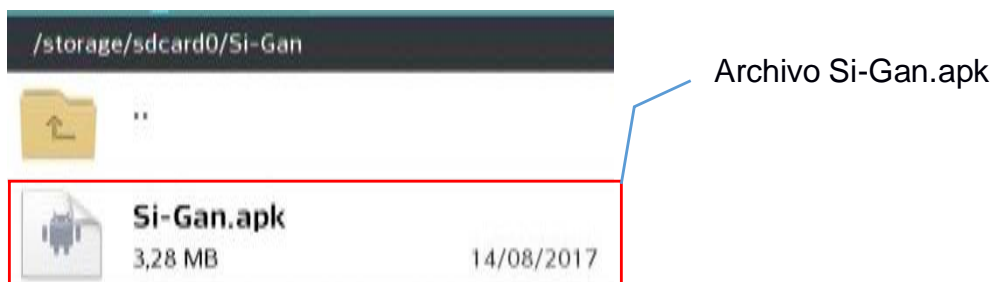


Ilustración 14 Aplicación Si-Gan

2. Al darle clic a el archivo lo mandara a la siguiente pantalla que se puede apreciar, donde se le dará a conocer los permisos que necesitara la aplicación. Para instalar la aplicación es necesario dar clic sobre el botón instalar.

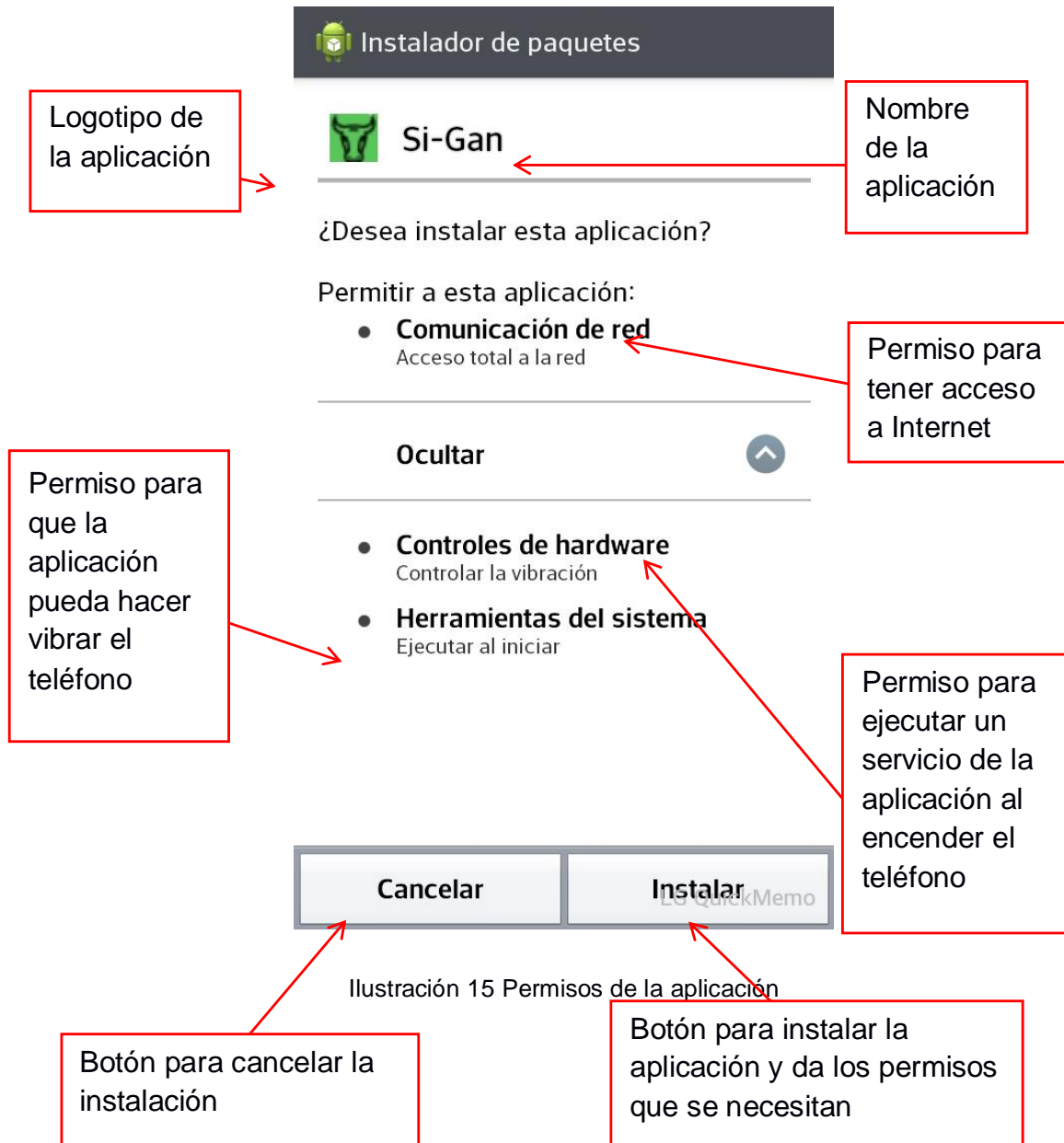


Ilustración 15 Permisos de la aplicación

3. Espere mientras se instala la aplicación en su dispositivo móvil.

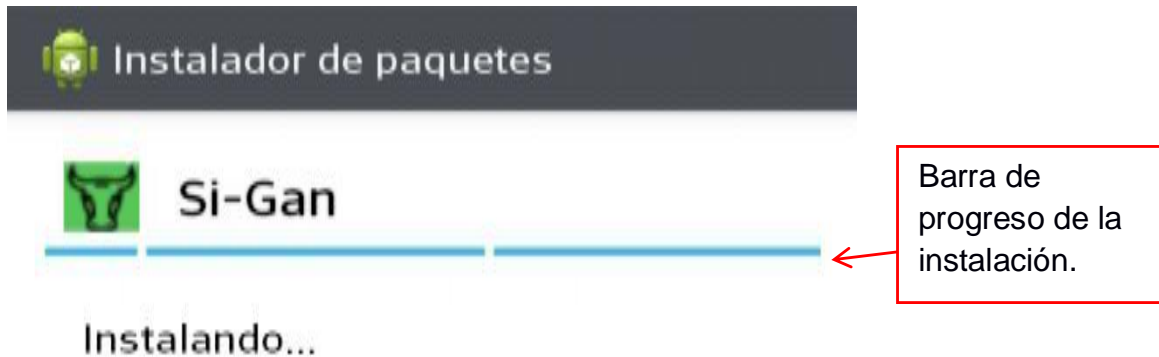


Ilustración 16 Instalando

4. Al terminar le aparecerá la pantalla siguiente donde le dice que la aplicación fue instalada satisfactoriamente. Y en la parte baja de la pantalla 2 botones, “**Listo**” para cerrar la ventana de instalación y “**Abrir**” abre la aplicación. En ambos casos la aplicación ya está instalada en su dispositivo móvil.



Ilustración 17 instalación finalizada

Anexo 2.

MANUAL DE USUARIO

APLICACIÓN MÓVIL PARA EL CONTROL DE
ANIMALES DE GRANJA.



Autor:

C. Gabriel Moreno Reyes

Índice “manual de usuarios”

Primeros pasos	1
Vista principal	4
Catálogo de animales.....	6
Catálogo ubicación	16
Catálogo alimentación.....	23
Catálogo gestación.....	27
Catálogo tarea	33
Catálogo vacunas.....	39
Catálogo usuarios.....	44

Índice de ilustraciones

Ilustración 1 Código de cliente.....	1
Ilustración 2 Descarga de datos	2
Ilustración 3 Login	3
Ilustración 4 Ventana principal.....	4
Ilustración 5 Cerrar sesión	5
Ilustración 6 Botón animales.....	6
Ilustración 7 Lista de animales vacía	6
Ilustración 8 Descripción nuevo animal.....	8
Ilustración 9 Botón tomar foto.....	9
Ilustración 10 Tomar foto	9
Ilustración 11 Fotografía del animal	9
Ilustración 12 Datos llenado del animal	10
Ilustración 13 Botón agregar.....	11
Ilustración 14 Salir	11

Ilustración 15 Botones padres	11
Ilustración 16 Seleccionar padres	12
Ilustración 17 Padres del animal	12
Ilustración 18 Lista de animales llena	13
Ilustración 19 Opciones del menú de animales	14
Ilustración 20 Datos del animal (actualizar/eliminar)	15
Ilustración 21 Botón ubicación	16
Ilustración 22 catálogo de ubicación	16
Ilustración 23 Opciones de ubicación	17
Ilustración 24 Seleccionar ubicación	17
Ilustración 25 Nueva ubicación normal	18
Ilustración 26 Nueva ubicación agregada	18
Ilustración 27 Ubicaciones	19
Ilustración 28 Mover estabulación	19
Ilustración 29 Nueva ubicación estabulación	20
Ilustración 30 Mover a estabulación	20
Ilustración 31 Ubicaciones	21
Ilustración 32 Animales por ubicación	21
Ilustración 33 Iniciar venta	22
Ilustración 34 Rellenar precio de venta	22
Ilustración 35 botón alimentación	23
Ilustración 36 Reporte de alimentación	23
Ilustración 37 Apartado de alimentación con error	24
Ilustración 38 Selección de fila	24
Ilustración 39 Nuevo alimento	24
Ilustración 40 Datos del alimento para agregar	25
Ilustración 41 Nuevo alimento	25
Ilustración 42 Datos actualizados	26
Ilustración 43 Cambio de fecha	26
Ilustración 44 Botón gestación	27
Ilustración 45 Gestación	27
Ilustración 46 Opciones del animal para la gestación	28
Ilustración 47 Revisar	28
Ilustración 48 Estatus estimulada	29
Ilustración 49 Animal en el estatus Estimulada	29

Ilustración 50 Pasos para seleccionar el padre	30
Ilustración 51 Estatus preñada	31
Ilustración 52 Estatus aborto	31
Ilustración 53 Historial de gestación.....	32
Ilustración 54 Registro de la cría	32
Ilustración 55 Botón tarea	33
Ilustración 56 lista de tareas	33
Ilustración 57 Hacer tarea	34
Ilustración 58 Botón agregar tarea	34
Ilustración 59 Agregar tarea	35
Ilustración 60 Presionar fila.....	35
Ilustración 61 Modificar tarea.....	36
Ilustración 62 Eliminar varias tareas	36
Ilustración 63 Guardar tarea realizada.....	37
Ilustración 64 Guardar tarea cuantificable	37
Ilustración 65 Menú tareas hechas	38
Ilustración 66 Lista de tareas hechas.....	38
Ilustración 67 Botón vacunas.....	39
Ilustración 68 Lista de vacunas	39
Ilustración 69 Sin Vacunas	39
Ilustración 70 Guardar Vacuna	40
Ilustración 71 Eliminar vacuna	40
Ilustración 72 Aplicar vacuna	41
Ilustración 73 Fila de vacuna	42
Ilustración 74 Datos de la vacuna	42
Ilustración 75 Vacunas realizadas.....	43
Ilustración 76 Botón usuarios	44
Ilustración 77 Lista de usuarios	44
Ilustración 78 Datos del usuario	45
Ilustración 79 Botón agregar usuario	45
Ilustración 80 Agregar usuario.....	46

PRIMEROS PASOS

1. Una vez que se tiene instalada la aplicación, se procede a abrirla, la pantalla a continuación requiere del código de cliente que le fue proporciona por la empresa. Ingresar el código en la caja de texto y presionar el botón Sincronizar. (Observación para realizar este paso es necesario contar con conexión a Internet).



