

# Sistema de información para la gestión de inventario y actividades en un hato ganadero

## Information system for inventory management and activities in a cattle farm

AHUMADA, Roberto C.<sup>1</sup>

CERVANTES, Luis<sup>2</sup>

MARTELO, Raúl J.<sup>3</sup>

### Resumen

En esta investigación, se desarrolló un Sistema de información para la gestión de inventario y actividades en un hato ganadero en el departamento de Bolívar, Colombia. El desarrollo se llevó a cabo bajo las metodologías SCRUM y RUP. La población de estudio estuvo conformada por los dueños de hatos ganaderos y la muestra fue de 40 ganaderos. Como resultado, el sistema le permitirá al ganadero, gestionar de forma óptima los animales que tiene a su cargo.

**Palabras clave:** ganadería, inventario, información, android

### Abstract

In this research, an information system was developed for the management of inventory and activities in a cattle farm in Bolívar, Colombia. The development was carried out under the SCRUM and RUP methodologies. The study population was made up of the owners of cattle farms, and the sample was 40 farmers. Regarding the results, the system will allow the farmer to optimally manage the animals in his charge.

**key words:** livestock farming, inventory, information, android

---

## 1. Introducción

El avance tecnológico se ha transformado en un medio para el desarrollo a nivel global, por lo cual permite la generación de beneficios competitivos en diversos sectores como la educación y la salud (Martelo, Ponce, & Acuña, 2015), y facilita en gran medida la forma de realizar las actividades tradicionales (Medina, 2014). En el sector agropecuario cada día se utiliza la tecnología como herramienta para mejorar los procesos, especialmente en lo referente al procesamiento de datos recogidos desde dispositivos electrónicos en granja para administrar las operaciones relacionadas al control de los animales (Vranken & Berckmans, 2017). En este sentido, el uso de la tecnología en la ganadería fomenta la optimización de los procesos para poderlos monitorear desde un ordenador o teléfono inteligente (Cevallos & González, 2018). El proceso de gestión en un hato ganadero consta de varias actividades, se registra el ganado y de los activos del hato, utilizando formularios, hojas de cálculo o

---

<sup>1</sup> Ingeniero de sistemas. Facultad de Ingeniería. Universidad de Cartagena. rcav356@hotmail.com

<sup>2</sup> Ingeniero de sistemas. Facultad de Ingeniería. Universidad de Cartagena. luis.cervantes.ortega@gmail.com

<sup>3</sup> PhD (e) – Magister en Informática. Líder del grupo de investigación INGESINFO. Profesor de planta del Programa Ingeniería de Sistemas de la Universidad de Cartagena. e-mail: rmartelog1@unicartagena.edu.co

programas informáticos que permitan administrar la información en general (Nallar, Rolón & Mollericona, 2017). Además, se debe consolidar un historial del ciclo de vida del ganado realizando control sobre la alimentación, vacunas, desparasitación, apareamiento, parto y finalmente sacrificio.

Para apoyar el manejo de información concerniente a la ganadería se han creado algunos sistemas, como es el caso en Ponnusamy et al (2016), donde con un sistema experto se apoya el control y la trazabilidad en el sistema de producción lechera de bovinos y búfalos. De igual manera, en Valenzuela (2016), desarrollaron una solución que incluye el ingreso de transacciones, inventario, control animal, control de potreros, clientes y proveedores, ingresos y egresos con sus respectivos reportes. En Galano & Novales (2014), se implementó un sistema informático para la gestión y control del ganado vacuno y la inseminación artificial de esta especie. Por otro lado, existen softwares como Bertha, Progan, Ganader SG, entre otros, los cuales pueden ser utilizados para la gestión y manejo en hatos ganaderos, sin embargo, aunque la existencia de estas herramientas contribuye al mejoramiento de procesos agroindustriales e incrementa el conocimiento en agricultores, hay poco interés en su implementación (Gómez-Prada, et al., 2019).

En algunos casos, el manejo de la información en los hatos ganaderos se limita a los registros manuales o mentales, realizados por el administrador del hato durante visitas periódicas, lo cual resulta ser poco efectivo, debido a que el propietario no dispone de herramientas que le permitan estar al tanto de las actividades y necesidades del hato (Carvajal, 2018). Además, la cantidad de documentos generados en el registro implica un alto riesgo de pérdida de información por el deterioro de los mismos, además de los cambios frecuentes que se presentan en el interior del hato. Actividades como el traslado de una res en estado de embarazo con la finalidad de llevar control del animal en dicha etapa y además mantenerla en condiciones óptimas de salud y saneamiento, o de un grupo específico que requiera un ser trasladado hasta otro terreno debido a condiciones climáticas o el riesgo de sufrir daño a la integridad del ganado, implica modificar una y otra vez el informe de inventario, borrando, corrigiendo y actualizando hasta terminar el día (Silva, 2019).

Por los inconvenientes anteriormente planteados y por los beneficios que ofrece la integración de las TIC en el sector pecuario, el objetivo de esta investigación fue implementar un sistema de información para la gestión de inventario y actividades en un hato ganadero a través de herramientas de software basadas en RUP (por su siglas en inglés de Rational Unified Process) utilizando plataformas móviles (Android). La investigación propuesta servirá de base para la construcción e implementación de un software de calidad, que atienda los requerimientos y las necesidades de los clientes, que gestione de manera rápida y eficaz el manejo de inventarios y la asignación de actividades, a través de un sistema que permita visualizar la información individual de las reses y la asignación de las mismas a una zona del hato dependiendo de sus características y necesidades.

