



**Aplicación móvil Android para gestión de empleados, de bovinos e informes  
sobre la producción de leche de la hacienda Indonesia.**

**Juan Camilo Fonseca Camargo**

Código: 10892217044

**Fabian Saavedra Saavedra**

Código: 10892219460

**Universidad Antonio Nariño**

Programa Especialización en Ingeniería de Software

Facultad de Ingeniería de Sistemas

Bogotá, Colombia

2022

**Aplicación móvil Android para gestión de empleados, de bovinos e informes  
sobre la producción de leche de la hacienda Indonesia.**

**Juan Camilo Fonseca Camargo - Fabian Saavedra Saavedra**

Proyecto de grado presentado como requisito parcial para optar al título de:

**Especialista Ingeniería de Software**

Director (a):

Msc. Dianalin Neme Prada

Msc. Iván Rodrigo Romero

**Universidad Antonio Nariño**

Programa Especialización en Ingeniería de Software

Facultad de Ingeniería de Sistemas

Bogotá, Colombia

2022

## NOTA DE ACEPTACIÓN

El trabajo de grado titulado

\_\_\_\_\_.

Cumple con los requisitos para optar

Al título de \_\_\_\_\_.

\_\_\_\_\_

Firma del Tutor

\_\_\_\_\_

Firma Jurado

\_\_\_\_\_

Firma Jurado

## CONTENIDO

<b>PRELIMINARES .....</b>	<b>9</b>
<b>RESUMEN.....</b>	<b>10</b>
<b>ABSTRACT.....</b>	<b>11</b>
<b>INTRODUCCIÓN .....</b>	<b>12</b>
<b>1. PLANTEAMIENTO DEL PROBLEMA .....</b>	<b>14</b>
1.1. DESCRIPCIÓN DEL PROBLEMA .....	14
1.2. FORMULACIÓN DEL PROBLEMA.....	16
<b>2. OBJETIVOS .....</b>	<b>17</b>
2.1. OBJETIVO GENERAL .....	17
2.2. OBJETIVOS ESPECÍFICOS.....	17
<b>3. MARCO DE REFERENCIA.....</b>	<b>18</b>
3.1. ESTADO DEL ARTE.....	18
3.1.1. CONTROL GANADERO.....	19
3.1.1.1. VENTAJAS .....	19
3.1.1.2. DESVENTAJAS.....	19
3.1.2. BOVCONTROL.....	20
3.1.2.1. VENTAJAS .....	20
3.1.2.2. DESVENTAJAS.....	20
3.1.3. BOVINAPP .....	20
3.1.3.1. VENTAJAS .....	21
3.1.3.2. DESVENTAJAS.....	21
3.2. COMPONENTE DE INNOVACIÓN.....	22
3.3. IMPACTO.....	22
3.3.1. ECONÓMICO.....	23
3.3.2. SOCIAL.....	23
3.3.3. TÉCNICO.....	23
3.4. MARCO TEÓRICO.....	24
3.4.1. BOVINO.....	24
3.4.2. ORDEÑO .....	24
3.4.3. APP.....	24
3.4.4. JAVA ANDROID .....	25
3.4.5. POSTGRESS SQL .....	25
3.4.6. SPRING BOOT .....	25

3.4.7. JWT .....	26
3.4.8. SQLITE .....	27
3.4.9. ANDROID STUDIO .....	28
3.4.10. ECLIPSE .....	28
3.4.11. HTTP – HTTPS .....	29
<b>4. METODOLOGÍA.....</b>	<b>29</b>
4.1. METODOLOGÍA KANBAN .....	29
4.2. REGLAS DE LA METODOLOGÍA .....	29
4.3. TABLERO .....	30
<b>5. PROCESO DE SOFTWARE .....</b>	<b>33</b>
5.1. IDENTIFICACIÓN DE SATKEHOLDERS .....	33
<b>5.2. REQUERIMIENTOS.....</b>	<b>33</b>
<b>5.2.1. REQUERIMIENTOS FUNCIONALES .....</b>	<b>34</b>
5.2.1.1. INICIAR SESIÓN .....	35
5.2.1.2. VISUALIZACIÓN MENÚ.....	36
5.2.1.3. GESTIÓN DE EMPLEADOS .....	36
5.2.1.4. GESTIÓN DE ANIMALES .....	37
5.2.1.5. GESTIÓN DE REBAÑO.....	38
5.2.1.6. GESTIÓN DE FINCA .....	39
5.2.1.7. GESTIÓN DE MUNICIPIO .....	39
5.2.1.8. REGISTROS DE PRODUCCIÓN .....	40
<b>5.2.1.9. GESTIÓN DE INFORMES.....</b>	<b>41</b>
<b>5.2.2. REQUERIMIENTOS NO FUNCIONALES.....</b>	<b>42</b>
<b>5.3. DISEÑO Y ARQUITECTURA .....</b>	<b>43</b>
<b>5.3.1. DIAGRAMA DE COMPONENTES .....</b>	<b>43</b>
<b>5.3.2. DIAGRAMA DE CASOS DE USO.....</b>	<b>45</b>
<b>5.3.3. DIAGRAMA DE CLASES .....</b>	<b>46</b>
<b>5.3.4. DIAGRAMA DE ESTADOS.....</b>	<b>47</b>
<b>5.3.5. DIAGRAMA DE COLABORACIÓN .....</b>	<b>48</b>
<b>5.3.6. DIAGRAMA DE SECUENCIA .....</b>	<b>49</b>
<b>5.4. ARQUITECTURA DE ALTO NIVEL.....</b>	<b>50</b>
<b>6. CONSTRUCCIÓN .....</b>	<b>52</b>
<b>6.1. INSTALACIÓN Y CONFIGURACIÓN .....</b>	<b>53</b>
<b>6.2. BACK-END .....</b>	<b>55</b>
<b>6.3. FRONT-END .....</b>	<b>57</b>
<b>6.4. PRINCIPIOS DE SEGURIDAD.....</b>	<b>59</b>
<b>7. APLICACIÓN .....</b>	<b>60</b>
<b>8. PRUEBAS.....</b>	<b>65</b>

<b>9. CONCLUSIONES .....</b>	<b>68</b>
<b>ANEXOS.....</b>	<b>69</b>
<b>REFERENCIAS BIBLIOGRÁFICAS.....</b>	<b>70</b>

## LISTA DE FIGURAS

Ilustración 1 - EA Control Ganadero .....	19
Ilustración 2 - EA BovControl .....	20
Ilustración 3 – EA BovinApp .....	20
Ilustración 4 - Kanban Metodología .....	31
Ilustración 5 - Kanban Tablero .....	32
Ilustración 6 – Diseño y arquitectura básico .....	43
Ilustración 7 - Diagrama de Componentes.....	44
Ilustración 8 - Componentes alto nivel .....	45
Ilustración 9 - Diagrama de casos de uso.....	46
Ilustración 10 - Diagrama de clases .....	47
Ilustración 11 - Diagrama de estados.....	48
Ilustración 12 - Diagrama de Colaboración .....	49
Ilustración 13 - Diagrama de secuencia .....	50
Ilustración 14 - Arquitectura de alto nivel .....	51
Ilustración 15 - Arquitectura Alto Nivel AWS .....	52
Ilustración 16 - Instancia Back-End.....	53
Ilustración 17 - Instancia RDS .....	54
Ilustración 18 - Grupo Seguridad Instancias.....	54
Ilustración 19 - Instancias VPC .....	55
Ilustración 20 - Arquitectura Back-End.....	56
Ilustración 21 - Back-End - Capa RestController .....	56
Ilustración 22 - Back-End - Capa Base de datos.....	57
Ilustración 23 - Back-End - Capa Interfaz .....	57
Ilustración 24 - Arquitectura Front-End .....	58
Ilustración 25 - Front-End – Complementos .....	59
Ilustración 26 - Seguridad - Conexión .....	60
Ilustración 27 - App Presentación.....	61
Ilustración 28 - App Inicio Sesión .....	61
Ilustración 29 - App Menú Principal .....	62
Ilustración 30 - App Listado Animales .....	62
Ilustración 31 - App Listado Empleados .....	63
Ilustración 32 - App Listado Control Lácteo .....	63
Ilustración 33 - App Informes.....	64
Ilustración 34 - App Reportes .....	64
Ilustración 35 - App Filtro Fecha.....	65

Ilustración 36 - Sonar.....	66
Ilustración 37 - Sonar 2.....	66
Ilustración 38 - Pruebas de calidad en Backend - Spring Boot.....	66

## LISTA DE TABLAS

Tabla 1 - Identificación de StakeHolders.....	33
Tabla 2 - Requerimientos Funcionales .....	34
Tabla 3 - CU - 01 - Iniciar Sesión.....	35
Tabla 4 - CU - 02 - Visualización Menú.....	36
Tabla 5 - CU - 03 - Gestión Empleados.....	36
Tabla 6 - CU - 06 - Gestión Animales .....	37
Tabla 7 - CU - 05 - Gestión de Rebaño .....	38
Tabla 8 - CU - 06 - Gestión de Finca.....	39
Tabla 9 - CU - 07 - Gestión de Municipio .....	39
Tabla 10 - CU - 08 - Registros de Producción.....	40
Tabla 11 - CU - 09 - Gestión de Informes .....	41
Tabla 12 - Requerimientos No Funcionales.....	42

## **PRELIMINARES**

Este proyecto es el resultado de un esfuerzo conjunto de todos nuestros grupos de trabajo constituyentes. Dedicamos este proyecto principalmente a Dios para que nos dio la fuerza necesaria para lograr este objetivo.