Ilustración 18 Código de cliente

2. En caso de ser correcto el código de cliente ingresado iniciara la descarga de los datos que se encuentren en el servidor en caso de tenerlos. Cuando termine la descarga se cerrara la ventana.



Ilustración 19 Descarga de datos

3. En la siguiente ventana se realizara la autenticación de usuario para ello, necesitara introducir el usuario y contraseña que le proporcione la empresa. Una vez llenado ambas cajas de texto podrá iniciar sesión. En caso que los datos no sean correctos se le dará a conocer con un mensaje (Observación no se admiten las cajas de texto vacía).

The image shows a login interface for 'Si-Gar'. The background features a cow in a field under a cloudy sky. The interface has a blue header with the text 'Inicio de Sesión'. Below this is a light blue box containing the login fields. The username field is labeled 'root' and has an annotation 'Ingresar el usuario'. The password field is masked with dots and has an annotation 'Ingresar la contraseña.'. Below the password field is a checkbox labeled 'Recordar' with an annotation 'Presionar en caso que se desee guardar el usuario y contraseña, presione después de rellenar los datos'. At the bottom of the light blue box is a blue button labeled 'INGRESAR' with an annotation 'Presionar para iniciar'.

Ingresar el usuario

Ingresar la contraseña.

Presionar en caso que se desee guardar el usuario y contraseña, presione después de rellenar los datos

Presionar para iniciar

Ilustración 20 Login

VISTA PRINCIPAL

4. Pantalla principal de la aplicación Si-Gan. (Observación el resto de los catálogos serán desbloqueado cuando se realicen ciertos pasos.)



Ilustración 21 Ventana principal

5. Para cerrar sesión solo necesita presionar el botón con la llave que se encuentra en la parte baja a la derecha con un icono de una llave. Si desea cerrar la sesión presionar donde dice aceptar de lo contrario presione cancelar.

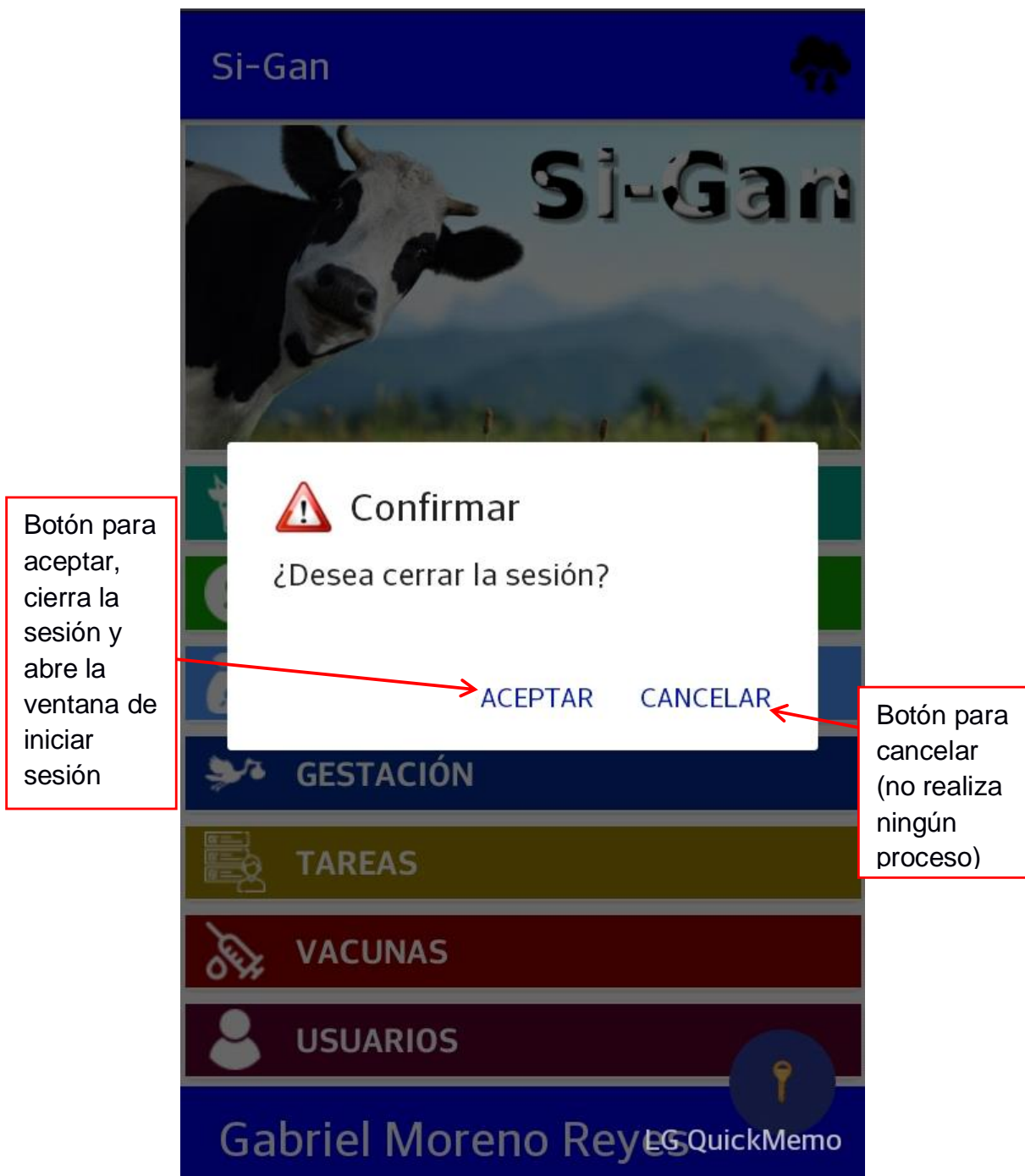


Ilustración 22 Cerrar sesión

CATÁLOGO DE ANIMALES



Ilustración 23 Botón animales

6. En el catálogo de animales se pueden encontrar todos los animales que sean agregados, la primer vez que se abra la aplicación con un nuevo cliente no aparecerán animales aun, para poder agregar un nuevo animal es necesario presionar el botón con el signo “+” que se encuentra en la parte baja a la derecha de su pantalla, ahora presione el botón que dice agregar con el icono de una vaca.

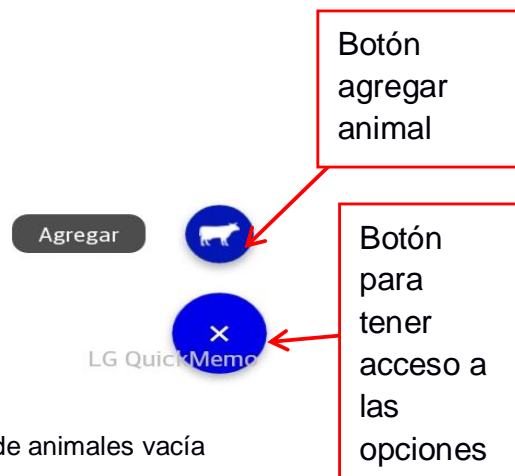


Ilustración 24 Lista de animales vacía

7. Una vez que se presione el Botón agregar se le mandara a una nueva ventana donde están los campos que se necesitan rellenar para poder agregar un nuevo animal (Conservación: los campos obligatorios se marcaran con "***" en la d)

The screenshot shows a mobile application interface for adding a new animal. The top section features a large image of a cow with the text "Si-Gan" overlaid. A green plus icon in the top right corner is labeled "Botón para guardar el nuevo". Below the image is a blue button labeled "Tomar foto", which is labeled "Botón para tomar una fotografía". To the right of the main form is a dropdown menu for "Familias" with options: Bovino, Caballar, Caprino, and Porcino. This is labeled "Familias a la que puede pertenecer el animal.". The form fields are as follows:

- Identificador**: A text input field with the placeholder "Ingrese el identificador". It is labeled "Identificador del animal, puede ser el número de la orejera o algo con lo que sea identificado del resto, puede llevar letras y números. ***".
- Nombre**: A text input field with the placeholder "Ingrese el nombre". It is labeled "Nombre del animal, puede ser el que guste puede almacenar letras y números. ***".
- Fecha de nacimiento**: A section with two options: a calendar icon and a checkbox labeled "Edad". The calendar icon is labeled "Si presiona el Botón con el calendario tendrá que seleccionar la fecha, si presiona edad tendrá que escribir en años la edad del animal. ***". Below this is a text input field with the placeholder "0000-00-00 00:00:00.0".
- Color**: A text input field with the placeholder "Ingrese el color". It is labeled "Color Del animal. ***".
- Raza**: A text input field with the placeholder "Ingrese la raza". It is labeled "Raza a la que pertenece el animal. ***".

Annotations also point to the image area, stating "Imagen será reemplazada si toma una fotografía".

Selección de padres:

- Seleccióné el padre** (Botón): Botón para seleccionar el padre/madre, en la ventana que aparecerá solo contendrá los machos/hembras mayores.
- Seleccióné la madre** (Botón): Una vez que se selecciona un padre/madre el identificador del aparecerá aquí.

Sexo: Macho (Sexo del animal Macho/Hembra)

Estatus: Vivo (Estatus del animal Vivo/Muerto)

Fecha de alta: 2017-08-14 12:55:18.0 (Fecha en que se ingresa el animal, no es editable es llenado automáticamente.)

Nota: Ingrese la nota (En caso de tener alguna nota sobre el animal (Opcional))

Costo: Ingrese el costo \$ (Precio del animal. ***)

Ilustración 25 Descripción nuevo animal

8. Una vez explicados los campos que son necesarios para agregar un nuevo animal se procederá a agregar uno, para ello se tomara una foto, presione el botón **“Tomar foto”** el cual lo mandara a la cámara de su teléfono, tome la foto de manera normal (para que se pueda visualizar mejor la foto tomarla de forma horizontal).

Un botón rectangular de color azul con el texto "Tomar foto" en blanco.

Ilustración 26 Botón tomar foto

9. Cuando haya tomado la foto le aparecerá una pantalla, en la cual se presionara **“OK”** para confirmar que si desea agregar esa foto, **“Cancelar”** para omitir la foto.



Ilustración 27 Tomar foto

10. En caso de presionar **“OK”** le aparecerá la fotografía tomada de esta manera como se aprecia en la siguiente imagen.



Ilustración 28 Fotografía del animal

11. En las siguientes ilustraciones se podrán apreciar imágenes de cómo quedaría una vez rellenado y seleccionado los campos correspondientes de manera correcta. Observe los campos madre y padre están vacíos recuerde que es el primer animal que se ingrese por lo tanto no hay forma que sus padres estén registrados. (Observación: los campos padre y madre pueden estar vacíos.)

Familia Bovino ▾	Fecha de nacimiento  <input type="checkbox"/> Edad 2014-08-14 12:58:40.0
Identificador 1999	Color Negro
Nombre Angus	Raza Aberdeen Angus
Selección de padre Selección de padre	Sexo Macho ▾
Selección de madre Selección de madre	Estatus Vivo ▾
Fecha de alta 2017-08-14 12:55:18.0	
Nota Aberdeen Color negro	
Costo 15000 LG QuickMemo	

Ilustración 29 Datos llenado del animal

12. Para guardar los datos que se tienen ingresados es necesario que presione el botón guardar que se encuentra en la parte superior derecha.