---

## 2. Metodología

La presente investigación se clasificó como aplicada, debido a que se orientó a la solución de un caso puntual en el manejo y disponibilidad de la información (Hernández, Fernández, & Baptista, 2014). En este caso se refiere a la información sobre inventario y actividades en hatos ganaderos que como caso de estudio se tomó al departamento de Bolívar. En cuanto al enfoque, se trató del cualitativo, porque se utilizaron técnicas interactivas de recolección de información como la entrevista (Galeano, 2004), además se realizó un análisis de la forma y métodos de distribución, con el fin de clasificar reses en por categorías y por las necesidades de cada una de ellas. Como técnicas de recolección de información se aplicó la entrevista a un experto en el tema y visitas de campo. Como fuentes secundarias de información, se acudió a libros, artículos y revistas para la obtención de una base de conocimientos en el área y descubrir tecnologías apropiadas para la población.

La población para este proyecto estuvo conformada por los dueños de hatos ganaderos en el departamento de Bolívar, como muestra se tomaron cuarenta (40) voluntarios entre los ganaderos de la zona interesados en

implementar la solución planteada. Por otro lado, se aplicó RUP, como metodología de desarrollo software, la cual es un procedimiento de desarrollo de software que permite un enfoque disciplinado para la asignación de tareas y responsabilidades dentro de un proyecto de desarrollo (Kruchten, 2007). Por otro lado, se decidió combinarla con la metodología SCRUM, la cual establece un proceso empírico, iterativo e incremental de desarrollo que intenta adquirir beneficios en cuanto a los procesos definidos (López, Rivera & Palomino, 2015). A continuación, se describe la adaptación realizada para estas metodologías:

## **2.1. Fase de Inicio**

El objetivo de esta fase fue definir el alcance del software, para lograrlo se realizó una primera iteración que abarcó entrevistas a fuentes primarias para el levantamiento de requerimientos. En una segunda iteración se realizó una revisión sistemática basada en fuentes secundarias con la intención de evaluar funcionalidades de soluciones tecnológicas existentes y estudios anteriores referente al tema.

## **2.2. Fase de elaboración**

Esta fase realizada en tres iteraciones, permitió definir los requisitos funcionales y no funcionales del sistema, realizar el análisis y diseño del software propuesto, determinar la arquitectura y modelar los artefactos relevantes que describen los componentes de la herramienta utilizando como herramientas el modelo de vistas arquitectónicas 4+1 y el lenguaje de modelado unificado UML.

## **2.3. Fase de Desarrollo**

La finalidad de esta fase fue desarrollar las funcionalidades del sistema mediante tres iteraciones en las cuales se tuvo en cuenta el alcance, los roles de usuarios definidos y los artefactos de diseño propuestos en la fase anterior. Se seleccionaron algunos Casos de Uso (a partir de historias de usuario en la metodología SCRUM), se analizaron, diseñaron y se procedió al desarrollo y a la realización de las respectivas pruebas.

## **2.4. Fase de transición**

En esta fase se aseguró el correcto funcionamiento, la disponibilidad del software para los usuarios finales, el ajuste de errores y defectos encontrados en las pruebas de aceptación, la capacitación a usuarios y provisión del soporte técnico.

---

## **3. Resultados**

En esta sección se detalla el proceso mediante el cual se llevó a cabo la investigación realizada y se establecen los hallazgos en cada una de las fases de las metodologías utilizadas, para la cual es importante recordar que se combinaron dos metodologías, por un lado, se siguió el estándar de desarrollo por fases de RUP, por el otro se utilizó el método iteracional de SCRUM, para definir y adaptar los requerimientos según las necesidades del usuario final.

Los procesos que serán descritos a continuación, hacen referencia a cómo se realizaron los componentes del proyecto de investigación, iniciando por el análisis del problema, diseño del sistema, codificación de la aplicación y realización de pruebas funcionales, para el desarrollo del producto.

### **3.1. Fase de inicio**

El análisis del problema es la fase inicial dentro del proceso de desarrollo de software, en ella se analiza la problemática y los factores que conllevan a la necesidad de la construcción de una solución informática. Utilizando los resultados obtenidos luego del análisis del problema, se procedió a realizar la especificación de requerimientos teniendo en cuenta el formato IEEE-830, lo que conllevó a que se detallaran los requerimientos

funcionales y no funcionales que debía contener la aplicación desarrollada. Como producto de esta interpretación se obtuvo lo descrito en la Tabla 1 y 2.

**Tabla 1**  
Especificación de requerimientos funcionales

Identificador	Nombre del requisito funcional	Descripción	Prioridad
RF01	Gestionar sesión	El sistema debe ser capaz de ingresar con un usuario y clave pertenecientes a un rol específico como: Administrador, Capataz, empleado	Media
RF02	Gestionar Semovientes	El sistema debe permitir crear, modificar, eliminar semovientes	Alta
RF03	Gestionar Hato	El sistema debe permitir crear, modificar, eliminar Hatos	Alta
RF04	Gestionar Finca	El sistema debe permitir crear, modificar, eliminar Fincas	Alta
RF05	Trasladar Semovientes	El sistema debe permitir trasladar semovientes entre Hatos	Alta
RF06	Gestionar Actividades	El sistema debe permitir crear, modificar(realizar, posponer) Actividades	Alta
RF07	Mostrar Estado Actividades	El sistema debe mostrar actividades en proceso y realizadas	Alta

Fuente: Autores

En la tabla 2 se pueden apreciar los requisitos no funcionales que se tuvieron en cuenta al momento de desarrollar el software.