## **AGRADECIMIENTOS**

Agradecemos a nuestras familias que son las que nos animaron a iniciar estas etapas y a quienes nos acompañaron hasta su culminación, agradecemos de la misma manera a nuestros docentes que nos guiaron en este camino de aprendizaje, gracias a ellos y a la universidad logramos innovar un producto que es muy útil para la sociedad.

## RESUMEN

Actualmente, la industria ganadera y el comercio internacional de producción de leche colombiana se encuentra en crecimiento, sin embargo, la falta de implementación de nuevas tecnologías, la infraestructura y el individualismo de empresarios, impide estar nivelado a países como Estados Unidos y Argentina. Asimismo, con el auge de las tecnologías de la información es imprescindible que toda organización de cualquier sector empresarial en este caso el sector agropecuario esté al servicio y facilite el acceso a la información y toma de decisiones rápidas y precisas.

La industria ganadera no puede ignorar esta tendencia y debe estar a la vanguardia para superar las dificultades diarias. Los dispositivos móviles, que brindan los mismos beneficios que un ordenador y que están al alcance de la mano son la mejor solución. Actualmente, la mayoría de las fincas en su plan estratégico se basan en decisiones correctivas más no preventivas, lo que retrasa el proceso y baja los índices de rentabilidad.

El objetivo principal del proyecto es crear una aplicación móvil Android donde se encontrará información puntual y veraz, diseñada y construida mediante el uso de metodologías ágiles de gestión de proyectos, herramientas de pruebas automatizadas y una arquitectura escalable que pueda mantener el sistema en funcionamiento.

Palabras clave: granja, bovino, ganadería, leche, producción apk, dispositivos móviles.

## ABSTRACT

Currently, the livestock industry and international trade of Colombian milk production is growing, however, the lack of implementation of new technologies, infrastructure and individualism of entrepreneurs, prevents being level with countries such as the United States and Argentina. Likewise, with the rise of information technologies, it is essential that any organization in any business sector, in this case the agricultural sector, be at the service of and facilitate access to information and quick and accurate decision making.

The livestock industry cannot ignore this trend and must be at the forefront to overcome daily difficulties. Mobile devices, which provide the same benefits as a computer and are within reach, are the best solution. Currently, most of the farms in their strategic plan are based on corrective rather than preventive decisions, which delays the process and low profitability rates.

The main objective of the project is to create an android mobile application where you will find timely and accurate information, designed and built using agile project management methodologies, automated testing tools and a scalable architecture that can keep the system running.

Keywords: farm, bovine, cattle, livestock, milk, apk production, mobile devices.

## INTRODUCCIÓN

El mundo empresarial colombiano hoy en día cuenta con muchos dispositivos capaces de conectarse a Internet para obtener información de diversas fuentes. Esto contribuyó a la globalización e integración de la ciencia, brindando soluciones a los típicos problemas de comunicación entre empleados, clientes y proveedores, sin importar dónde se encuentren, facilitando así rutinas, procedimientos, toma de decisiones en el ámbito financiero, médico y, más recientemente, agrícola.

Según el observatorio del sector lácteo Colombia con referencia en su análisis de coyuntura 2021-2022 1er Trimestre, indica que durante 2021 se exportaron 7.290 toneladas de leche y derivados lácteos por un valor de USD 28.16 millones, un crecimiento de 58,1% en términos de volumen en relación con 2020. La coyuntura exportadora se vio favorecida por la tasa de cambio con valores promedio superiores a \$3.900/dólar y por precios internacionales al alza. Teniendo en cuenta la favorabilidad del mercado internacional y teniendo en cuenta los crecimientos locales también muestran una reacción positiva que tienen las principales cuencas lecheras (Antioquia, Cundinamarca, Nariño y Boyacá) durante el mes de marzo (2022) con aumento de 14,3% con respecto al mes anterior y 6,0% comparado con marzo de 2021 (incremento equivalente a 16 millones de litros).

Dados estos escenarios y el aumento del uso de las tecnologías de la información y comunicación (TIC) invertir en ciencia y tecnología es esencial para el crecimiento

continuo del sector agrícola de Colombia. Estas inversiones, que aumentan los rendimientos, ahorran costos y permiten un mayor control sobre las materias primas, tienen un impacto directo en la competitividad de las cadenas productivas y forman parte de una estrategia de apertura de nuevos mercados para el sector agropecuario.

Sin embargo, todavía se utilizan sistemas bastante primitivos en la ganadería y la agricultura, y una gran parte de la industria (incluyendo dueños y administradores de fincas) ve con escepticismo tales soluciones y artículos tecnológicos. Por ello, todavía existen inconvenientes y dificultades de por medio, y las soluciones propuestas no son del todo satisfactorias ni completas. El proyecto tiene como objetivo contribuir a este campo mediante el diseño y desarrollo de una aplicación móvil Android que integre e informe la producción, gestión de empleados, gestión de animales y crías integrando a los actores principales en el proceso la cual estará disponible 24/7, con información actualizada, verificada y en tiempo real.

## **1. PLANTEAMIENTO DEL PROBLEMA**

### **1.1. DESCRIPCIÓN DEL PROBLEMA**

En Colombia, la Unidad de Seguimiento de Precios de Leche (USP) del Ministerio de Agricultura y Desarrollo Rural estima que en 2021 en Colombia la producción de leche se contrajo -5,7% bordeando 6.789 millones de litros/año. Así mismo, el acopio por parte de la industria disminuyó -7,0% alcanzando 3.113 millones de litros (235 millones de litros menos que en 2020).

El balance negativo para el año 2021 respecto a la producción tiene que ver con las importaciones de leche en polvo, las cuales continuaron generando dificultades a los productores de leche, motivo por el cual Fedegan durante el 2021 realizó un proceso de solicitud para la imposición de una medida de salvaguardia bilateral a las importaciones, con el fin de proteger a los productores de leche colombianos.

Esta tendencia negativa cambiará gracias al alza internacional del precio del dólar lo que disminuiría drásticamente las importaciones de productos lácteos dando una perspectiva positiva para los productos con respecto al año que se aproxima.

Según datos de los perfiles económicos departamentales para septiembre del 2022 la composición sectorial del PIB del departamento de Boyacá ubica al sector de la

agricultura, ganadería y pesca en el tercer puesto con un porcentaje del 12.5% seguido de industrias manufactureras y por encima del PIB nacional con una diferencia porcentual del 5.1, lo cual ubica al departamento como uno de los principales proveedores en materia básica de la canasta familiar.

En la hacienda Indonesia ubicada en Paipa - Boyacá se realiza recolección de leche diaria dos veces al día de aproximadamente de 50 a 100 bovinas. La recolección de leche es efectuada por los empleados de la hacienda, en la cual se hace pasar una a una las vacas por máquina de ordeño. Durante este proceso se llenan manualmente los formatos de control de producción de leche, haciendo observaciones del estado de salud del animal y finalizando con el dato exacto en litros de la producción de leche de cada de cada bovino, colocando la cantidad recolectada junto con el nombre de cada animal y el nombre del empleado que realizó la tarea. Al final del día se ejecuta una suma manual de todas las hojas de los puestos de trabajo y se totaliza en un libro general identificando la cantidad total de producción. El procesamiento manual de la recolección hace que el trabajo sea más complejo y menos exacto en cada jornada, conllevando a pérdidas en el registro de los datos y sin conocimiento claro de la totalidad de leche producida, por parte del administrador y/o dueño de la finca.

Se identifica la falta información en tiempo real, además de registros gráficos detallados de los cuales se puedan identificar los cambios de producción, detalles del estado del animal y finalización de periodo de producción del mismo que finalmente genera fallas en la toma de decisiones preventivas, adicionalmente, durante la recolección

manual de la información y de acuerdo con el clima de la región (humedad, lluvias), repetidamente se deterioran las hojas de registro, llevando a pérdidas de información y a resultados erróneos.

Es necesario desarrollar un sistema de información que le permita a la hacienda Indonesia mantener, alimentar y asegurar la integridad de la información de producción, ofreciendo un acceso seguro y en tiempo real que permita identificar deficiencias en procesos, estados de producción individual y general de la finca.

Dadas las razones anteriormente mencionadas, este proyecto tendrá como objetivo el desarrollo de una aplicación móvil en sistema operativo Android, que permita ingresar los registros de la producción de leche, con un alcance más profundo en la administración de animales, tiempos de lactancia, salud y gestión de empleados de la hacienda.

## 1.2. FORMULACIÓN DEL PROBLEMA

¿El diseño y desarrollo de un sistema de gestión de información de animales, de empleados y producción de leche, puede optimizar el proceso de recolección de datos en la hacienda Indonesia?

## 2. OBJETIVOS

### 2.1. OBJETIVO GENERAL

Crear una aplicación móvil Android que pueda contener el proceso de gestión de empleados, animales e información de producción lechera de la hacienda Indonesia.

### 2.2. OBJETIVOS ESPECÍFICOS

- Eliminar la pérdida de información a causa de daños físicos de las hojas de registro.
- Centralizar la información de producción de leche de la hacienda con el propósito de tomar decisiones futuras generando un valor agregado al negocio.
- Crear informe de animales productivos, empleados activos, producción de leche diaria, semanal y mensual, determinando las causas de rendimiento y/o pérdidas de la producción.
- Guardar y visualizar en formato PDF la información general (edad, peso, sexo), estados de salud, producción e inseminación por animal (aplica para hembras).
- Permitir la captura de información sin conexión a internet que sincronice los datos cuando la conexión sea establecida.