Botón para guardar el nuevo animal

Ilustración 30 Botón agregar

13. O en caso de no querer agregar el animal presione la tecla “**atrás**” de su teléfono o el botón “**atrás**” que se encuentra en la esquina izquierda superior.

Botón atrás, lo regresa a la lista de animales.



Ilustración 31 Salir

14. Repítase en todos los animales que desee guardar, cuando ya se tengan animales machos y hembras ya se podría agregar un madre y/o madre a el nuevo animal siempre y cuando sean de la misma familia y sea menor la cría. (Recuerde que el campo fecha de nacimiento debe estar previamente llenado así como tener elegida la familia correcta). Para agregar los padres la única diferencia con el ejemplo anterior es seleccionar un nuevo padre/madre.

Seleccione el padre

Seleccione la madre

Ilustración 32 Botones padres

15. Cuando se haya presionado le aparecerá la siguiente ventana donde seleccionara el padre/madre según el Botón presionado, en la siguiente ventana se puede hacer una búsqueda de animales ya sea por su identificador o el nombre para ello debe rellenar el campo de búsqueda y después presionar el Botón buscar, para elegir el animal que será el padre/madre seleccione una fila. (para cerrar la ventana de clic fuera de la ventana emergente)



Ilustración 33 Seleccionar padres

16. Cuando se seleccione el padre/madre se escribirá el identificador en el campo padre/madre como se aprecia en la siguiente imagen, guarde el nuevo animal en caso de ser correctos los padres y los datos.

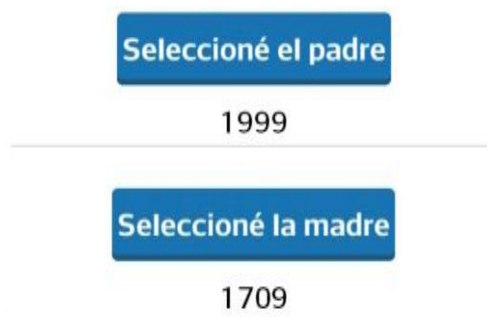


Ilustración 34 Padres del animal

17. Una vez que se agregue el anterior animal, presione la tecla atrás como lo dice el punto “13”. Lo mandara a la lista de animales, como se aprecia se pueden ver 3 animales agregados, los padres y la cría.

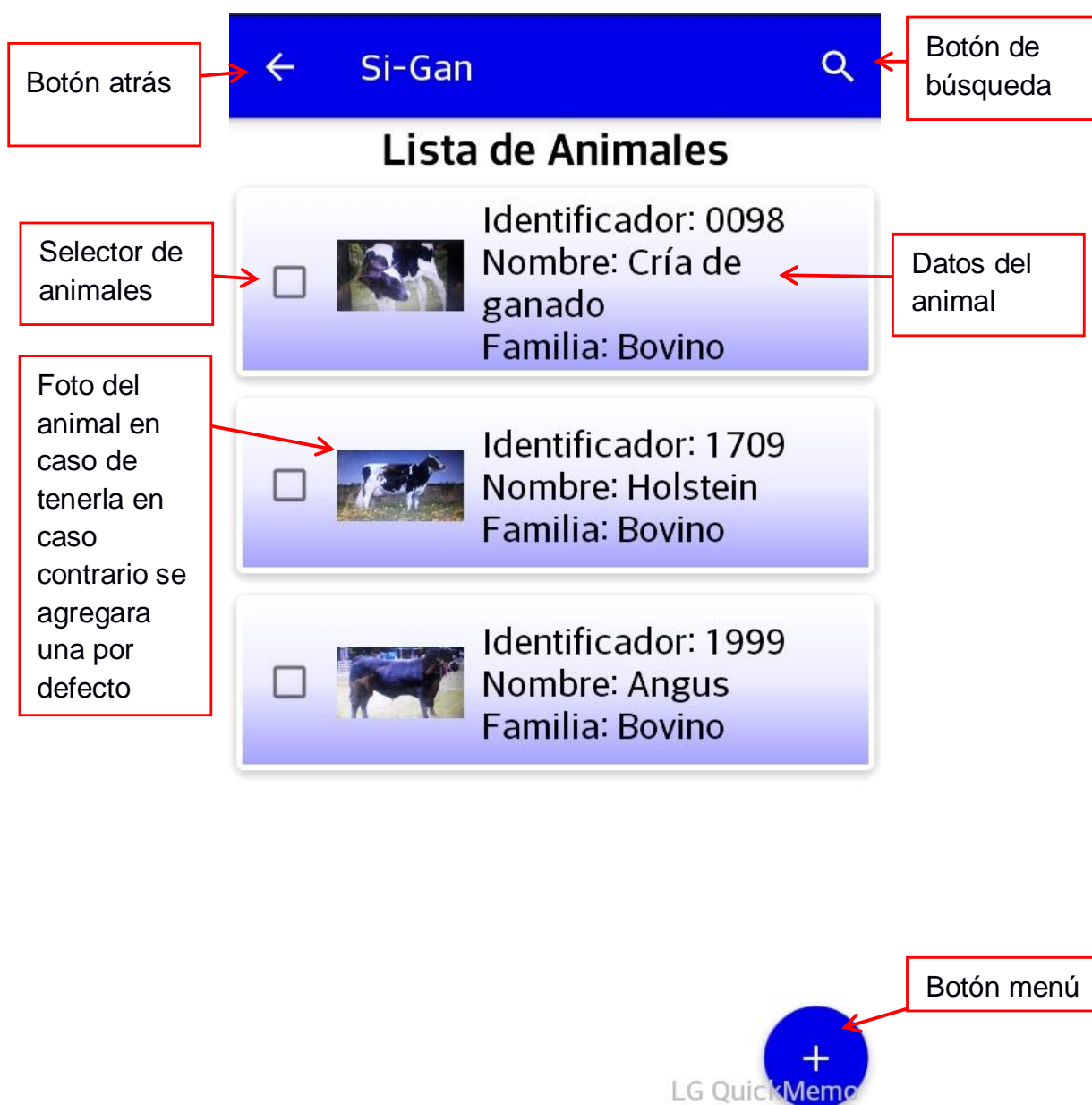


Ilustración 35 Lista de animales llena

18. Para seleccionar varios animales a la vez, se debe presionar una fila por unos segundos hasta que el botón de selección este con una palomita o bien de clic sobre el botón de selección.



Ilustración 36 Opciones del menú de animales

19. Si desea actualizar un animal, presione la pantalla sobre una fila con un toque corto, le mandara la siguiente pantalla donde se podrán editar los campos que desee. Una vez que haya editado los campos correspondientes diríjase hacia la parte de arriba donde se pueden observar botones para eliminar y actualizar. En caso de no querer modificarlo solo presione atrás como se le explica en el paso “13”.



Ilustración 37 Datos del animal (actualizar/eliminar)

CATÁLOGO UBICACIÓN



Ilustración 38 Botón ubicación

20. El catálogo de ubicación se encuentran las ubicaciones agregadas, cuando se agrega un nuevo animal este es agregado a la ubicación **“Sin ubicación”**, cuando se inicia por primera vez solo estará la ubicación antes mencionada, también se encuentra el apartado para vender animales.



Ilustración 39 catálogo de ubicación

21. Para mover/vender un animal se debe seleccionar una ubicación, presionando sobre una fila, al tener 1 o más animales aparecerán las opciones en la parte baja de la pantalla.



Ilustración 40 Opciones de ubicación

22. presione el botón mover para mover a una ubicación el animal, una vez que se abra la ventana podrá seleccionar una ubicación o agregar una nueva, en este caso como aún no hay más ubicaciones se agregara una nueva. Para ello de clic en el botón **"Nueva"**.

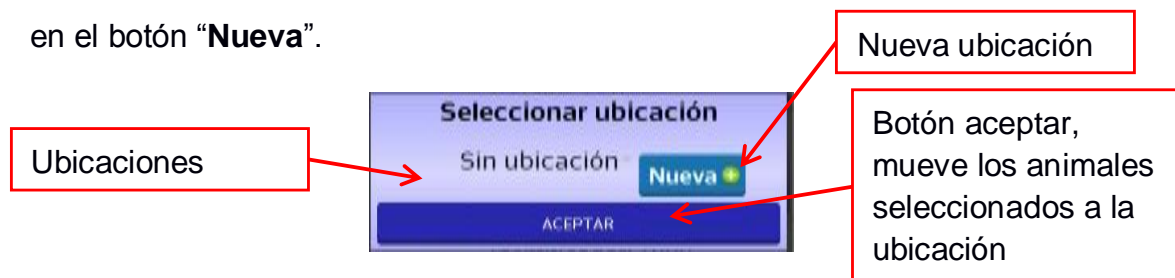


Ilustración 41 Seleccionar ubicación

23. Una vez que se dé clic en nuevo le aparecerá una caja de texto donde introducirá la nombre de la ubicación, en el ejemplo se puso “Corral 1” si no quiere agregar la ubicación presione sobre cancelar. El botón que se encuentra debajo que dice “Normal” tiene 2 estados el primero sería normal y el segundo Estabulación (la estabulación consiste en mantener a los animales que se crían dentro de un establecimiento, es decir, un lugar donde estén estos animales durante gran parte de su vida) el primer caso quedará en normal, en caso de estar seguro/a que sea así la ubicación seleccione el botón aceptar para agregar los animales seleccionados en la nueva ubicación.

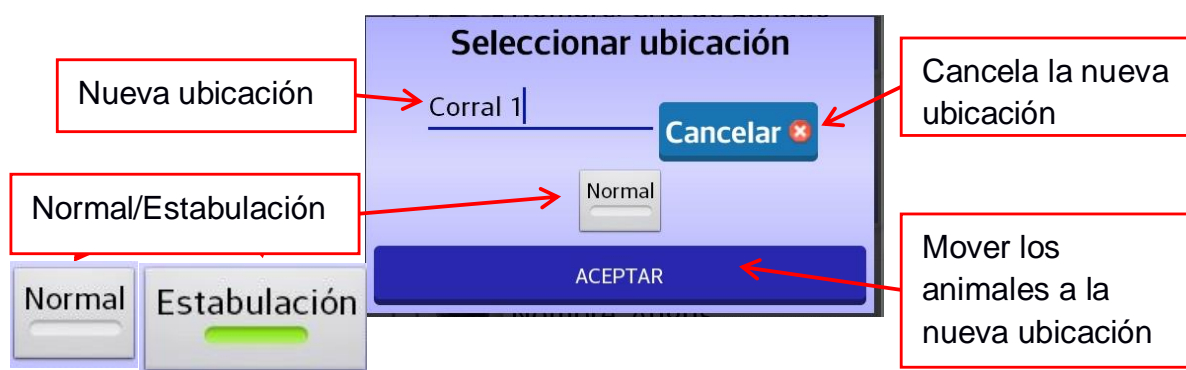


Ilustración 42 Nueva ubicación normal

24. Como se puede mostrar en la siguiente imagen el animal se movió a la nueva ubicación.



Ilustración 43 Nueva ubicación agregada

25. Si presiona sobre el ítem “**Corral 1**” le aparecerán todas las ubicaciones que hay hasta el momento

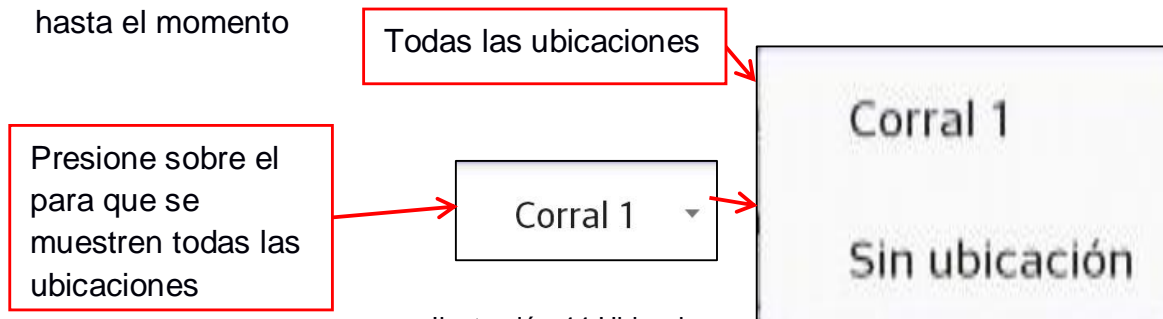


Ilustración 44 Ubicaciones

26. Seleccione la ubicación “**Sin ubicación**” para ver los animales que están en esa ubicación, donde se seleccionara un animal y damos clic en el botón “**Mover**”.