**Tabla 2**  
Listado de requisitos no funcionales

Identificador	Nombre del requisito no funcional	Descripción
RNF01	Eficiencia	El sistema debe ser capaz de procesar un número alto de peticiones.
RNF02	Seguridad lógica y de datos	La seguridad del sistema está dada por el uso de contraseñas, las cuales permitirán que el acceso al sistema sea dado sólo a las personas que tienen autorización. Lo que conlleva a la creación de roles y a la asignación a cada usuario dependiendo su funcionalidad
RNF03	Disponibilidad	El sistema debe permanecer disponible para los usuarios, de tal modo que pueda ser usado en cualquier momento.
RNF04	Tiempos de Respuesta	El sistema debe tener un tiempo de respuesta máximo acorde a los requisitos del hardware
RNF05	Facilidad de uso	El sistema debe ser fácil de utilizar para cualquier tipo de usuario.

### 3.2. Fase de elaboración

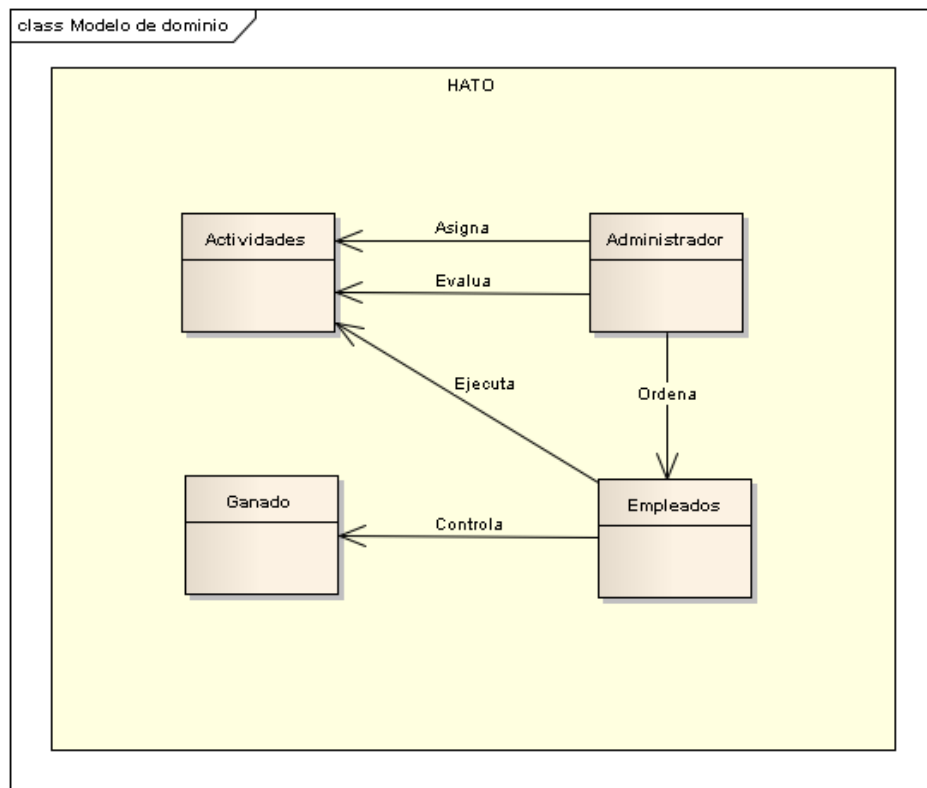
En esta etapa se diseñaron los diagramas que definieron la arquitectura del sistema construido, este diseño permitió sentar las bases para el proceso de codificación del software. Adicionalmente, se diseñaron las vistas a través del uso de la especificación de requerimientos elaborada anteriormente. En la primera iteración se propuso el modelo de dominio el cual se verificó con base en los requerimientos identificados en la fase anterior. A continuación, se describirán los diagramas obtenidos.

#### 3.2.1. Modelo de Dominio

El modelo de dominio permitió analizar la situación problema en un contexto real, de esta forma, se entiende la forma en que se relacionan los componentes del sistema y de los actores que interactúan en dicho ambiente. En

la Figura 1, se evidencian los conceptos que forman parte de este documento, con lo cual, el modelo del negocio es identificado y así, facilitar el análisis de la situación para ejecutar completamente de las etapas de desarrollo de software.