### 3. MARCO DE REFERENCIA

#### 3.1. ESTADO DEL ARTE

La leche es uno de los alimentos que proporciona energía y nutrientes y uno de los pocos que se consume durante toda la vida. Para que este producto llegue a los hogares se realiza un proceso extenso que va desde la recolección hasta la distribución.

Actualmente, la industria alimentaria ha desarrollado estrategias y diferentes sistemas para lograr mantener este gran proceso, sin embargo, grandes empresas como Colanta o Alpina, dentro del ámbito nacional, recogen el consolidado en diferentes haciendas, dejando la tarea de ordeño a las fincas o haciendas de la región. Las haciendas enfrentan la tarea de mantener un control y calidad sobre la leche desde el momento en que sale de la ubre, debido a que contiene diferentes microbios e inmediatamente debe efectuarse una tarea de aseguramiento de la leche evitando que se propague alguna presencia bacteriana.

La recolección de leche se vuelve una tarea prioritaria para estas haciendas y la cual debe ser realizada de manera adecuada y los registros deben permitir obtener una información clara sobre la producción de cada bovino, para ello, existen algunas aplicaciones que han logrado ayudar este gremio y es importante conocerlas para validar sus alcances, ventajas y desventajas.

Se hace una investigación en la tienda de aplicación para la plataforma Android (Play Store) con el fin de condensar las ventajas y desventajas condensado en la tabla 1, presentes en el mercado de apps para la gestión de información de haciendas lecheras e implícitamente obtener el componente de innovación que se creara para la aplicación.

### 3.1.1. CONTROL GANADERO



Ilustración 1 - EA Control Ganadero

#### 3.1.1.1. VENTAJAS

- Seguimiento de producción y ventas de leche y carne.
- Recordatorio de compras y tareas a realizar a través de un sistema de alertas: registros de peso, desinfectar cuadras, cambiar vendajes.
- Repartir roles entre los usuarios: responsable, trabajador y veterinario.

#### 3.1.1.2. DESVENTAJAS

- Ante varios celulares, la aplicación pierde sincronía y muestra valores de producción diferentes en cada celular.
- No tiene cómo exportar los datos para realizar un análisis de la producción de leche.

### 3.1.2. BOVCONTROL



Ilustración 2 - EA BovControl

#### 3.1.2.1. VENTAJAS

- Es posible seguir la administración y el manejo del bovino desde cualquier lugar.
- Planes Profesionales. Stock de maquinaria.
- Administrador Financiero

#### 3.1.2.2. DESVENTAJAS

- Problemas de sincronización.
- Cuando se inicia sesión desde otro móvil, no las imágenes.
- No deja eliminar

### 3.1.3. BOVINAPP



Ilustración 3 – EA BovinApp

### 3.1.3.1. VENTAJAS

- Realiza un seguimiento de los nacimientos, cría, inseminaciones, muertes, ventas, tratamientos del veterinario.
- Organiza animales en rebaños y realiza un seguimiento de los movimientos.
- Gestiona ganado bovino (bovinas de carne o leche), caballos, ovejas y cabras

### 3.1.3.2. DESVENTAJAS

- No permite agregar una foto del animal.
- Membresía.

Dada esta información, se puede encontrar que existe una falencia en aplicaciones para un mercado virgen en la región que puede ser aprovechable para incursionar en tecnología y permita facilitar el proceso de recolección de leche.

Las aplicaciones mencionadas son de suscripción y los costos asociados al uso son bastantes elevados para la cantidad de producción. Por esto, la aplicación planteada en este proyecto permitirá solucionar y sistematizar la forma de recolección de leche de la hacienda.

### 3.2. COMPONENTE DE INNOVACIÓN

Este proyecto acoge los fundamentos teóricos y prácticos de la ingeniería del software y específicamente en las áreas de gestión de proyectos y arquitectura de software, que tienen como objetivo brindar las herramientas necesarias para la correcta ejecución del desarrollo de software y la elaboración de proyectos de ingeniería.

Con base en la investigación de otras aplicaciones con funcionalidades parecidas se determina que la aplicación de gestión de producción lechera para la hacienda Indonesia será:

- La condensación de información de producción o centralización de esta de manera que ningún tercero pueda acceder a esta y, por el contrario, se llegue a ella de forma segura y en tiempo real.
- Evitará la manipulación de información y asegurará la integridad de datos.
- Ingreso de información de manera offline y sincronización online.
- El dueño y/o administrador puede evidenciar que empleado realiza la tarea de ordeño a cada animal.

### 3.3. IMPACTO

El proyecto actual, permitirá la introducción de la tecnología a un sector ganadero que actualmente no tiene una solución tecnológica establecida para la gestión de la

hacienda y especialmente del control de producción de leche. Se pretende que la solución propuesta brinde la posibilidad de mejorar la administración estratégica de la hacienda. A continuación, se especifican diferentes impactos que se tendrán en la implementación de esta solución.

### 3.3.1. ECONÓMICO

Dueño y/o administrador de la finca se le presentarán disminución de tiempos de respuesta, procesamiento de información y análisis de los datos de la producción de leche de la hacienda. Cada registro tomado de forma manual debe ser actualmente digitalizado en tablas de Excel y mediante un proceso manual se logra obtener un resultado para el análisis, sin embargo, este proceso requiere tiempo y los procesos manuales pueden presentar errores al final del día que impactan directamente a nivel económico.

### 3.3.2. SOCIAL

Con el desarrollo de esta solución tecnológica se ataca un sector específico donde la tecnología y puesta en marcha de recursos tecnológicos se hacen cada vez más complejos, como lo es el sector agropecuario, donde la experiencia empírica y los métodos manuales de recolección de información hacen parte del día a día.

### 3.3.3. TÉCNICO

El equipo de desarrollo para esta aplicación se ve inmerso en un fragmento de mercado donde no se ha realizado el desarrollo de software en los procesos de producción y menos para fincas lecheras donde su producción no es masiva.

### 3.4. MARCO TEÓRICO

El marco teórico reúne los conceptos necesarios para el entendimiento del proyecto en la cual se utilizarán a lo largo del documento.

#### 3.4.1. BOVINO

Es el ganado utilizado para generar carne o leche, en este caso se mencionará sobre el bovino como el utilizado para obtención de leche (Real Academia Española, s.f.).

#### 3.4.2. ORDEÑO

Proceso mediante el cual se extrae la leche de las glándulas mamarias, llamadas ubre, de un mamífero, habitualmente del ganado lechero (Real Academia Española, s.f.).

#### 3.4.3. APP

Las computadoras de escritorio y portátiles no son las únicas que pueden ejecutar el software. Estos programas y programas no son exclusivos de estas computadoras. Las aplicaciones también funcionan en dispositivos móviles como tabletas y teléfonos inteligentes. Están especialmente diseñados para facilitar el trabajo y comprimir todo el

mundo de las herramientas y accesorios cotidianos en un solo dispositivo. Entonces un aplicación móvil es aquel software que utiliza en un dispositivo móvil como herramienta de comunicación, gestión, venta de servicios-productos orientados a proporcionar al usuario las necesidades que demande de forma automática e interactiva (Benítez, L. F. , 2015).

#### 3.4.4. JAVA ANDROID

Es un lenguaje de programación moderno, pero ya maduro destinado a hacer más felices a los desarrolladores. Es conciso, seguro, interoperable con Java y otros lenguajes, y proporciona muchas formas de reutilizar código entre múltiples plataformas para una programación productiva (Android Studio dolphin. 2022).

#### 3.4.5. POSTGRESS SQL

PostgreSQL es un poderoso sistema de base de datos relacional de objetos de código abierto con más de 35 años de desarrollo activo que le ha valido una sólida reputación por su confiabilidad, robustez de funciones y rendimiento (PostgreSQL. 2022).

#### 3.4.6. SPRING BOOT

Spring Boot facilita la creación de aplicaciones basadas en Spring independientes y de grado de producción que puede simplemente ejecutar.

Tomamos una opinión obstinada de la plataforma Spring y las bibliotecas de terceros para que pueda comenzar con el mínimo esfuerzo. La mayoría de las aplicaciones de Spring Boot necesitan una configuración mínima de Spring.

Características:

- Se pueden crear aplicaciones Spring independientes
- Incrusta Tomcat, Jetty o Undertow directamente (no es necesario implementar archivos WAR)
- Proporcione dependencias iniciales obstinadas para simplificar su configuración de compilación
- Configura automáticamente Spring y bibliotecas de terceros siempre que sea posible
- Proporciona funciones listas para producción, como métricas, controles de estado y configuración externalizada
- Absolutamente sin generación de código y sin requisitos para la configuración XML
- 

Se da más información debido a ser el lenguaje de programación utilizada para el desarrollo del proyecto.

#### 3.4.7. JWT

Son Web Token (JWT) es un formato de representación de reclamos compacto diseñado para entornos con limitaciones de espacio como HTTP Encabezados de

autorización y parámetros de consulta de URI. Los JWT codifican reclamos para ser transmitido como un objeto JSON [RFC7159] que se utiliza como la carga útil de una estructura JSON Web Signature (JWS) [JWS] o como la texto sin formato de una estructura JSON Web Encryption (JWE) [JWE], lo que permite las afirmaciones de estar firmadas digitalmente o de proteger su integridad con un Mensaje de Código de Autenticación (MAC) y/o encriptado. Los JWT son siempre representados usando la Serialización Compacta JWS o la Compacta JWE Publicación por entregas (Jones, M., Bradley, J., & Sakimura, N. 2015).