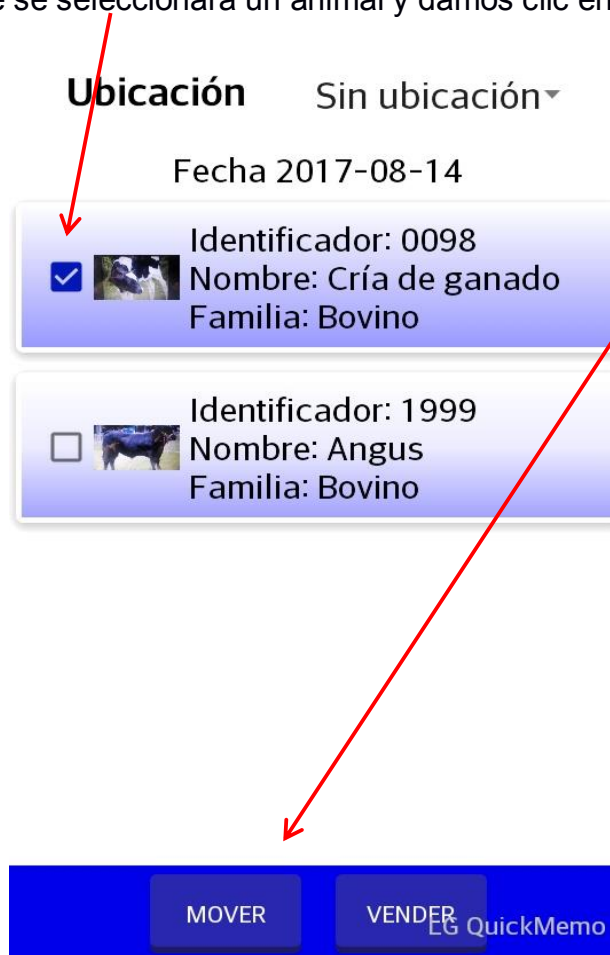


Ilustración 45 Mover estabulación

27. Le aparecerá nuevamente la ventana donde dará clic en el botón nuevo y se escribirá la nueva ubicación y seleccionaremos el botón **“Normal”** para que se ponga en estado **“Estabulación”**, presione aceptar.



Ilustración 46 Nueva ubicación estabulación

28. Al dar clic sobre aceptar se le mandara a la siguiente ventana donde se tendrá que rellenar el campo peso, (Observaciones. Solo se moverá si cumple con dos condiciones: 1ra.- que tenga el peso ingresado. 2da que esta seleccionada la fila. Las filas que no cumplan con las dos condiciones serán omitidas). Cuando de rellenen los campos peso presione sobre **“ACEPTAR”** para confirmar que si se desea mover o **“CANCELAR”** esta opción cancela el mover los animales.



Ilustración 47 Mover a estabulación

29. Seleccione la ubicación “**Estabulación**” para poder ver los animales que están en esa ubicación.

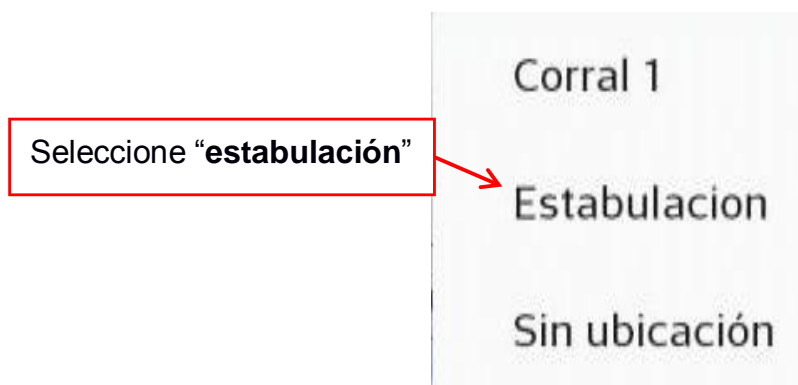


Ilustración 48 Ubicaciones

30. Al seleccionar una ubicación se mostraran todos los animales que están en ella.



Ilustración 49 Animales por ubicación

31. Para poder vender animales es necesario que se seleccione la ubicación donde se encuentran los animales a vender, seleccione todos los animales que desea vender para luego dar clic en el botón “**Vender**”.

Ubicación Sin ubicación ▾

Fecha 2017-08-14

<input type="checkbox"/>		Identificador: 1999 Nombre: Angus Familia: Bovino
<input checked="" type="checkbox"/>		Identificador: 8009 Nombre: Animal #5 Familia: Bovino

MOVER VENDER LG QuickMemo

Ilustración 50 Iniciar venta

32. Una vez que presiono el botón vender le aparecerá la siguiente pantalla donde le pedirá el precio en que será vendido. (Observaciones. Solo se moverá si cumple con dos condiciones: 1ra.- que tenga el precio de venta. 2da que esta seleccionada la fila. Las filas que no cumplan con las dos condiciones serán omitidas).

Ubicación

Fecha 2017-08-14

<input checked="" type="checkbox"/>		Identificador: 8009 Nombre: Animal #5 Familia: Bovino Ingrese el precio de venta 7800
-------------------------------------	---	---

ACEPTAR CANCELAR LG QuickMemo

Ilustración 51 Rellenar precio de venta

CATÁLOGO ALIMENTACIÓN



Ilustración 52 botón alimentación

33. En el catálogo de alimentación solo se toman en cuenta las ubicaciones que están en estabulación. En este caso como se vio anteriormente agregamos una ubicación llamada “**Estabulacion**” que es la única que aparece por el momento en la cual se pueden ver algunos datos como: Nombre de la ubicación, la suma de la cantidad de los alimentos, la suma del costo total de los alimentos, el total de alimentos que se dio en la ubicación y el total de animales. Pero en primera instancia la mayoría aparecerá en cero solo en animales que esta el que se agregó en esa ubicación. Como se puede ver este reporte está en un rango de fechas las cuales se pueden cambiar a su gusto dando clic sobre el botón con el icono de un calendario.

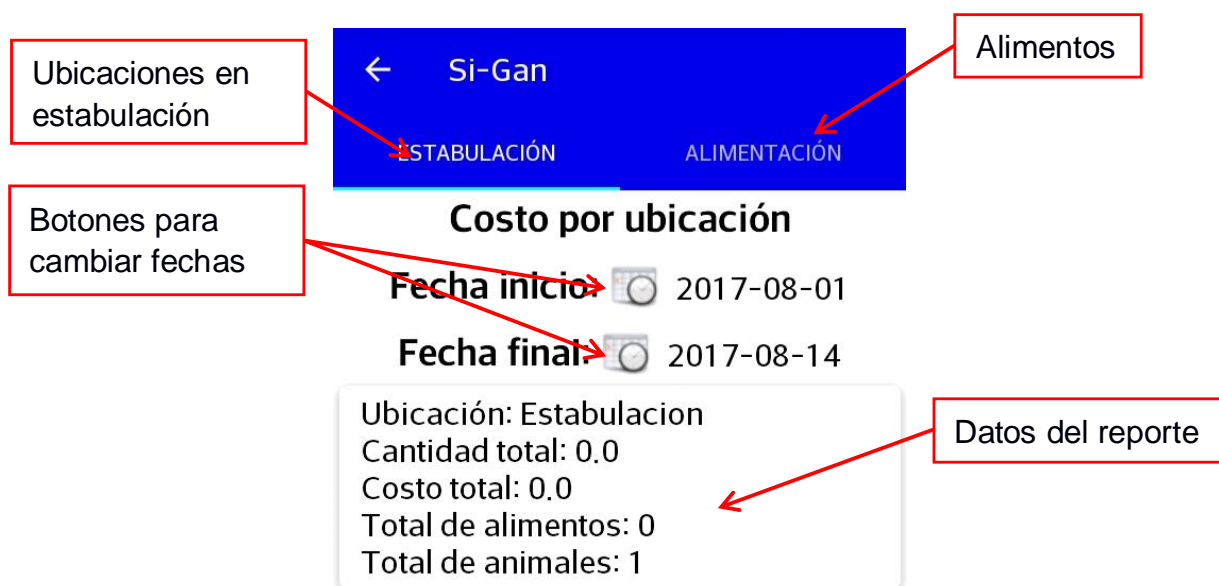


Ilustración 53 Reporte de alimentación

34. Si en estos momentos se dirige al apartado alimentación le dará el siguiente mensaje.



Ilustración 54 Apartado de alimentación con error

35. Para que se muestren los alimentos o agregar un nuevo alimento es necesarios que seleccione una fila, al seccionarla automáticamente lo mandara a el apartado alimentación.

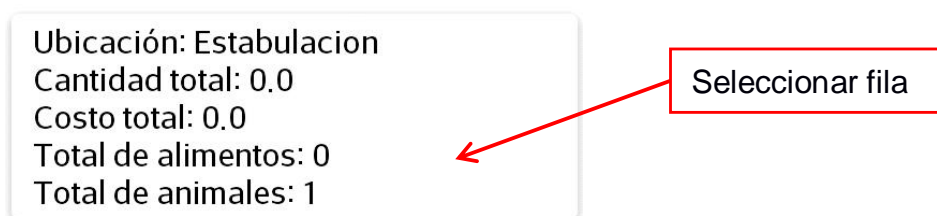


Ilustración 55 Selección de fila

36. Después de haber seccionado la fila le aparecerá la ubicación que seleccionó y en la parte baja a la derecha estará un botón al presionarlo le saldrá la opción agregar que es para agregar un nuevo alimento.