**Figura 1**  
Modelo de dominio

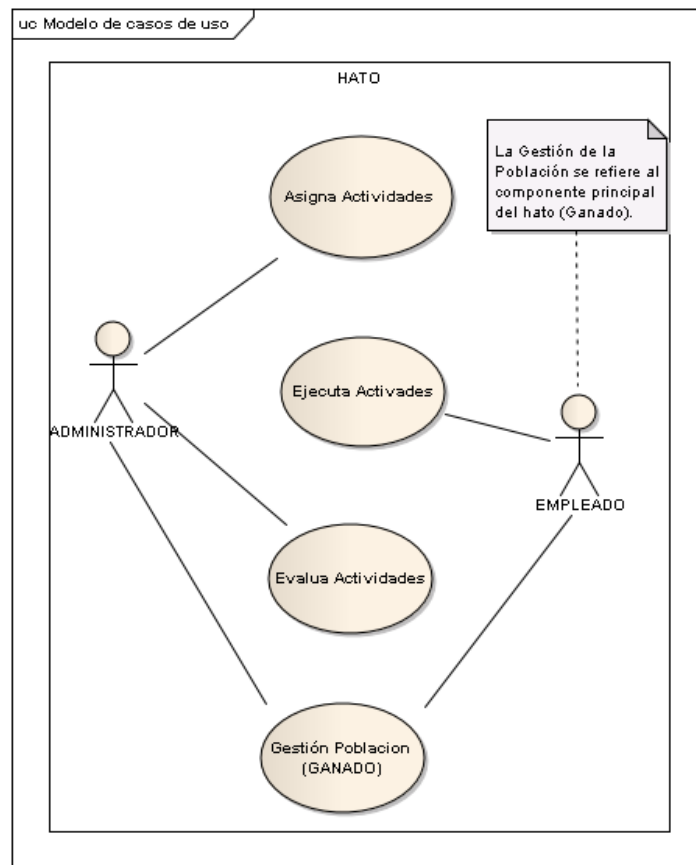


Fuente: Autores

### 3.2.2. Casos de Uso (General)

Los casos de usos guían la ejecución de la metodología de desarrollo de software RUP, son representados por historias de usuario en SCRUM y por lo tanto su definición debe ser tratada con planificación y exactitud, para determinar la correcta interacción de un actor con el sistema. En la Figura 2, se puede visualizar el diagrama que corresponde a los casos de uso. En el diagrama se pueden observar dos actores y 4 casos de uso, donde el Administrador asigna actividades, evalúa actividades y gestiona el ganado, mientras que el empleado ejecuta las actividades y también puede gestionar el ganado. Este proceso corresponde a una segunda iteración en la cual se verificó su pertinencia teniendo en cuenta el modelo de dominio planteado anteriormente.

**Figura 2**  
Diagrama de casos de uso



Fuente: Autores

### 3.3. Fase de construcción

Esta etapa hace referencia a la codificación de la aplicación, este proceso fue realizado con base en los diagramas elaborados en la fases anteriores. La funcionalidad del sistema es construida en esta etapa, por lo cual, los artefactos generados son el código fuente y la interfaz gráfica. En una primera iteración se procedió a preparar el ambiente de dearrollo, es decir, a instalar las herramientas necesarias para el desarrollo de la aplicación. Este sistema de información propuesto para la gestión de inventario y actividades en un hato ganadero (Siggaha), se pensó y desarrolló para facilitar la Gestión de Inventario y actividades al interior de un Hato Ganadero. A continuacion, en la Figura 3 se muestra un fragmento de código que muestra la funcionalidad de una Actividad.

**Figura 3**  
Fragmento de código de la Funcionalidades de una Actividad

```
public class ActivityResource {

    private static final Logger LOG = Logger
        .getLogger(ActivityResource.class.getName());

    private final ActivityService service = ServiceFactory.getActivityService();

    @GET
    public List<ActivityData> getAllActivities(@QueryParam("assignedToId") Long id) {
        if (id != null)
            return service.findByAssignedTo(id);

        return service.getAll();
    }

    @GET
    @Path("{activityId}")
    public ActivityData getActivityById(@PathParam("activityId") Long id) {
        return service.get(id);
    }

    @POST
    public Response saveActivity(ActivityData data,
        @HeaderParam("Authorization") String authHeader,
        @Context UriInfo uriInfo) {
        LOG.info("Save a activity");
        // String authToken = authHeader.replace("Bearer ", "");
        // try {
```

Fuente: Autores

**Figura 4**  
Fragmento de código de la Funcionalidades de una Granja

```
@Produces(MediaType.APPLICATION_JSON)
@Consumes(MediaType.APPLICATION_JSON)
public class FarmResource {

    private final FarmService service = ServiceFactory.getFarmService();

    @GET
    public List<FarmData> getAllFarms() {
        List<FarmData> farms = service.getAll();
        return farms;
    }

    @GET
    @Path("{farmId}")
    public FarmData getFarm(@PathParam("farmId") Long farmId) {
        FarmData farmData = service.get(farmId);
        return farmData;
    }

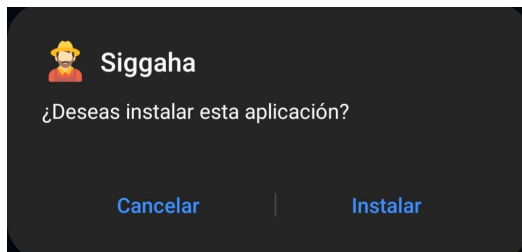
    @POST
    public FarmData addNewFarm(FarmData data) {
        FarmData newFarmData = service.add(data);

        return newFarmData;
    }
}
```

Fuente: Autores

Al ser una aplicación para dispositivos móviles (Android Versión 7 o superior), la primera actividad a realizar para hacer uso de la aplicación, consiste en la instalación del APK (Paquete de Aplicación de Android), como se observa en la Figura 5.