#### 3.4.8. SQLITE

SQLite es una biblioteca en lenguaje C que implementa un motor de base de datos SQL pequeño, rápido, autónomo, de alta confiabilidad y con todas las funciones. SQLite es el motor de base de datos más utilizado en el mundo. SQLite está integrado en todos los teléfonos móviles y en la mayoría de las computadoras y viene incluido dentro de innumerables otras aplicaciones que la gente usa todos los días. Más información.

El formato de archivo SQLite es estable, multiplataforma y compatible con versiones anteriores y los desarrolladores se comprometen a mantenerlo así hasta el año 2050. Los archivos de base de datos SQLite se usan comúnmente como contenedores para transferir contenido enriquecido entre sistemas y como un formato de archivo a largo plazo para datos. Hay más de 1 billón (1e12) de bases de datos SQLite en uso activo. El código

fuelle de SQLite es de dominio público y es gratuito para todos para cualquier propósito (SQLite Home Page., s/f).

#### 3.4.9. ANDROID STUDIO

Android Studio es el IDE oficial para el desarrollo de Android e incluye todo lo necesario para compilar apps para Android para ejecutar a través de Java o Kotlin. (Android Studio dolphin. 2022, julio 10)

#### 3.4.10. ECLIPSE

Eclipse es una plataforma de software compuesto por un conjunto de herramientas de programación de código abierto multiplataforma para desarrollar lo que el proyecto llama Aplicaciones de Cliente Enriquecido, opuesto a las aplicaciones cliente-liviano basadas en navegadores. Esta plataforma, típicamente ha sido usada para desarrollar entornos de desarrollo integrados (del inglés IDE), como el IDE de Java llamado Java Development Toolkit (JDT) y el compilador (ECJ) que se entrega como parte de Eclipse (y que son usados también para desarrollar el mismo Eclipse).

Eclipse es también una comunidad de usuarios, extendiendo constantemente las áreas de aplicación cubiertas. Un ejemplo es el recientemente creado Eclipse Modeling Project, cubriendo casi todas las áreas de Model Driven Engineering.

Eclipse fue desarrollado originalmente por IBM como el sucesor de su familia de herramientas para VisualAge. Eclipse es ahora desarrollado por la Fundación Eclipse, una organización independiente sin ánimo de lucro que fomenta una comunidad de código abierto y un conjunto de productos complementarios, capacidades y servicios. (contributors; Eclipse software, s/f).

#### 3.4.11. HTTP – HTTPS

Es un protocolo de comunicación destinado a la transferencia segura de datos denominada textualmente “Transferencia segura de Hypertexto”. A través de este protocolo se realizará la conexión y se publicará el API de consumo de los servicios que utiliza el FrontEnd.

## 4. METODOLOGÍA

### 4.1. METODOLOGÍA KANBAN

El equipo de desarrollo para esta aplicación utilizará la metodología kanban en la cual se van completando las tareas por medio de un tablero y tarjetas con las tareas pendientes. Esta metodología tiene la ventaja de fácil identificación de las responsabilidades y se adapta bien al tamaño del equipo de desarrollo.

### 4.2. REGLAS DE LA METODOLOGÍA

Para lograr el éxito del proyecto es necesario cumplir adecuadamente con la metodología utilizada, para este caso se mencionará las reglas claves a tener en cuenta:

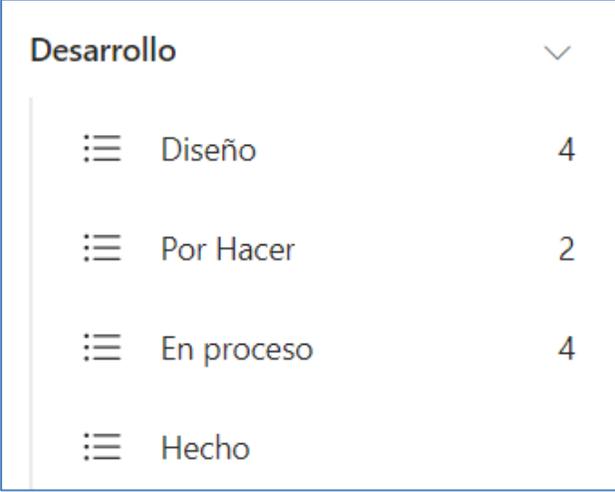
Sección de diseño adicional a la metodología para las interfaces a utilizar en el proyecto

- No pasar a la siguiente fase hasta que se cumpla con el objetivo
- Solo se debe cumplir con lo necesario
- Organizar el proceso como un todo del proyecto
- Confiar en el sistema y su ejecución
- Diseñar los procesos estandarizados y racionalizados
- Tener herramientas efectivas y fiables

#### 4.3. TABLERO

El desarrollo del proyecto se basa en la agilidad a través de la metodología Kanban y controlar el proceso de ejecución. Se utiliza un tablero digital con 4 secciones siguientes:

- Diseño
- Por hacer
- En proceso
- Hecho



Desarrollo		▼
☰	Diseño	4
☰	Por Hacer	2
☰	En proceso	4
☰	Hecho	

Ilustración 4 - Kanban Metodología

Al utilizar las 4 secciones se logra cumplir según cronograma principal en la cual se definen fechas de ejecución y tiempos de entrega.

La elaboración del tablero de control se realizó con la plataforma To Do de Microsoft dando un alcance con mayor control al ofrecer funcionalidades de seguimiento.

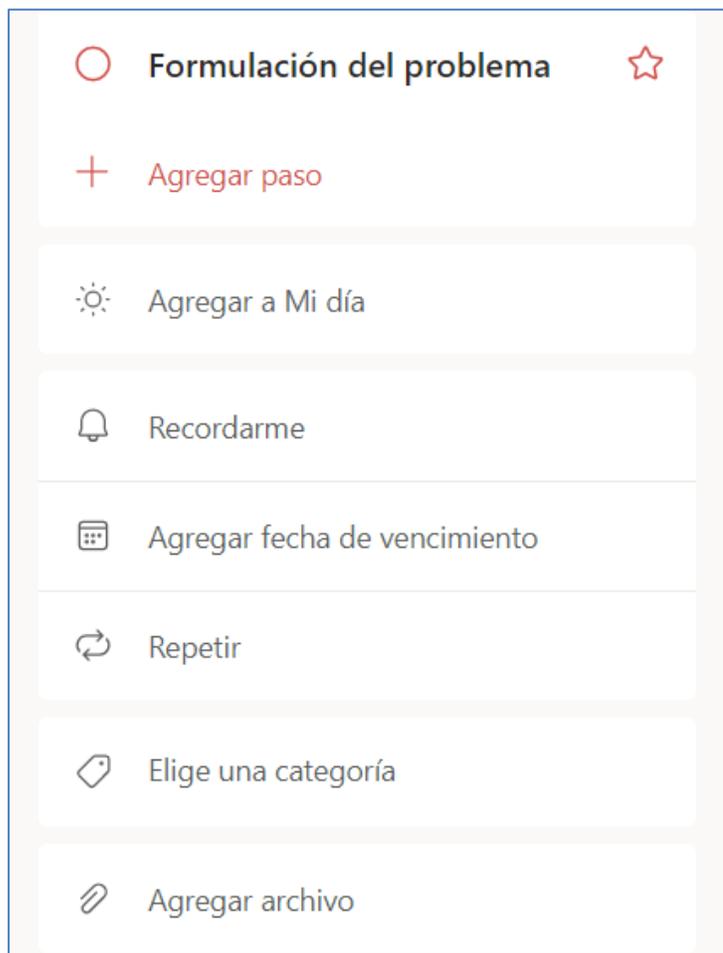


Ilustración 5 - Kanban Tablero

Con esta planeación se puede organizar el día a día y actualizar la lista de manera diaria, semanal y mensual de las tareas pendientes. Adicionalmente, se puede instalar en la aplicación para tener a la mano en cualquier momento las tareas a realizar y que no se exceda de los tiempos establecidos.

## 5. PROCESO DE SOFTWARE

### 5.1. IDENTIFICACIÓN DE STAKEHOLDERS

Tabla 1 - Identificación de Stakeholders

Stakeholder	Descripción
Administrador de finca	Persona que administra el talento humano y da una visión general de los procesos diarios de producción en la finca.
Operadores de ordeño	Equipo de talento humano que está encargado de manejar el equipo de ordeño, adicional alimenta el registro de litros totales diario por bovino.
Ingenieros de desarrollo	Personas encargadas de desarrollar, instalar y proveer soporte técnico al proyecto que se implementará
Comprador lechero	Empresa del sector productivo y transformador encargada en adquirir la producción diaria de leche, validará el total de producción por periodos de tiempo (diario, semanal, mensual) contra los datos recopilados por la aplicación a desarrollar.
Otros actores	Fincas aledañas. Empleados no relacionados al proceso de ordeño
Dueño de finca	Persona que recibe dividendos de la producción de la finca. Tomará decisiones tipo gerenciales de acuerdo a estadísticas arrojadas por el proyecto.