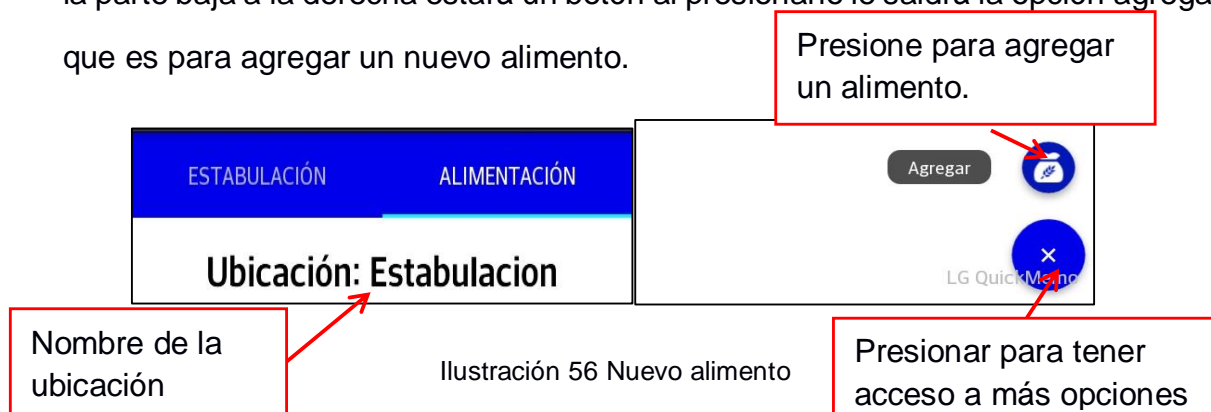


Ilustración 56 Nuevo alimento

37. Al presionar el Botón agregar le abrirá la ventana que se mira a continuación, debe rellenar los datos ningún campo debe quedar vacío. Una vez rellenado los datos puede presionar el botón **“Guardar”** para guardar el alimento o presionar el botón **“Cancelar”** para cerrar la ventana (no se guardara nada).

The screenshot shows a form titled 'Alimento' with the following fields and annotations:

- Nombre del alimento:** Points to the 'Caña' text field.
- Cantidad del alimento solo acepta números:** Points to the '150' text field.
- Precio total del alimento:** Points to the '5000' text field.
- Fecha:** Points to the date/time field showing '2017-08-14 15:03:09.0'.
- Botón para cambiar la fecha en que se dio:** Points to the calendar icon next to the date field.
- Guarda los datos del alimento:** Points to the 'Guardar' button.
- Cierra la ventana:** Points to the 'Cancelar' button.

Ilustración 57 Datos del alimento para agregar

38. Cuando se guarda el alimento se agregara a la lista de alimentos en la ubicación.

The screenshot shows a list titled 'Ubicación: Estabulacion' with a tab 'ALIMENTACIÓN' selected. A new food item is added to the list:

- Nuevo alimento:** Points to the 'Alimento: Caña' text.
- Alimento:** Caña
- Precio:** 5000,0
- Fecha:** 2017-08-14 15:03:09.0

Ilustración 58 Nuevo alimento

39. Regresando al apartado de Estabulación se podrá ver como se actualizaron los datos.

ESTABULACIÓN ALIMENTACIÓN

Costo por ubicación

Fecha inicio: 2017-08-01

Fecha final: 2017-08-14

Ubicación: Estabulacion
Cantidad total: 150.0
Costo total: 5000.0
Total de alimentos: 1
Total de animales: 1

Campos actualizados

Ilustración 59 Datos actualizados

40. Si se cambian las fechas se podrá ver que son nuevamente actualizados los datos ya que la fecha no se encuentra en el rango.

ESTABULACIÓN ALIMENTACIÓN

Costo por ubicación

Fecha inicio: 2017-08-15

Fecha final: 2017-08-21

Ubicación: Estabulacion
Cantidad total: 0.0
Costo total: 0.0
Total de alimentos: 0
Total de animales: 1

Campos actualizados

Ilustración 60 Cambio de fecha

CATÁLOGO GESTACIÓN



Ilustración 61 Botón gestación

41. El apartado gestación se podrán ver en primer momento todos los animales que tengan como sexo **"Hembra"**. Este apartado es para llevar un control de la gestación de las hembras. Hay 3 tipos de estado, **"No preñada"**, **"Estimulada"** cuando se estimula la hembra para que se preñe y **"Preñada"** cuando la hembra se encuentre preñada. Para que se identifiquen de mejor manera se dividió por familias, en el siguiente ejemplo muestra las hembras de la familia bovino que no están preñadas. (Observación: solo aparecerán las familias que están en uso.)

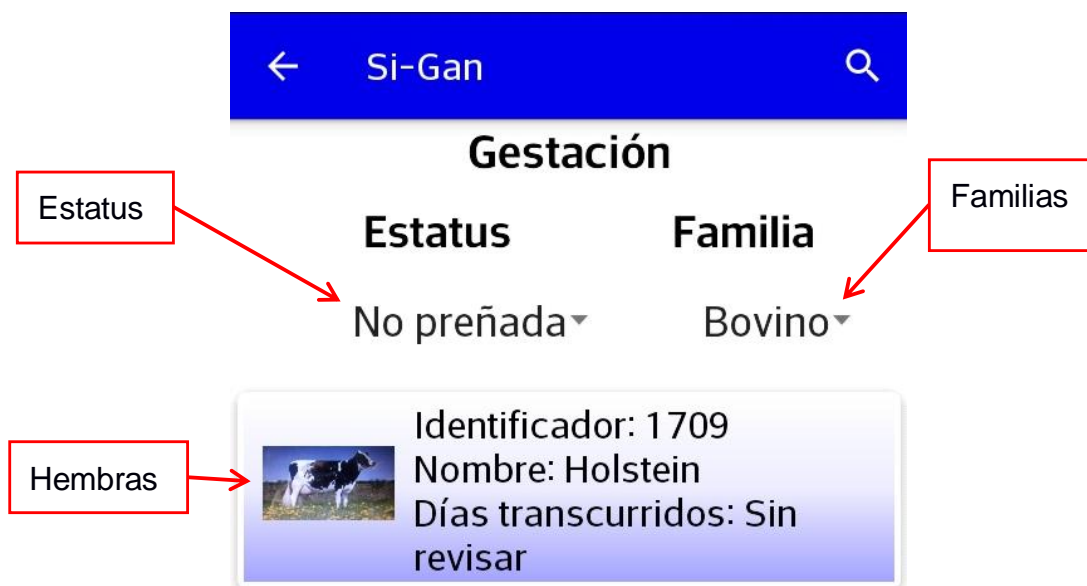


Ilustración 62 Gestación

42. Para cambiar hacer una revisión de la hembra presione sobre una fila, lo cual ara que aparezcan 2 opciones la primera será la de “**REVISAR**” en el cual se podrá cambiar el estatus del animal.

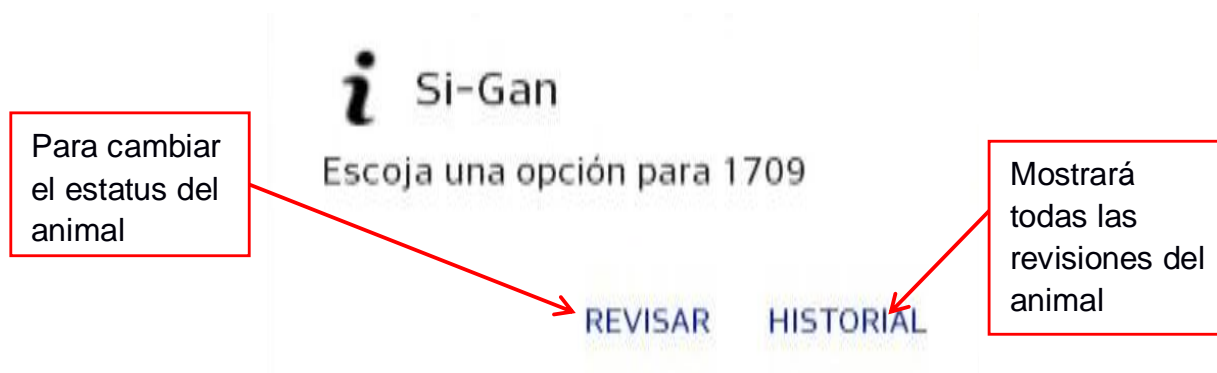


Ilustración 63 Opciones del animal para la gestación

43. Cuando haya presionado sobre el botón “**REVISAR**” le aparecerá a siguiente ventana emergente donde se podrá cambiar el estatus que se tiene, presione sobre el estatus actual “**No preñada**” donde le aparecerán los siguientes estatus. Para guardar le revisión presione sobre el botón “**Aceptar**”.