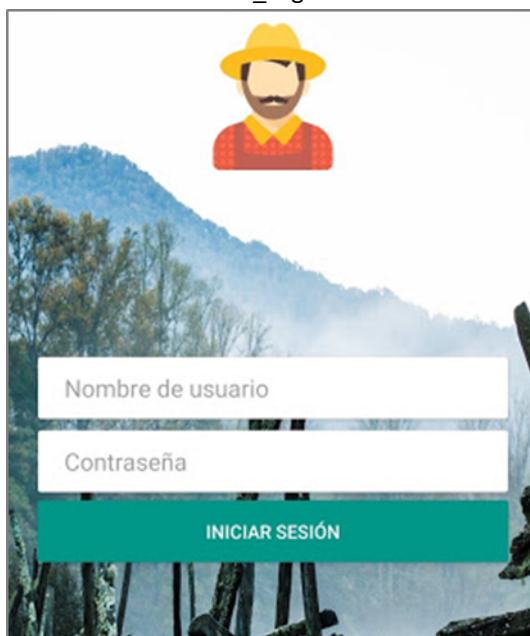
**Figura 5**  
Instalación APK



Fuente: Autores

Una vez Instalada la aplicación, se inicia sesión. Por defecto se crea un Usuario Administrador. Posteriormente en la iteración 2 de la fase de construcción, se procedió a implementar los procesos de ingreso y cierre de la aplicación.

**Figura 6**  
Gui\_Login

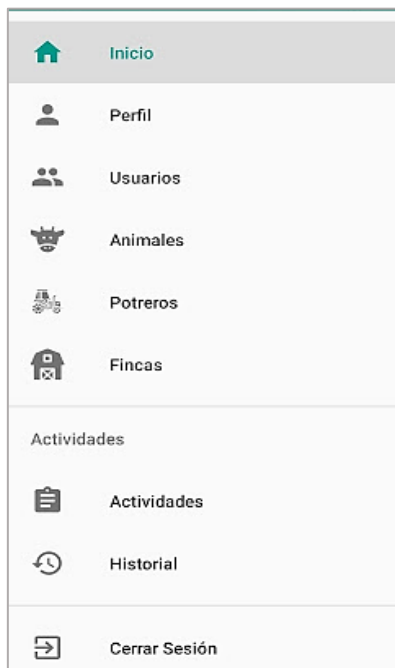


Fuente: Autores

Se inicia Sesión y se despliega la siguiente ventana y menú principal ( ver Figura 7).



**Figura 7**  
Menú Principal

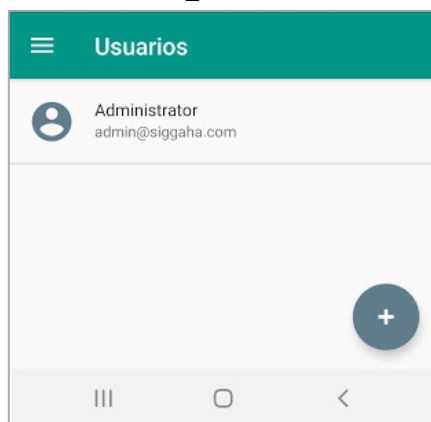


Fuente: Autores

Esta vista despliega las opciones disponibles. En primer lugar, se ingresa la información para crear nuevos usuarios. Como se puede apreciar, la interfaz de usuario cumple con los estándares de ANDROID 7.0 en adelante.

Para agregar un usuario nuevo, se selecciona la opción usuarios y se despliega la ventana que aparece en la Figura 8.

**Figura 8**  
Gui\_Usuarios



Fuente: Autores

Luego se selecciona el botón (+) y se procede a llenar el formulario con los datos que aparecen en la Figura 9.

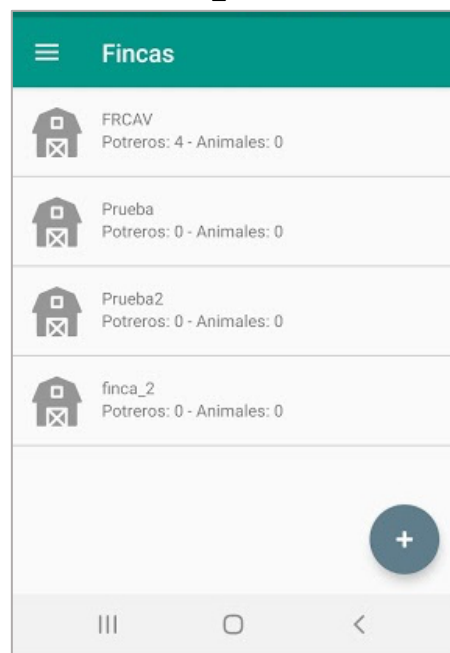
**Figura 9**  
Gui\_Agregar\_Usuario



Fuente: Autores

Con los usuarios necesarios ya creados, se procede a crear Fincas, Potreros y Animales. Las Fincas, corresponden al terreno físico, Los potreros a las divisiones físicas del terrero. Para agregar una finca, se selecciona la opción Finca y aparece una ventana como la que se observa en la Figura 10.

**Figura 10**  
Gui\_Fincas



Fuente: Autores

Se selecciona el botón (+) y se digitan los datos solicitados en el formulario, con la información de la Finca, los datos solicitados son los que se observan en la Figura 11.