### 5.2. REQUERIMIENTOS

Para definir los requisitos necesarios para el funcionamiento de la aplicación, realizamos un análisis y llegamos a los siguientes requisitos, dividiéndolos en funcionales y no funcionales:

### 5.2.1. REQUERIMIENTOS FUNCIONALES

Tabla 2 - Requerimientos Funcionales

<b>Nº RF</b>	<b>REQUERIMIENTO FUNCIONAL</b>	<b>PRIORIDAD</b>
<b>RF01</b>	La aplicación móvil permitirá registrar al Administrador	<b>ALTA</b>
<b>RF02</b>	La aplicación móvil le permitirá al administrador agregar, modificar o eliminar empleados	<b>ALTA</b>
<b>RF03</b>	La aplicación móvil le permitirá al empleado agregar, modificar o eliminar animales	<b>ALTA</b>
<b>RF04</b>	<p>La aplicación móvil deberá permitir agregar información del animal, identificado por su código único de identificación: -</p> <ul style="list-style-type: none"> <li>● Número único de identificación.</li> <li>● Método de adquisición: Nacida en la finca o comprada.</li> <li>● Fecha de nacimiento.</li> <li>● Sexo.</li> <li>● Raza.</li> <li>● Tipo ganaderia</li> <li>● Peso al nacer o al adquirir</li> </ul>	<b>ALTA</b>
<b>RF05</b>	<p>La aplicación permitirá llevar un registro y control de la producción:</p> <ul style="list-style-type: none"> <li>● Peso rutinario (frecuencia 2 meses).</li> <li>● Producción de leche.</li> <li>● Producción de carne.</li> </ul>	<b>ALTA</b>
<b>RF06</b>	La aplicación generará informes donde se evidenciará el seguimiento por animal.	<b>ALTA</b>
<b>RF07</b>	La aplicación móvil permitirá generar gráficos de análisis para llevar el seguimiento de control y producción bovina.	<b>MEDIA</b>
<b>RF08</b>	La aplicación móvil permitirá guardar la información en una base de datos relacional.	<b>ALTA</b>
<b>RF09</b>	La aplicación móvil permitirá hacer Logout.	<b>MEDIA</b>

Fuente: Propia de los Autores.

A continuación, se describen los casos de uso para la construcción de la APP, éstos fueron elaborados siguiendo requerimientos funcionales.

#### 5.2.1.1. INICIAR SESIÓN

Tabla 3 - CU - 01 - Iniciar Sesión

<b>Identificador</b>	CU - 01
<b>Nombre</b>	Iniciar sesión
<b>Descripción</b>	La aplicación permitirá autenticarse al usuario mediante el envío de las credenciales de este al servidor, el cual se encargará en último momento de la autenticación y responderá a la aplicación afirmativamente si las credenciales son correctas o negativamente si no lo son. La aplicación ha de ser capaz de diferenciar entre los dos casos.
<b>Precondiciones</b>	Que el usuario tenga credenciales (nombre de usuario y contraseña) para iniciar sesión, y que no esté autenticado ya. Debe existir un formulario con los datos de usuario y contraseña.
<b>Postcondiciones</b>	Si los datos son correctos debe continuar con la aplicación en la interfaz principal, si los datos son incorrectos debe mostrar el mensaje de datos incorrectos.
<b>Criterios de Aceptación</b>	<ol style="list-style-type: none"> <li>1. Verificar que la aplicación tenga la opción de inicio de sesión para la autenticación en la plataforma.</li> <li>2. Verificar el acceso a la aplicación con las credenciales correctas y asignación de la vista.</li> <li>3. Verificar los mensajes de inicio de sesión en caso de que las credenciales no sean correctas</li> <li>4. Verificar la conexión en el inicio de sesión</li> <li>5. Verificar el perfil al ingresar a la aplicación</li> </ol>

Fuente: Autores.

## 5.2.1.2. VISUALIZACIÓN MENÚ

Tabla 4 - CU - 02 - Visualización Menú

<b>Identificador</b>	CU- 02
<b>Nombre</b>	Visualización Menú (Administrador - Empleado)
<b>Descripción</b>	La aplicación permitirá mostrar menú de administrador cuando este rol sea autenticado, mostrando las opciones de gestión de empleados y gestión de reportes de producción. Para los roles de empleado, el usuario estará habilitado para ver los animales, rebaños, registros de producción.
<b>Precondiciones</b>	Que el usuario se haya autenticado exitosamente
<b>Postcondiciones</b>	Una vez el ingreso al sistema sea correcto, se debe asignar el rol específico, de no ser asignado el rol correctamente, el sistema saldrá automáticamente.
<b>Criterios de aceptación</b>	<ol style="list-style-type: none"> <li>1. Verificar la asignación correcta del perfil de usuario logueado.</li> <li>2. Verificar el menú del usuario y que corresponda con el asignado.</li> </ol>

Fuente: Propia de los Autores.

## 5.2.1.3. GESTIÓN DE EMPLEADOS

Tabla 5 - CU - 03 - Gestión Empleados

<b>Identificador</b>	CU - 03
<b>Nombre</b>	Gestión de Empleados (Administrador)
<b>Descripción</b>	El sistema debe permitir al administrador hacer la gestión completa de los empleados de la finca.

<b>Precondiciones</b>	El administrador podrá realizar la creación, edición, listado y borrado de cada uno de los empleados de la finca.
<b>Postcondiciones</b>	Luego de la creación tendrá la opción de asignación de finca y animal
<b>Criterios de aceptación</b>	<ol style="list-style-type: none"> <li>1. Verificar que cuente con el menú de gestión de empleados</li> <li>2. Verificar que cuenta con las opciones de edición, visualización y eliminación de los funcionarios.</li> </ol>

Fuente: Propia de los Autores.

#### 5.2.1.4. GESTIÓN DE ANIMALES

Tabla 6 - CU - 06 - Gestión Animales

<b>Identificador</b>	CU - 04
<b>Nombre</b>	Gestión de Animales (Administrador - Empleado)
<b>Descripción</b>	<p>El sistema debe permitir al administrador hacer la gestión completa de los animales en la finca.</p> <p>La aplicación mostrará lista de bovinos activos e inactivos, adicional mostrará la opción de ingresar uno nuevo, dentro de las opciones está: eliminar y actualizar bovino con se seleccione un ítem de la lista.</p> <p>El usuario autenticado tendrá la posibilidad de asignar un animal a un rebaño determinado.</p>
<b>Precondiciones</b>	El administrador podrá realizar la creación, edición, listado y borrado de cada uno de los animales en la finca.
<b>Postcondiciones</b>	Luego de la creación tendrá la opción de asignación de empleado y asignación de Rebaño
<b>Criterios de aceptación</b>	<ol style="list-style-type: none"> <li>1. Verificar que cuente con el menú de gestión de animales.</li> <li>2. Verificar que se pueda visualizar un animal determinado.</li> <li>3. Verificar que se pueda agregar un animal nuevo.</li> </ol>

	<ol style="list-style-type: none"> <li>4. Verificar que se pueda editar un animal.</li> <li>5. Verificar que se pueda eliminar un animal (Administrador)</li> <li>6. Verificar que pueda realizar la asignación de un animal a un rebaño.</li> </ol> <p>Fuente: Propia de los Autores.</p>
--	--

#### 5.2.1.5. GESTIÓN DE REBAÑO

Tabla 7 - CU - 05 - Gestión de Rebaño

<b>Identificador</b>	CU - 05
<b>Nombre</b>	Gestión de Rebaño (Administrador - Empleado)
<b>Descripción</b>	<p>El sistema debe permitir al administrador hacer la gestión completa de los rebaños en la finca (Entiéndase como rebaño al grupo de animales asignados a cierta finca con un determinado número de animales)</p> <p>La asignación de un rebaño se puede hacer a una finca determinada.</p>
<b>Precondiciones</b>	<p>El administrador podrá realizar la creación, edición, listado y borrado de cada uno de los rebaños en la finca.</p> <p>El empleado tendrá las mismas opciones excepto las opciones de eliminar.</p>
<b>Postcondiciones</b>	Luego de la creación tendrá la opción de asignación de animal y asignación de Finca
<b>Criterios de aceptación</b>	<ol style="list-style-type: none"> <li>1. Verificar que cuente con el menú de gestión de rebaño</li> <li>2. Verificar que pueda acceder a visualizar el rebaño y los animales asignados a dicho rebaño.</li> <li>3. Verificar que cuente con la opción de agregar un rebaño.</li> <li>4. Verificar que cuente con la opción de editar un rebaño.</li> </ol>

	<p>5. Verificar que cuenta con la opción de eliminación de los rebaños. (Administrador)</p> <p>6. Verificar la opción de asignar el rebaño a una finca</p>
--	--

Fuente: Propia de los Autores.

#### 5.2.1.6. GESTIÓN DE FINCA

Tabla 8 - CU - 06 - Gestión de Finca

<b>Identificador</b>	CU - 06
<b>Nombre</b>	Gestión de Finca (Administrador)
<b>Descripción</b>	El sistema debe permitir al administrador hacer la gestión completa de la finca.
<b>Precondiciones</b>	El administrador podrá realizar la creación, edición, listado y borrado de cada una de las fincas
<b>Postcondiciones</b>	Luego de la creación tendrá la opción de asignación de rebaño.
<b>Criterios de aceptación</b>	<ol style="list-style-type: none"> <li>1. Verificar que cuente con el menú de gestión de finca</li> <li>2. Verificar que pueda acceder a visualizar una finca.</li> <li>3. Verificar que pueda agregar fincas</li> <li>4. Verificar que pueda Editar Fincas</li> <li>5. Verificar que pueda Eliminar Fincas. (Administrador)</li> <li>6. Verificar la opción de agregar un rebaño a la finca.</li> <li>7. Verificar la asignación de finca al municipio.</li> </ol>

Fuente: Propia de los Autores.