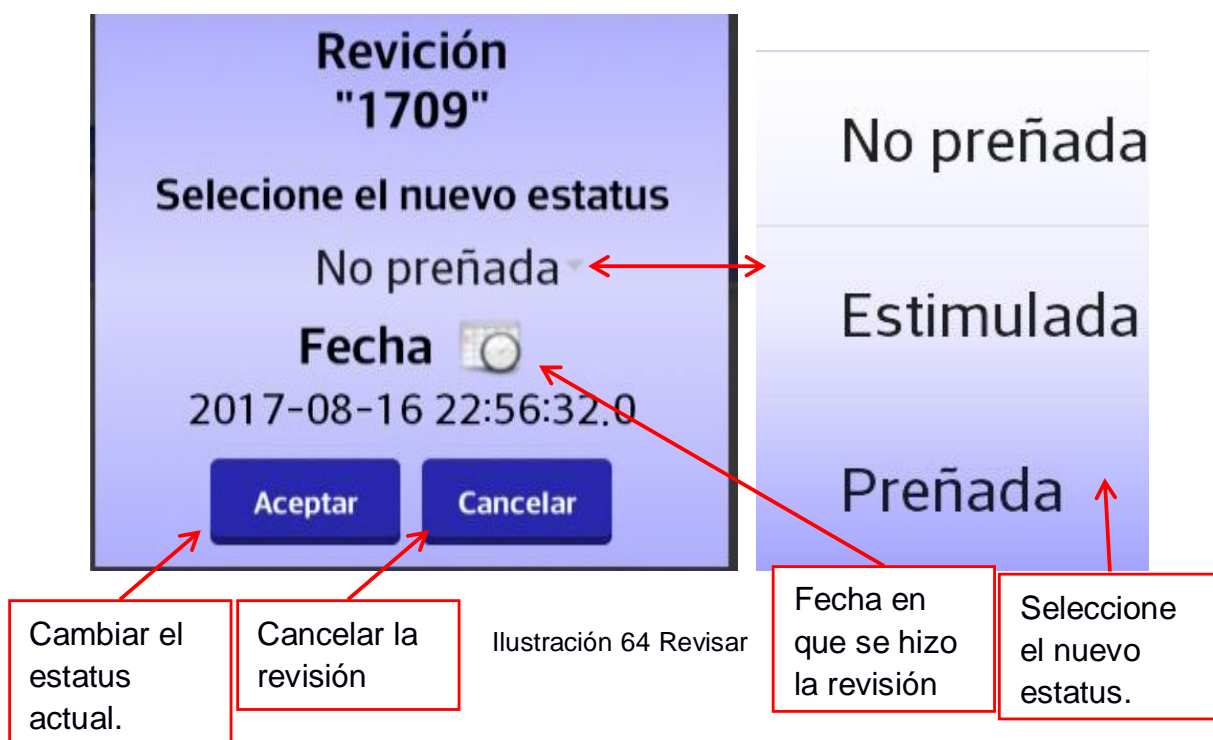


Ilustración 64 Revisar

44. En el siguiente ejemplo seleccionaremos el estatus estimulada y se guardaremos el estatus.

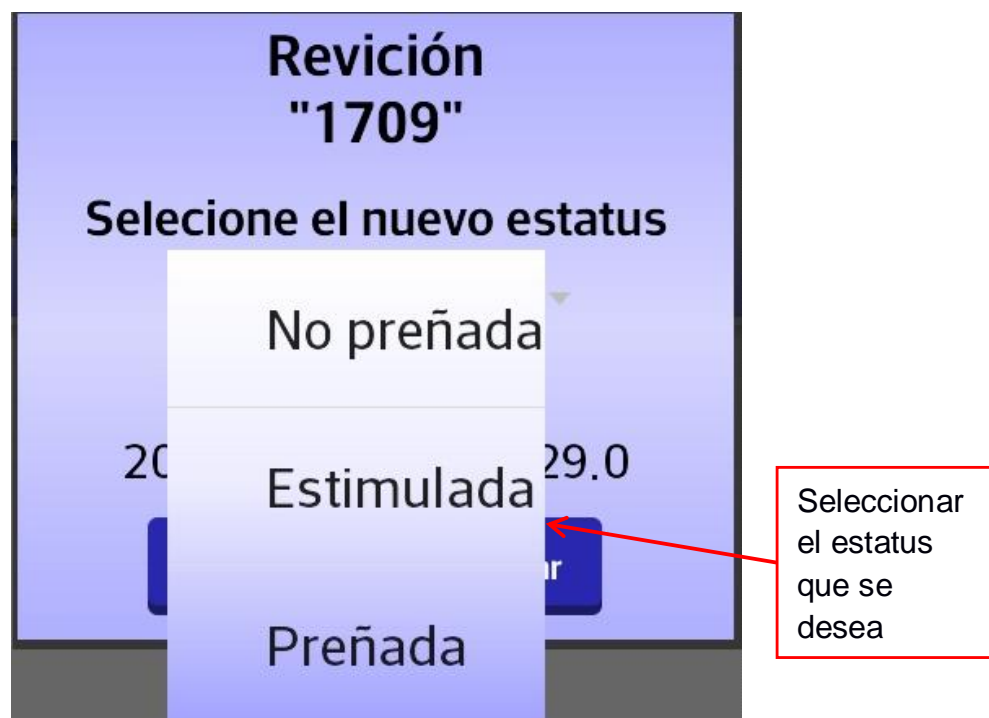


Ilustración 65 Estatus estimulada

45. Una vez guardada, se puede ver que se encuentra en el estatus "**Estimulada**".

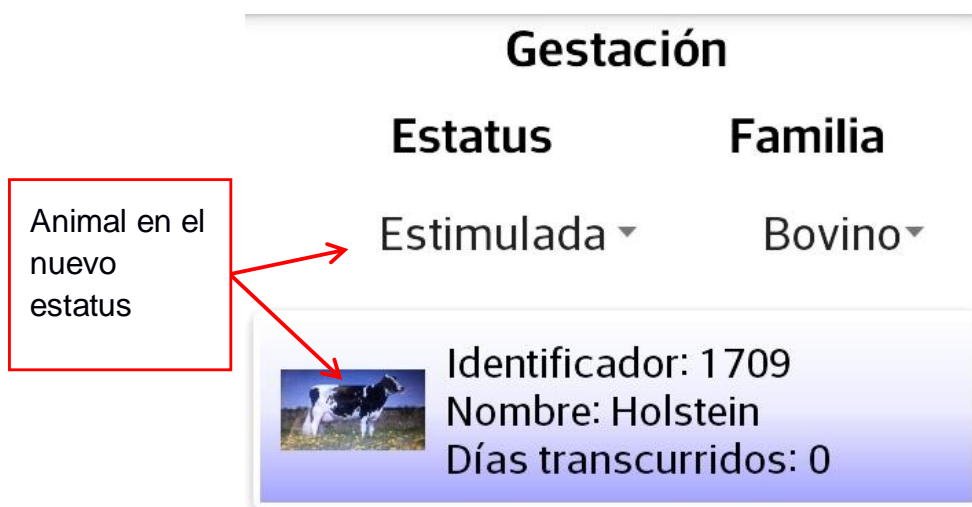


Ilustración 66 Animal en el estatus Estimulada

46. Ahora se cambiaremos el estatus a **“Preñada”** donde aparecerá un nuevo campo en el cual se debe seleccionar el padre de la cría en caso que el padre este registrado de lo contrario puede estar vacío. Para seleccionar el padre presione el botón **“Seleccioné el padre”** y elija la fila del padre. Presione el botón **“Aceptar”** para agregar la revisión.

Revisión "1709"
Seleccione el nuevo estatus
Preñada
Fecha 2017-08-14 15:30:57.0
Seleccioné el padre
Seleccione el padre
Aceptar Cancelar

Buscar el Padre
Buscar
Identificador: 0098
Nombre: Cría de ganado
Familia: Bovino
Identificador: 1999
Nombre: Angus
Familia: Bovino

Revisión "1709"
Seleccione el nuevo estatus
Preñada
Fecha 2017-08-14 15:33:48.0
Seleccioné el padre
1999
Aceptar Cancelar

Ilustración 67 Pasos para seleccionar el padre

47. Cuando se guarde se podrá ver que se encuentra en el estatus “**Preñada**”.

Gestación

Estatus	Familia
Preñada ▼	Bovino ▼



Identificador: 1709

Nombre: Holstein

Días transcurridos: 0

Padre: 1999


Ilustración 68 Estatus preñada

48. Para la revisión de una hembra cuando se encuentra en el estatus “**Preñada**” se puede elegir un nuevo estatus que sería el de “**Aborto**” este es para cuando no se concluye la gestación satisfactoriamente y el producto es abortado, tomaremos esto como ejemplo. Seleccione el estatus “**Aborto**” y guarde.

Revisión "1709"

Seleccione el nuevo estatus

Aborto ▼

Fecha 

2017-08-14 15:42:01.0

Aceptar

Cancelar

Ilustración 69 Estatus aborto

49. Como se mira en la “Ilustración 63 Opciones del animal para la gestación” seleccionar la opción historial donde le mandara todas las relaciones que se hayan hecho, como se puede ver se realizan 2 revisiones automáticamente cuando se selecciona el estatus “**Aborto**” en primer lugar se guarda el estatus “**Aborto**” y después al estatus “**No preñada**”.



Ilustración 70 Historial de gestación

50. Ahora supongamos que la gestación fue correcta y el animal procedió con el parto y se cambiara el estatus a cualquier otro estatus a excepción del de “**Aborto**”, le saldrá un mensaje por si desea agregar la cría como un nuevo animal.

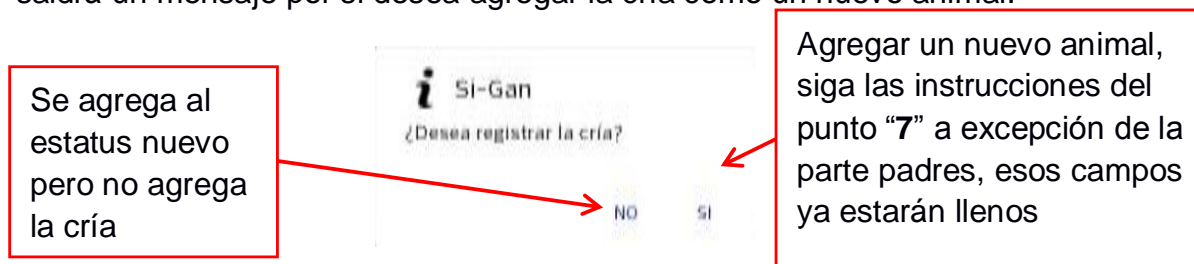


Ilustración 71 Registro de la cría

CATÁLOGO TAREA



Ilustración 72 Botón tarea

51. En el catálogo de tareas se encuentran todas aquellas tareas que se deben realizar con los animales.

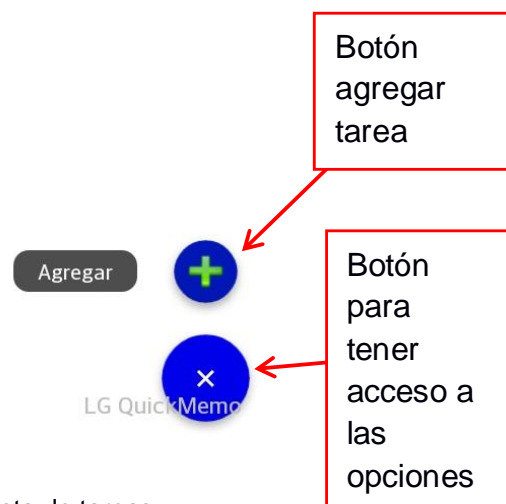


Ilustración 73 lista de tareas

52. En la pestaña “**HACER TAREA**” se pueden encontrar a todos los animales filtrados por ubicación.

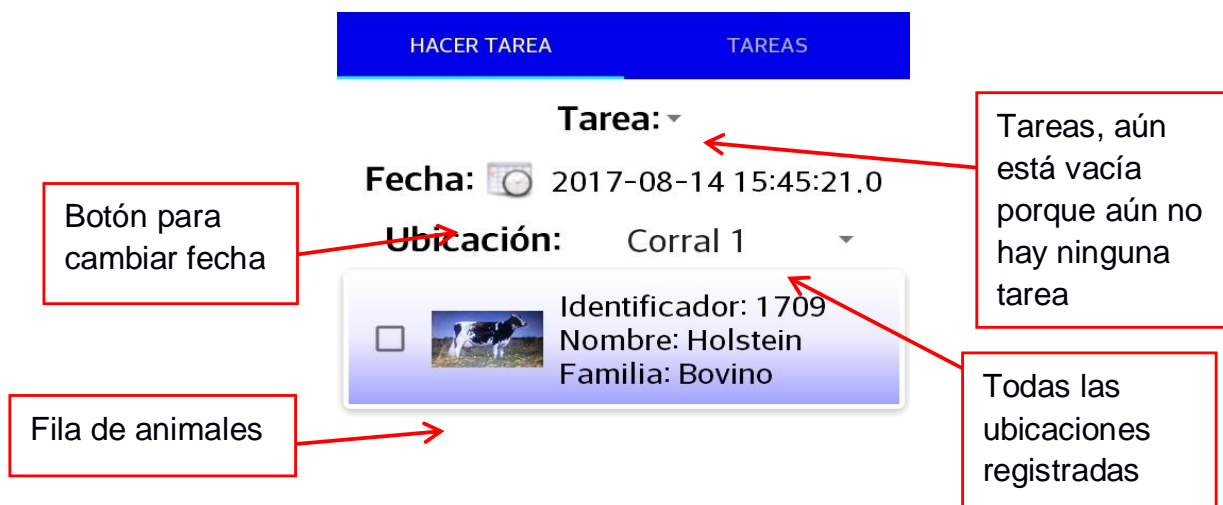


Ilustración 74 Hacer tarea

53. Para agregar una tarea diríjase a la pestaña “**TAREAS**”. Estando allí presione el botón “**Agregar**” para poder agregar una nueva tarea.