**Figura 11**  
Gui\_Crear Finca

×

**Nueva Finca**

GUARDAR

(RANDOM)

Nombre

**La cabaña**

Descripción

**finca Dr Galo Viana**  
**30 hectáreas. en la colina ( ARJONA)**

Fuente: Autores

Se selecciona la opción Guardar y automáticamente se actualiza la Cantidad de fincas creadas. Una vez creadas las fincas, se procede a crear los potreros, para lo cual se accede al menú Potreros y aparece la ventana que se observa en la Figura 12.

**Figura 12**  
Gui\_Potreros

☰

**Potreros**

POTRERO 1  
Animales: 7

POTRERO 2  
Animales: 11

potrero 3  
Animales: 8

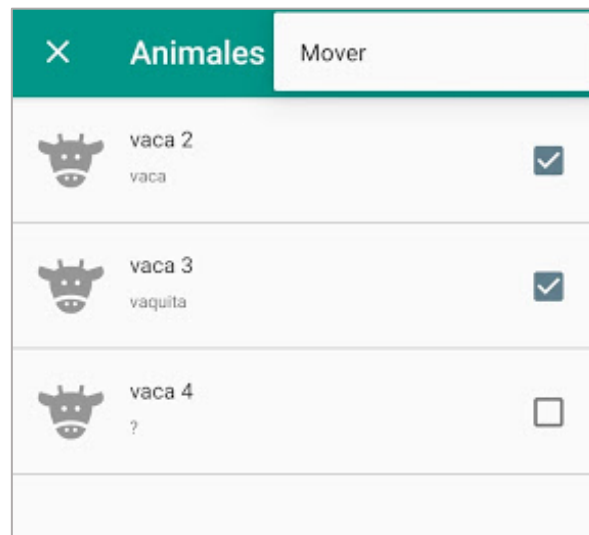
Potrero 4  
Animales: 8

+

Fuente: Autores

Para agregar un nuevo potrero se selecciona el Botón (+) y se llenan los datos del formulario. Al guardar, se actualiza la lista de potreros, si se desea realizar movimientos entre potreros, se selecciona la opción ver animales. Seguido se mostrarán los semovientes asignados al potrero seleccionado, ver la Figura 13.

**Figura 13**  
Gui\_Mover\_Animal

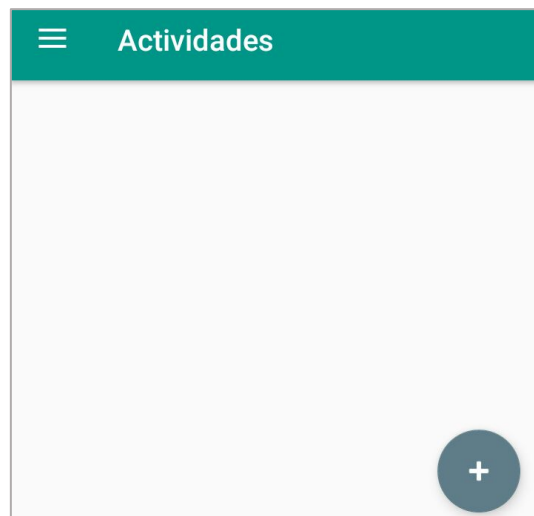


Fuente: Autores

Se seleccionan los animales que se desean mover y se elige la opción mover. Lo cual actualiza el menú de potreros con el número de animales asignados.

Con la Población definida, se procede a crear y asignar las Actividades en el menú actividades (ver Figura 14).

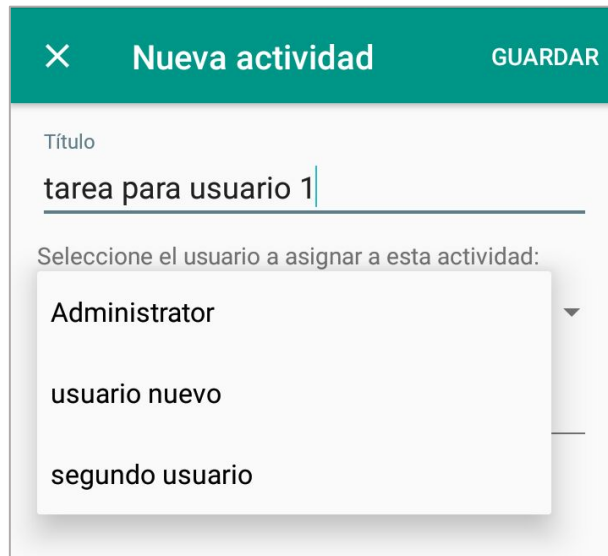
**Figura 14**  
Gui\_Actividades



Fuente: Autores

Se crea la primera actividad y se asigna el usuario encargado de realizarla (ver Figura 15).

**Figura 15**  
Gui\_Crear\_Actividad




The screenshot shows a mobile application interface for creating a new activity. At the top, there is a green header bar with a close icon (X), the title 'Nueva actividad', and a 'GUARDAR' button. Below the header, there is a text input field for the title, which contains the text 'tarea para usuario 1'. Underneath the title field, there is a label 'Seleccione el usuario a asignar a esta actividad:' followed by a dropdown menu. The dropdown menu is open, showing three options: 'Administrator', 'usuario nuevo', and 'segundo usuario'. The 'Administrator' option is currently selected.

Fuente: Autores

Posteriormente, se selecciona en la lista de actividades pre configuradas la que se desea asignar, así como el Potrero en que será realizada. En la Figura 16, se observa la forma de asignar una actividad.