#### 5.2.1.7. GESTIÓN DE MUNICIPIO

Tabla 9 - CU - 07 - Gestión de Municipio

<b>Identificador</b>	CU - 07
----------------------	---------

<b>Nombre</b>	Gestión de Municipio (Administrador)
<b>Descripción</b>	El sistema debe permitir al administrador hacer la gestión completa de los municipios. Esta asignación va destinada al crecimiento de la hacienda y las posibles sedes que pueda tener en las regiones de Colombia.
<b>Precondiciones</b>	El administrador podrá realizar la creación, edición, listado y borrado de cada uno de los municipios asignados a la hacienda
<b>Postcondiciones</b>	Luego de la creación tendrá la opción de asignación de municipio a la hacienda.
<b>Criterios de aceptación</b>	<ol style="list-style-type: none"> <li>1. Verificar que cuente con el menú de gestión de Municipio</li> <li>2. Verificar que el usuario pueda ingresar a ver el municipio.</li> <li>3. Verificar que el usuario pueda asignar un municipio.</li> <li>4. Verificar la opción de asignar la finca a un municipio.</li> </ol>

Fuente: Propia de los Autores.

#### 5.2.1.8. REGISTROS DE PRODUCCIÓN

Tabla 10 - CU - 08 - Registros de Producción

<b>Identificador</b>	CU - 08
<b>Nombre</b>	Registros de producción (Administrador - Empleado)
<b>Descripción</b>	El sistema debe ofrecer una vista donde se agregue cada uno de los registros de producción de leche por animal de acuerdo a su identificador.
<b>Precondiciones</b>	El sistema debe permitir que cada registro sea asignado al animal respectivo
<b>Postcondiciones</b>	Los datos almacenados en los registros podrán ser consultado por el administrador de la finca.
<b>Criterios de aceptación</b>	<ol style="list-style-type: none"> <li>1. Verificar que cuente con el menú de registros de producción</li> </ol>

	<ol style="list-style-type: none"> <li>2. Verificar que el usuario autenticado pueda ingresar al animal y realizar el registro.</li> <li>3. Verificar que el usuario pueda editar el registro.</li> <li>4. Las opciones de eliminar y actualizar únicamente están dispuesta para el administrador de la finca</li> </ol>
--	--

Fuente: Propia de los Autores.

### 5.2.1.9. GESTIÓN DE INFORMES

Tabla 11 - CU - 09 - Gestión de Informes

<b>Identificador</b>	CU - 09
<b>Nombre</b>	Gestión de Informes (Administrador)
<b>Descripción</b>	El sistema debe ofrecer una vista donde el usuario autenticado con perfil administrador pueda ver todos los informes de la granja.
<b>Precondiciones</b>	El sistema debe permitir ver los informes de los elementos de la granja, como son; empleados, registros de producción, rebaños y animales.
<b>Postcondiciones</b>	El administrador podrá descargar estos informes en formatos PDF.
<b>Criterios de aceptación</b>	<ol style="list-style-type: none"> <li>1. Verificar que cuente con el menú de informes</li> <li>2. Verificar que el usuario autenticado pueda ingresar a la opción de informes.</li> <li>3. Verificar que el usuario administrador pueda visualizar informe de empleados de la hacienda</li> <li>4. Verificar que el usuario administrador pueda visualizar el informe de animales de la hacienda.</li> <li>5. Verificar que el usuario administrador pueda visualizar el informe de Fincas de la hacienda.</li> <li>6. Verificar que el usuario administrador pueda visualizar informe de rebaños de la hacienda</li> </ol>

	7. Verificar que el usuario administrador pueda exportar los informes en PDF.
--	---

Fuente: Propia de los Autores.

### 5.2.2. REQUERIMIENTOS NO FUNCIONALES

Dentro de los requerimientos no funcionales para la construcción de la aplicación se obtuvo los siguientes:

Tabla 12 - Requerimientos No Funcionales

<b>N° RNF</b>	<b>REQUERIMIENTO NO FUNCIONAL</b>	<b>PRIORIDAD</b>
<b>RNF01</b>	Los datos ingresados a la aplicación serán sincronizados en línea o cuando se establezca la conexión	<b>ALTA</b>
<b>RNF02</b>	La aplicación debe soportar a lo sumo 20 usuarios concurrentes.	<b>ALTA</b>
<b>RNF03</b>	El acceso a la aplicación será únicamente con usuarios creados previamente.	<b>ALTA</b>
<b>RNF04</b>	El administrador es el único usuario con todos los privilegios.	<b>ALTA</b>
<b>RNF05</b>	La aplicación debe estar disponible 24/7 debido a que la toma de registros se realiza en horas de la madrugada.	<b>ALTA</b>
<b>RNF06</b>	Se debe guardar log de acceso de inicio de sesión, registros y cambios para auditoría.	<b>ALTA</b>
<b>RNF07</b>	Se deben guardar los mensajes de error que pueda presentar la aplicación	<b>ALTA</b>

Fuente: Propia de los Autores.

### 5.3. DISEÑO Y ARQUITECTURA

Para definir los requisitos necesarios para el funcionamiento de la aplicación, realizamos un análisis y llegamos a los siguientes requisitos, dividiéndolos en funcionales y no funcionales.

Se presentan a continuación diferentes vistas de los detalles de arquitectura relevantes de la aplicación. Los modelos presentados tienen como fin mostrar la información del programa, su despliegue, la comunicación y dejar claridad de la funcionalidad completa de la aplicación.

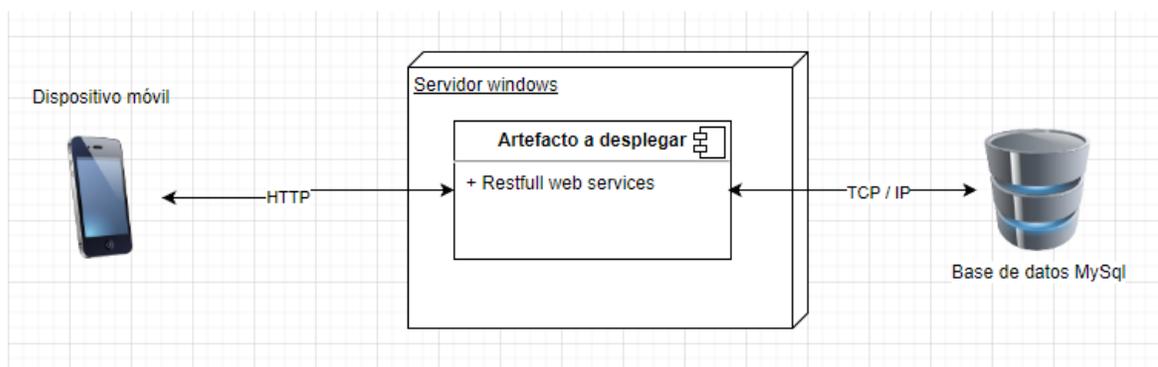


Ilustración 6 – Diseño y arquitectura básico

#### 5.3.1. DIAGRAMA DE COMPONENTES

En este diagrama se modela la relación entre los componentes físicos y lógicos de la aplicación. Los componentes utilizados se hicieron a través de las clases e interfaces del sistema divididos en una capa de presentación (Front End), una capa de lógica de negocio (Back-end) y esta última interactúa con la base de datos.

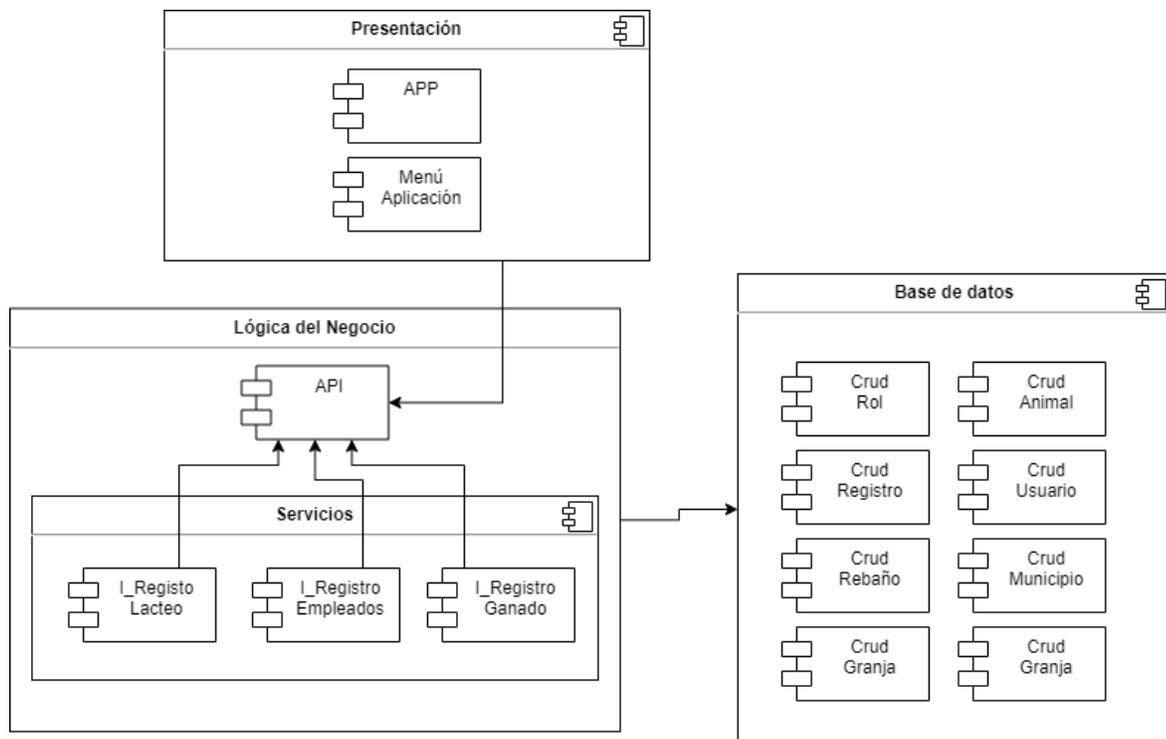


Ilustración 7 - Diagrama de Componentes

Una vista del componente simplificado, sin embargo, en la arquitectura de alto nivel se especificará un poco más detallado.