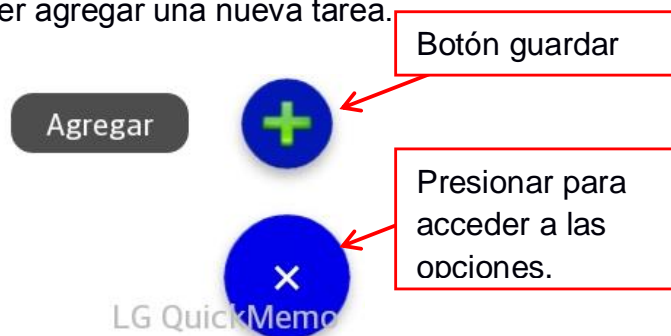


Ilustración 75 Botón agregar tarea

54. Para agregar una tarea se debe rellenar los campos como el siguiente ejemplo.

Una vez que estén rellenos los campos deberá presionar el botón “**Guardar**” situado en la parte superior derecha.

The screenshot shows a mobile application interface for adding a task. At the top is a blue header bar with a back arrow, the text 'Si-Gan', and a green plus icon. Below the header is the title 'Agregar Tarea'. The form consists of several fields: 'Tarea' with the value 'Bañar', '¿Cuantificable?' with a 'No' button, 'Regularidad' with the value '5', and 'Estatus' with a dropdown menu showing 'Activo'. Red arrows point from text boxes to these fields: 'Nombre de la tarea' points to 'Tarea'; 'Regularidad, cada cuanto tiempo se realiza la tarea en semanas' points to 'Regularidad'; 'Para saber si la tarea es cuantificable. Para saber si se puede contar como por ejemplo: el bañar no se puede cuantificar de lo contrario se cuantifica cuanto peso' points to '¿Cuantificable?'; and 'Estatus de la tarea Activo/Inactivo' points to the 'Estatus' dropdown.

← Si-Gan +

Agregar Tarea

Tarea
Bañar

¿Cuantificable?
No

Regularidad
5

Estatus
Activo ▾

Nombre de la tarea

Regularidad, cada cuanto tiempo se realiza la tarea en semanas

Para saber si la tarea es cuantificable. Para saber si se puede contar como por ejemplo: el bañar no se puede cuantificar de lo contrario se cuantifica cuanto peso

Estatus de la tarea Activo/Inactivo

Ilustración 76 Agregar tarea

55. Regresamos y nos dirigimos a la pestaña “**TAREAS**” donde se pueden ver las tareas que agregamos. Para acceder a todos los datos o se necesita editar/eliminar la tarea. Para ello presione sobre una fila.

The screenshot shows a mobile application interface with two tabs: 'HACER TAREA' and 'TAREAS'. The 'TAREAS' tab is selected. Below the tabs is the title 'Lista de Tareas'. There is a list item with a checkbox, the text 'Tarea: Bañar', and 'Regularidad: Cada 5 semanas'. A red arrow points from a text box to this list item.

HACER TAREA TAREAS

Lista de Tareas

☐ Tarea: Bañar
Regularidad: Cada 5 semanas

Presionar sobre la fila.

Ilustración 77 Presionar fila

56. Datos de la tarea correspondiente, se puede modificar todos los datos de la tarea. Para modificar presionar el botón “**Actualizar**” y para el eliminar presione el botón “**Eliminar**”

Ilustración 78 Modificar tarea

57. Para eliminar una o más tareas desde la lista de tareas estas deben estar seleccionadas y presione el botón eliminar.

Ilustración 79 Eliminar varias tareas

58. Para poder registrar una tarea realizada diríjase a la pestaña “**HACER TAREA**” donde seleccionara a los animales con los que se realizara la tarea para luego presionar el botón guardar.

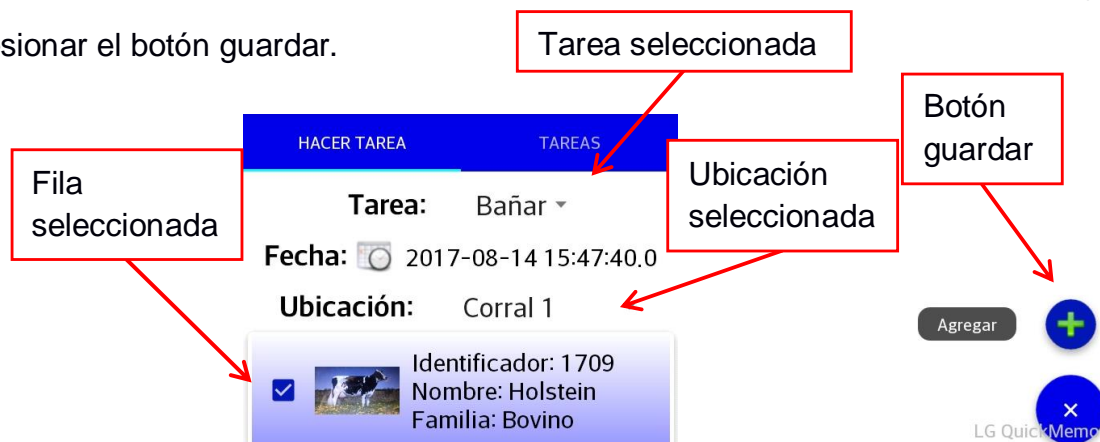


Ilustración 80 Guardar tarea realizada

59. Para registrar una tarea echa pero la tarea está registrada que si es cuantificable se necesitara realizar un paso extra, supongamos que en el ejemplo anterior se eligió la tarea pesar y se presionó el botón guardar, ahora como pesar si es cuantificable le pedirá la cantidad (Observación: solo se guardara la tarea echa cuando este seleccionada la fila además de tener una cantidad de lo contrario se omitirá).

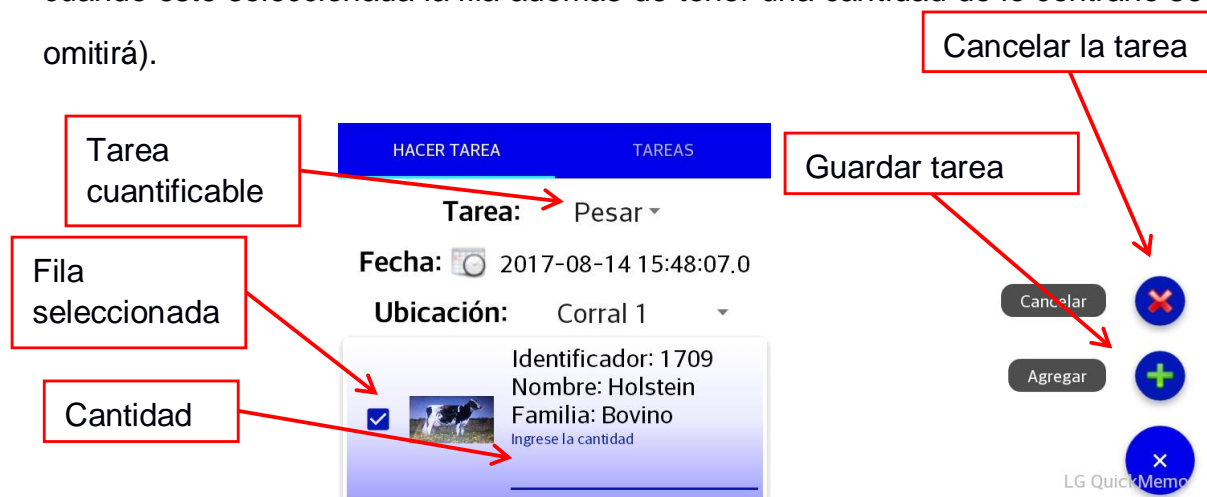


Ilustración 81 Guardar tarea cuantificable

60. Una vez que se realizaron las tareas diríjase a la pestaña “**TAREAS**” y presione una sobre una fila en la cual le aparecerán los datos de la tarea pero ahora veremos las tareas realizadas de esta misma. Para eso presione sobre el botón con tres puntos, donde le saldrá la opción “**Tareas hechas**” presione sobre él.

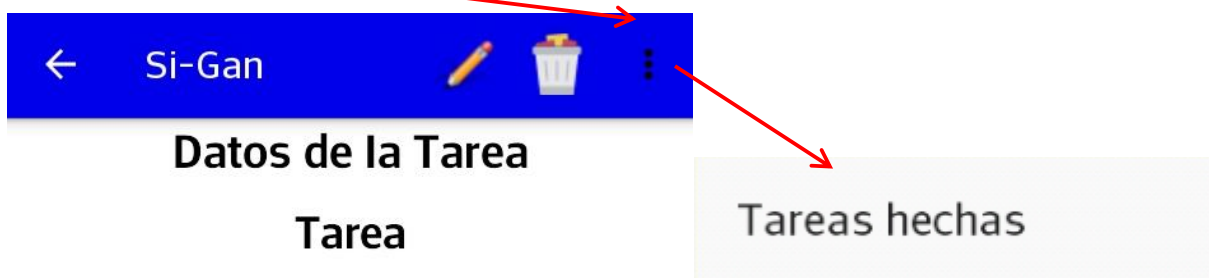


Ilustración 82 Menú tareas hechas

61. En el listado de las tareas realizadas le aparecerán una lista de tareas hechas según sea cuantificable o no.



Ilustración 83 Lista de tareas hechas

CATÁLOGO VACUNAS



Ilustración 84 Botón vacunas

62. En el catálogo de vacunas se podrá guardar las vacunas correspondientes, para llevar un control de las vacunas para cada animal.

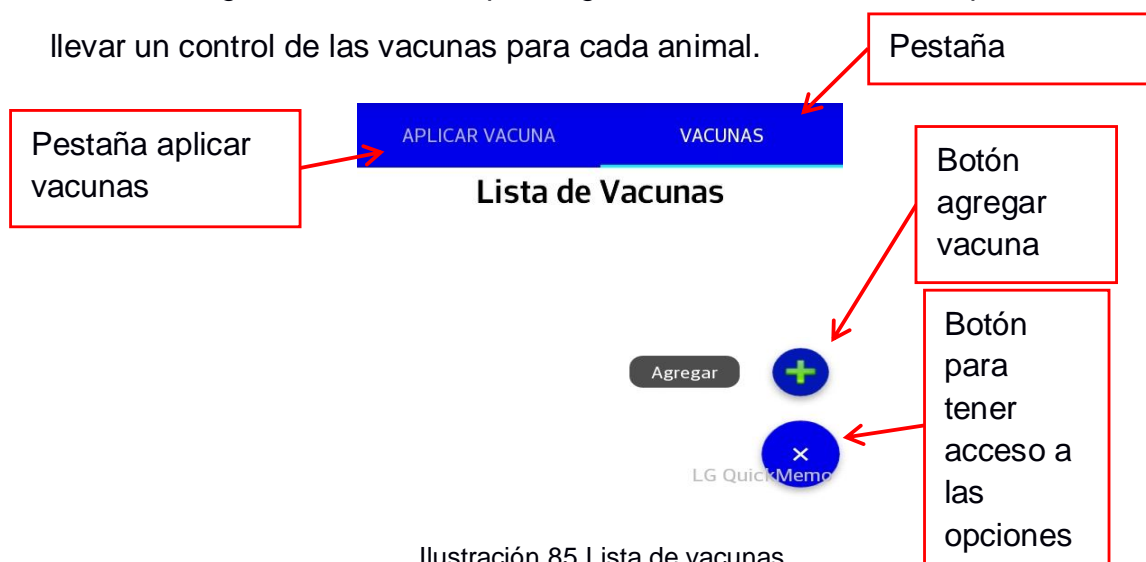


Ilustración 85 Lista de vacunas

63. En la pestaña “**Aplicar Vacuna**” se puede apreciar que aún no hay vacunas agregadas por ello aún no se pueden aplicar vacunas. Primero deberá agregar una vacuna o más para poder aplicarlas, a continuación se procederá a agregar una vacuna para ello sitúese en la pestaña “**VACUNAS**” donde deberá presionar el botón “**Agregar**” como se mira en la “Ilustración 85 Lista de vacunas”.