**Figura 16**  
Gui\_Asignar\_Actividad



The screenshot shows the same 'Nueva actividad' form, but now with more fields filled out. The title field contains 'Desmonte tractor potrero rcav'. The user dropdown menu is still set to 'Administrator'. Below the user selection, there is a 'Descripción' field with the text 'Desmonte (Tractor y/o machete)'. At the bottom, there is a location selection field with a house icon and the text 'potrero 3' and 'Animales: 2'. The 'Animales: 2' part is in a smaller font and appears to be a sub-selection or detail for the location.

Fuente: Autores

Como se observa en la Figura 16, se asigna un título a la actividad, se selecciona el usuario que va a ejecutar la actividad, la descripción, la actividad y el lugar.

Para realizar la actividad, se selecciona de la lista, la actividad a implementar y se mostrará la información correspondiente al tipo de actividad que se seleccione. Se digitan los datos que pide el formulario y se selecciona la opción realizar ( ver Figura 17).

**Figura 17**  
Realizar actividad



Actividades	
Fecha inicio	Fecha fin
Personal Utilizado	+
Trabajador 1	
Nombre	
Costo día	Días trabajados
Materiales Utilizados	
Rollos alambre grueso	Valor
Rollos alambre delgado	Valor
Cajas de grapas delga...	Valor
Cajas de grapas grues...	Valor

POSPONER REALIZAR

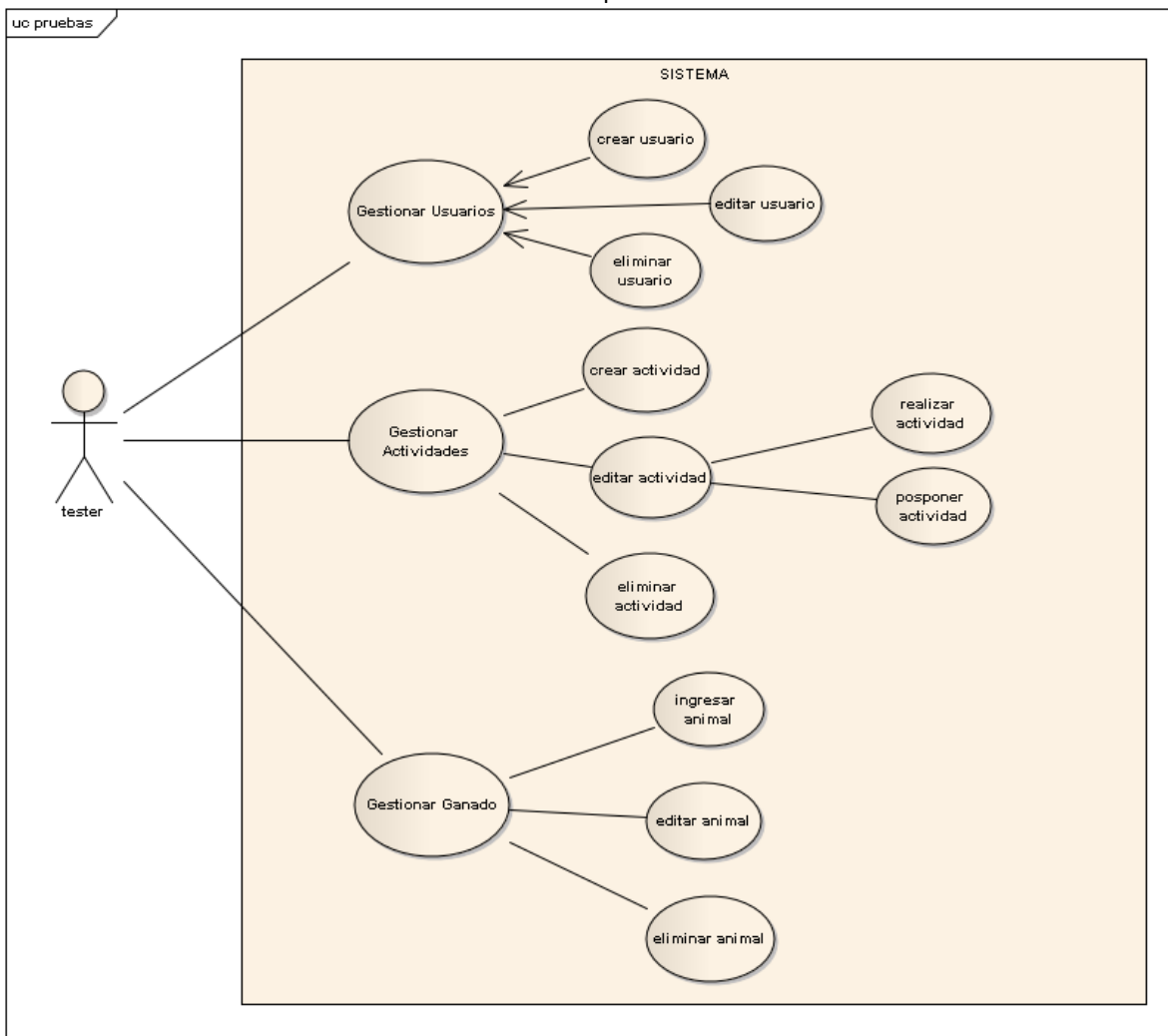
Fuente: Autores

Una vez realizada la actividad, pasa a la ventana de historial de actividades, donde se puede ver el resumen de la actividad, valor e información cargada.