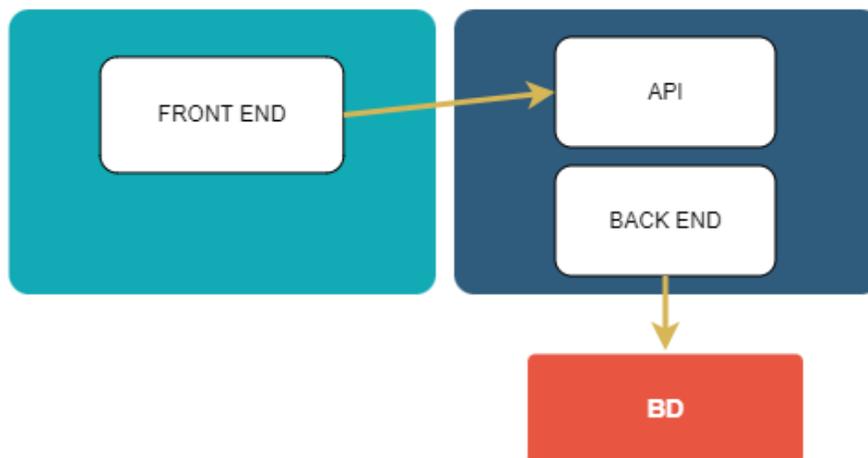


Ilustración 8 - Componentes alto nivel

### 5.3.2. DIAGRAMA DE CASOS DE USO

El diagrama de casos de uso refleja dos actores principales en el sistema y las posibles acciones que puede hacer cada uno de ellos. Se basó en dos subsistemas donde la fuente principal de acción está reflejada en la lista de los animales. El administrador tiene todas las opciones del empleado y el empleado tendrá unas opciones determinadas. Para lograr esta independencia el administrador tiene una generalización sobre el empleado lo cual le permite agregar todas las opciones que tiene el empleado y el empleado no puede acceder a todas las opciones que tiene el administrador. El diagrama de casos de uso expuesto se expresa por medio de operaciones generales de listado de opciones (municipio, finca, rebaño, empleado, animales) y para realizar las operaciones de acción es necesario la dependencia del listado general de cada una de ellas y a su vez se extienden a dichas operaciones.

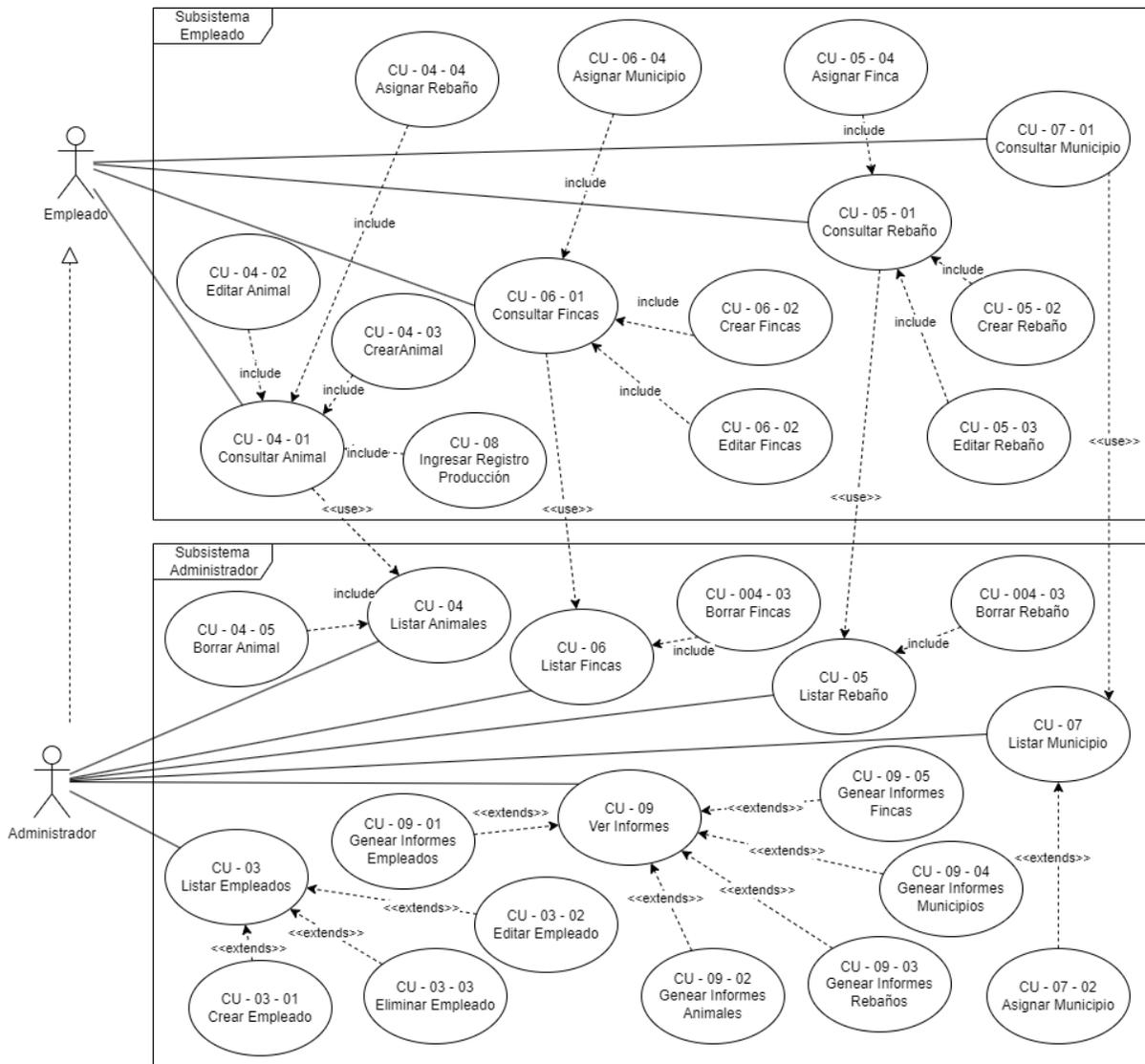


Ilustración 9 - Diagrama de casos de uso

### 5.3.3. DIAGRAMA DE CLASES

El diagrama de clases expuesto describe la estructura del proyecto y las funcionalidades en concreto. Este diagrama nos ayuda a comprender mejor el esquema de la aplicación y nos muestra las necesidades a solucionar con el software. El diagrama

mostrará los modelos de datos que serán utilizados. A continuación, el diagrama de clases utilizado.