Ilustración 86 Sin Vacunas

64. Cuando se haya presionado el botón agregar en la pestaña de vacunas lo mandará a la siguiente ventana donde se tendrán que rellenar todos los datos que se piden. En la siguiente imagen se puede apreciar un ejemplo de cómo quedaría, si se desean guardar los cambios presione el botón guardar.

The screenshot shows a mobile application interface for adding a vaccine. At the top is a blue header bar with a back arrow, the text 'Si-Gan', and a green plus icon. Below this is the title 'Agregar Vacuna' and a sub-label 'Vacuna'. The form contains several fields: 'Multibacterina 7' (with an annotation 'Nombre de la vacuna'), 'Familia' with a dropdown menu showing 'Bovino' (with an annotation 'Familia de animales a la que se aplicara'), 'Regularidad' with a value of '30' (with an annotation 'Regularidad en días'), and 'Estatus' with a dropdown menu showing 'Activo' (with an annotation 'Estatus de la vacuna Activo/Inactivo'). At the top right, a red box points to the green plus icon with the label 'Botón guardar'. The caption below the form is 'Ilustración 87 Guardar Vacuna'.

65. Si se diera el caso que desee eliminar una o más tareas solo seleccione todas las filas que desee eliminar y le parecerá una opción más con la cual podrá eliminar las vacunas seleccionadas. Para eliminarlas presione el botón “**Eliminar**”

The screenshot shows a mobile application interface for a list of vaccines. At the top is a blue header bar with two tabs: 'APLICAR VACUNA' and 'VACUNAS'. Below this is the title 'Lista de Vacunas'. The list contains two items, each with a checkbox and text: 'Vacuna: Multibacterina 7', 'Familia: Bovino', 'Regularidad: Cada 30 días' and 'Vacuna: Nasalgent ip', 'Familia: Bovino', 'Regularidad: Cada 90 días'. To the right of the list are three circular buttons: 'Eliminar' (with a trash icon), 'Agregar' (with a plus icon), and a button with an 'x' icon. A red arrow points from the text 'Eliminar' in the instruction above to the 'Eliminar' button. The caption below the form is 'Ilustración 88 Eliminar vacuna'.

66. Para aplicar una vacuna diríjase a la pestaña “**APLICAR VACUNA**”, rellene los datos que se le pide y seleccione los animales a los que se le desea aplicar la vacuna con los datos ingresados. (Observación: solo aparecerán los animales que sean de la misma familia que la vacuna.)

The screenshot shows a mobile application interface for applying vaccines. The top navigation bar has two tabs: "APLICAR VACUNA" (selected) and "VACUNAS". The form contains the following fields and annotations:

- Vacuna:** Multibacterina 7 (Annotation: Vacuna)
- Fecha:** 2017-08-14 16:02:24.0 (Annotation: Fecha de aplicación, pointing to the date picker icon)
- Costo unitario:** 50 (Annotation: Botón para cambiar fecha, pointing to the date picker icon; also Costo por la aplicación de la vacuna a cada animal)
- Aplico:** Gabriel Moreno Reyes (Annotation: Nombre de quien aplico la vacuna)
- Ubicación:** Corral 1 (Annotation: Animales clasificados por ubicación, pointing to the location dropdown)
- Animals List:** A list of animals with a checked box for the first entry. The entry details are: Identificador: 1709, Nombre: Holstein, Familia: Bovino. (Annotation: Fila seleccionada, pointing to the checked box)
- Buttons:** "Agregar" (Annotation: Botón agregar, se guardan las vacunas realizadas) and "LG QuickMemo".

Ilustración 89 Aplicar vacuna

67. Para actualizar o eliminar una vacuna diríjase a la pestaña “**VACUNAS**” y presione sobre una fila.

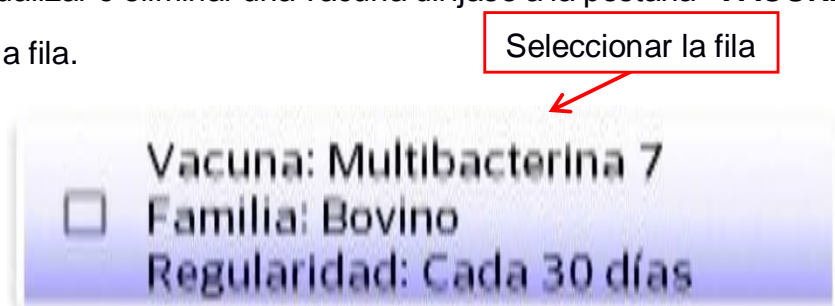


Ilustración 90 Fila de vacuna

68. Si desea actualizar los datos es necesario que sobrescriba los datos actuales y presione el botón “**Actualizar**” o si desea eliminar la vacuna presione el botón “**Eliminar**”. Como vimos anteriormente se guardaron unas vacunas aplicadas para ver estas es necesario que presione sobre el botón con 3 puntos donde saldrá el mensaje “**Vacunas aplicadas**” presione sobre él.

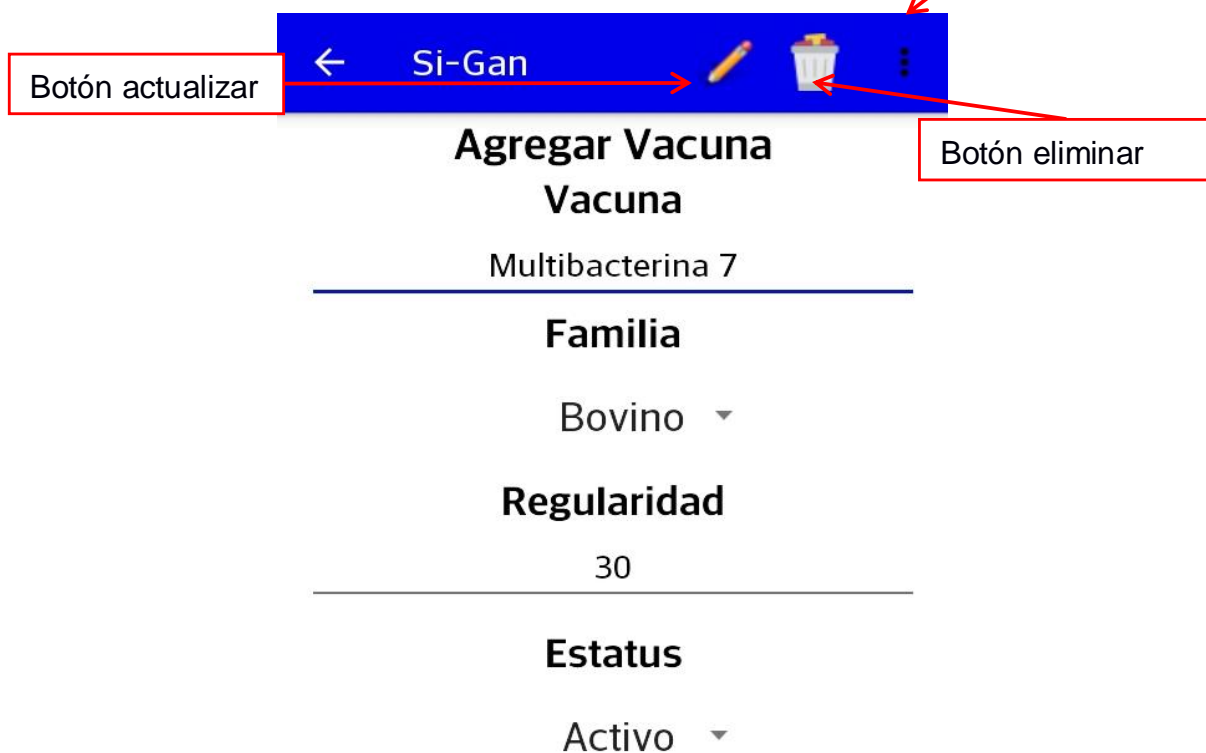


Ilustración 91 Datos de la vacuna

69. En el siguiente listado aparecerán todas las vacunas "**Multibacterine 7**" aplicadas.

← Si-Gan 🔍

Lista de Vacunas aplicadas

Vacuna: "Multibacterina 7"

Animal: 1709
Vacuna: Multibacterina 7
Aplico: Gabriel Moreno Reyes
Costo 50.0
Fecha: 2017-08-14 16:03:26.0

Animal: 0099
Vacuna: Multibacterina 7
Aplico: Gabriel
Costo 50.0
Fecha: 2017-08-14 16:11:20.0

Ilustración 92 Vacunas realizadas

CATÁLOGO USUARIOS.



Ilustración 93 Botón usuarios

70. En el catálogo de usuarios se podrán encontrar a los usuarios registrados por default siempre hay un usuario el cual no se puede eliminar. Este usuario es con el que ocupamos para iniciar sesión el cual se le es proporcionado. Lo que sí se puede hacer con el usuario es actualizar los datos. Para ello presione sobre la fila.



Ilustración 94 Lista de usuarios

71. Se abrirá la siguiente ventana donde se pueden apreciar los datos del usuarios, si desea cambiar un campo proceda a escribir los nuevos datos y presione sobre el botón “**actualizar**” para guardar los cambios, si desea eliminar el usuario presione el botón “**eliminar**” tome en cuenta que el primer usuario no lo podrá eliminar.



Botón actualizar

← Si-Gan

Botón eliminar

Datos del Usuario

Nombre
Gabriel Moreno Reyes

Usuario
root

Password
.....

Estatus
Activo ▾

Ilustración 95 Datos del usuario

72. Si desea agregar un nuevo usuario proceda a regresar a la lista de usuarios y presionar el botón “**agregar**”.



Ilustración 96 Botón agregar usuario

73. Cuando lo haya presionado le mandara la siguiente ventana con los campos vacíos pero para el ejemplo se utilizaran los datos que se pueden apreciar. Cuando haya rellenado los datos correctamente presione el botón agregar.

The image shows a mobile application screen for adding a user. At the top is a blue header bar with a back arrow, the text 'Si-Gan', and a green plus icon. Below the header is the title 'Agregar Usuario'. The form consists of several fields: 'Nombre' (labeled 'Nombre completo del' with an arrow pointing to the text 'Samuel Chandomi Camera'), 'Usuario' (labeled 'usuario' with an arrow pointing to the text 'usuario'), 'Password' (labeled 'Contraseña que se utilizara en el inicio de sesión' with an arrow pointing to the masked text '.....'), and 'Estatus' (labeled 'Estatus del usuario (Activo/inactivo) si le pone estado inactivo no' with an arrow pointing to the dropdown menu showing 'Activo'). A red box labeled 'Botón guardar' points to the green plus icon. Another red box labeled 'Usuario para utilizar en el inicio de sesión' points to the 'Usuario' field.

← Si-Gan +

Agregar Usuario

Nombre
Nombre completo del Samuel Chandomi Camera

Usuario
usuario

Password
.....

Estatus
Activo

Botón guardar

Nombre completo del

Usuario para utilizar en el inicio de sesión

Contraseña que se utilizara en el inicio de sesión

Estatus del usuario (Activo/inactivo) si le pone estado inactivo no

Ilustración 97 Agregar usuario