### 3.4. Pruebas Del Sistema

Una de las ventajas de la metodología Scrum es la posibilidad de realizar pequeñas entregas funcionales que permiten evaluar el desarrollo del aplicativo por etapas de forma que el mantenimiento y las modificaciones se realicen en el sprint siguiente, concentrando los requerimientos y abordándolos de manera organizada. Para este aplicativo se trabajaron 15 casos de prueba, representados a través de UML, cuyos resultados fueron exitosos. En la Figura 16 se puede observar el UML que representa las pruebas.

**Figura 18**  
Casos de prueba



Fuente: Autores

## 4. Conclusiones

De acuerdo a los resultados obtenidos se concluye lo siguiente:

La combinación de dos metodologías de desarrollo software fue exitosa y de gran ayuda, debido a que facilitó el proceso de análisis del problema, determinación y elaboración de requerimientos, desarrollo de arquitectura y del aplicativo y posteriormente, la etapa de pruebas funcionales. Al tomar las ventajas del desarrollo iterativo de SCRUM y la división de tareas de RUP, se obtuvo una forma de trabajo satisfactoria y ágil para el equipo y para el cliente. De igual manera, se puede concluir que el producto cumple con el objetivo propuesto y brinda una solución de calidad al problema propuesto, sirviendo como una herramienta que permite de forma sencilla, ágil y económica, el mejoramiento del sector ganadero y contribuyendo de forma circunstancial en la manera de realizar los procedimientos del sector. Además, se plantea la posibilidad de extender el producto y agregar funcionalidades para el crecimiento del valor agregado del mismo.

## Referencias bibliográficas

Carvajal, G. (2018). Manejo de registros en sistemas doble propósito. AGROSAVIA. (No. Doc. 10275)\* CO-BAC, Santafé de Bogotá.

- Cevallos, H., & González, O. (2018). Ganadería de precisión en la provincia de El Oro. Diagnóstico situacional. *Espirales Revista Multidisciplinaria de Investigación*, 2(17). DOI: <https://doi.org/10.31876/re.v2i17.263>
- Galano, R., & Novales, R. (2014). GAVIAC, Sistema para la gestión y control del ganado vacuno y la inseminación artificial. *Revista Avanzada Científica*, 17(3), 17-28. ISSN-e 1029-3450.
- Galeano, M.(2004). Diseño de proyectos en la investigación cualitativa.Fondo editorial Universidad Eafit. Primera Ed. ISBN:958-8173-78-7.
- Gómez-Prada, U., Orellana-Hernández, M., & Salinas-Ibáñez, J. (2019). Apropiación de Sistemas de Tecnologías de la Información para toma de Decisiones de Productores Agroindustriales Basada en Videojuegos Serios. Una Revisión. *Información tecnológica*, 30(5), 331-340. DOI: <http://dx.doi.org/10.4067/S0718-07642019000500331>
- Hernández, R., Fernández, C., & Baptista, P. (2014). Metodología de la investigación. México: Mc Graw Hill Education.
- Kruchten, P. (2007). *The Rational Unified Process: An Introduction*. Addison-Wesley Professional. Boston. ISBN: 0321197704.
- López, L. , Rivera, M. , & Palomino, N. (2015). Análisis de aplicaciones empleando la computación en la nube de tipo PaaS y la metodología ágil Scrum. *Industrial data*, 18(1), 149-160. ISSN: 1560-9146 (Impreso) / ISSN: 1810-9993 (Electrónico)
- Martelo, R., Ponce, A., & Acuña, F. (2015). Metodológica para el Diseño de un Plan Estratégico Informático en Instituciones de Educación Superior. *Formación universitaria* 9(1), 91-98. DOI: <http://dx.doi.org/10.4067/S0718-50062016000100010>
- Medina, I. (2014). Estado del arte de las metodologías y modelos de los Objetos Virtuales de Aprendizaje (OVAS) en Colombia. *Entornos*, (28), 93-107. DOI: <https://doi.org/10.25054/01247905.528>
- Nallar, R., Rolón, W., & Mollericona, J.(2017). Manual para la gestión de una ganadería sostenible. Wildlife Conservation Society. La Paz. Bolivia, 124.
- Ponnusamy K.,Sriram, N., y otros. (2016).Effectiveness of cattle and buffalo expert system in knowledge management among the farmers, *Indian Journal of Animal Sciences*, ISSN: 0367-8318, 86(5), 604-608.
- Silva, M.(2019). Uso de IoT e loNT en animales: Estudio de herramientas de recolección de datos para el análisis y comparación de datos vinculados a la sanidad animal (Tesis Doctoral, Universidad Nacional de La Plata). Recuperado de: <http://sedici.unlp.edu.ar/handle/10915/88952>
- Vranken, E., & Berckmans, D. (2017). Precision livestock farming for pigs. *Animal Frontiers*, Volume 7(Issue 1), 32–37. DOI: <https://doi.org/10.2527/af.2017.0106>
- Valenzuela, R. (2016). Sistema informático para la gestión ganadera bovina en Guatemala. *Revista de la Escuela de Estudios de Postgrado*, 1(1), 81-85.

Esta obra está bajo una Licencia Creative Commons  
Atribución-NoCommercial 4.0 International