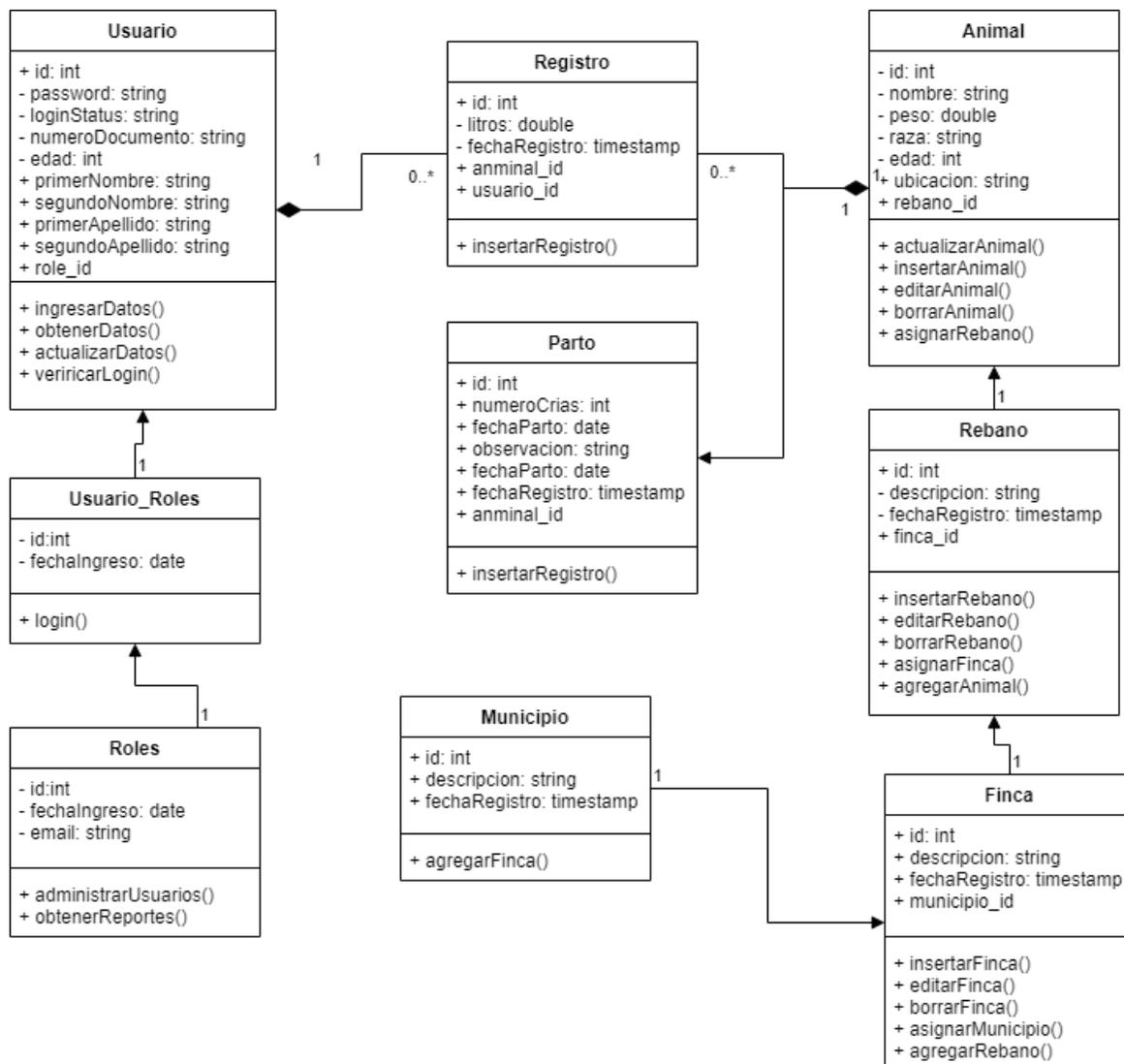


Ilustración 10 - Diagrama de clases

### 5.3.4. DIAGRAMA DE ESTADOS

Para entender el comportamiento del sistema, se elabora un diagrama de estados con la mayor precisión posible. En este caso es posible identificar que sucede en un

momento determinado en la aplicación y cuáles son las opciones o caminos para seguir en el punto específico. A continuación, el diagrama dispuesto.



Ilustración 11 - Diagrama de estados

### 5.3.5. DIAGRAMA DE COLABORACIÓN

Es importante tener en cuenta los actores y roles que se tendrán en el sistema y las determinaciones interacciones. En un diagrama básico se puede reflejar las acciones o comportamientos principales del sistema. A continuación, el diagrama de colaboración dispuesto para el sistema.

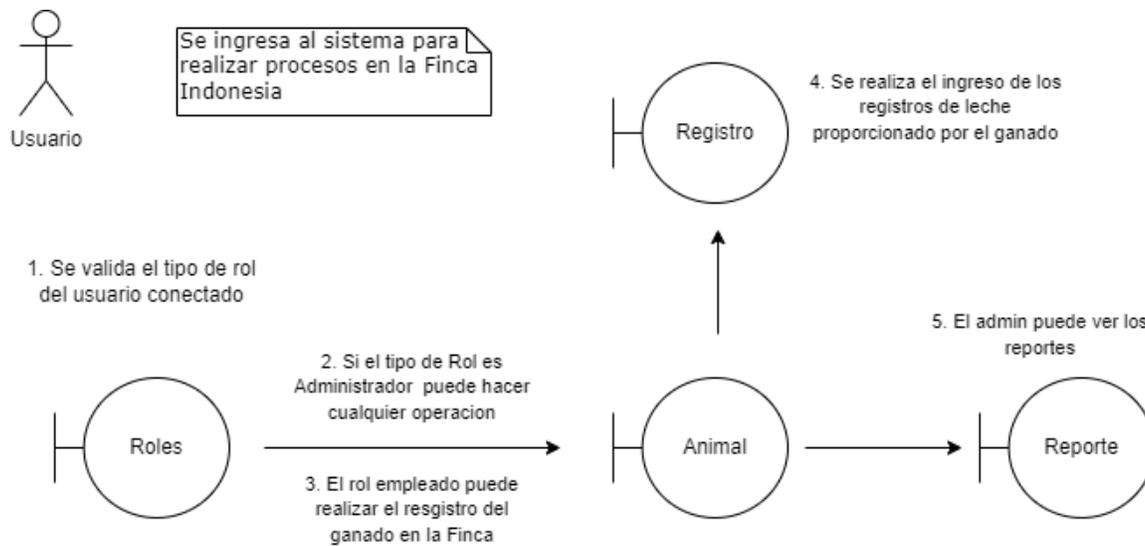


Ilustración 12 - Diagrama de Colaboración

**5.3.6. DIAGRAMA DE SECUENCIA**

Para ver los procesos y los objetivos de la aplicación, se muestra el diagrama de secuencia que muestra la línea de vida con el objetivo y los mensajes dispuestos en la ejecución de las tareas.

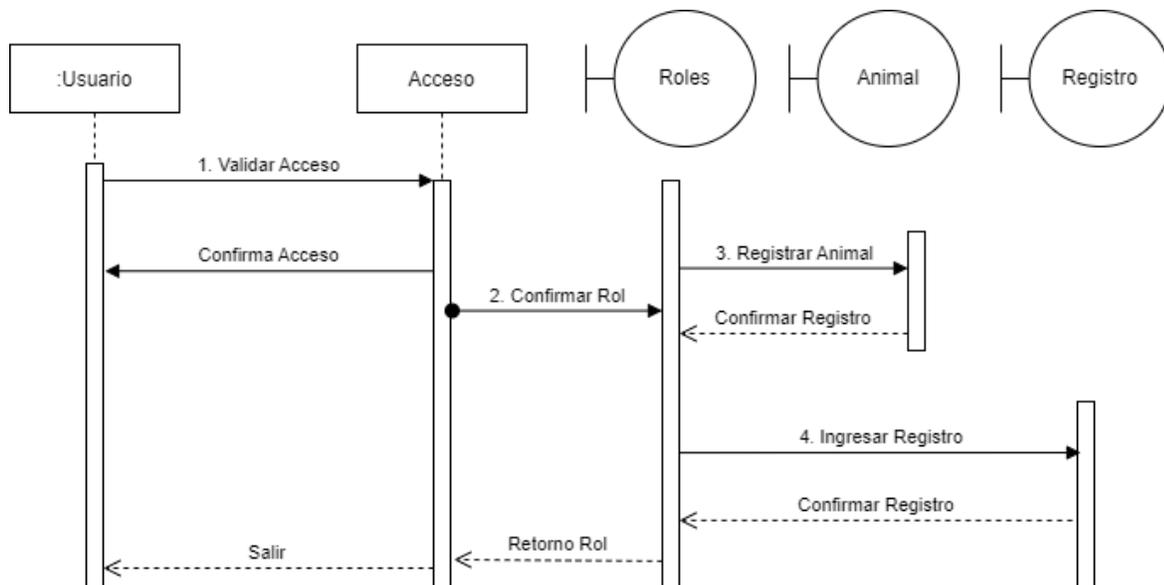


Ilustración 13 - Diagrama de secuencia

#### 5.4. ARQUITECTURA DE ALTO NIVEL

Para desarrollar este proyecto se implementaron diversas tecnologías, las cuales fueron adoptadas y ayudaron a solucionar este problema. Con esto en mente, la implementación de la infraestructura en la nube de AWS se identificó como un piloto de análisis de costos para respaldar y entregar el desarrollo de aplicaciones.

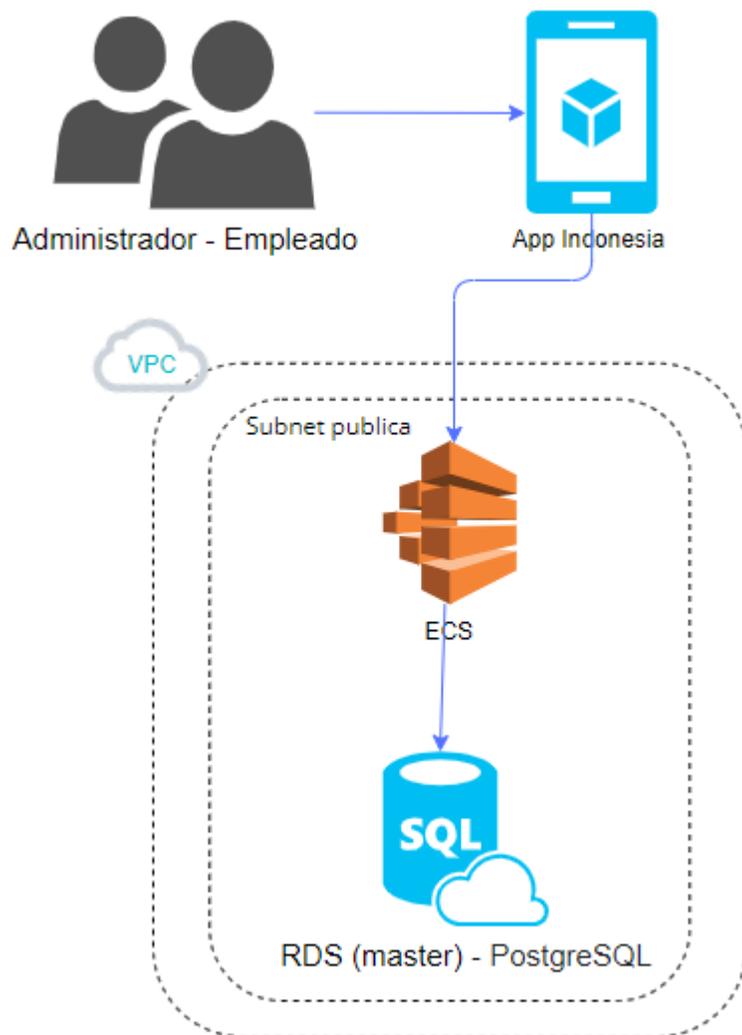


Ilustración 14 - Arquitectura de alto nivel

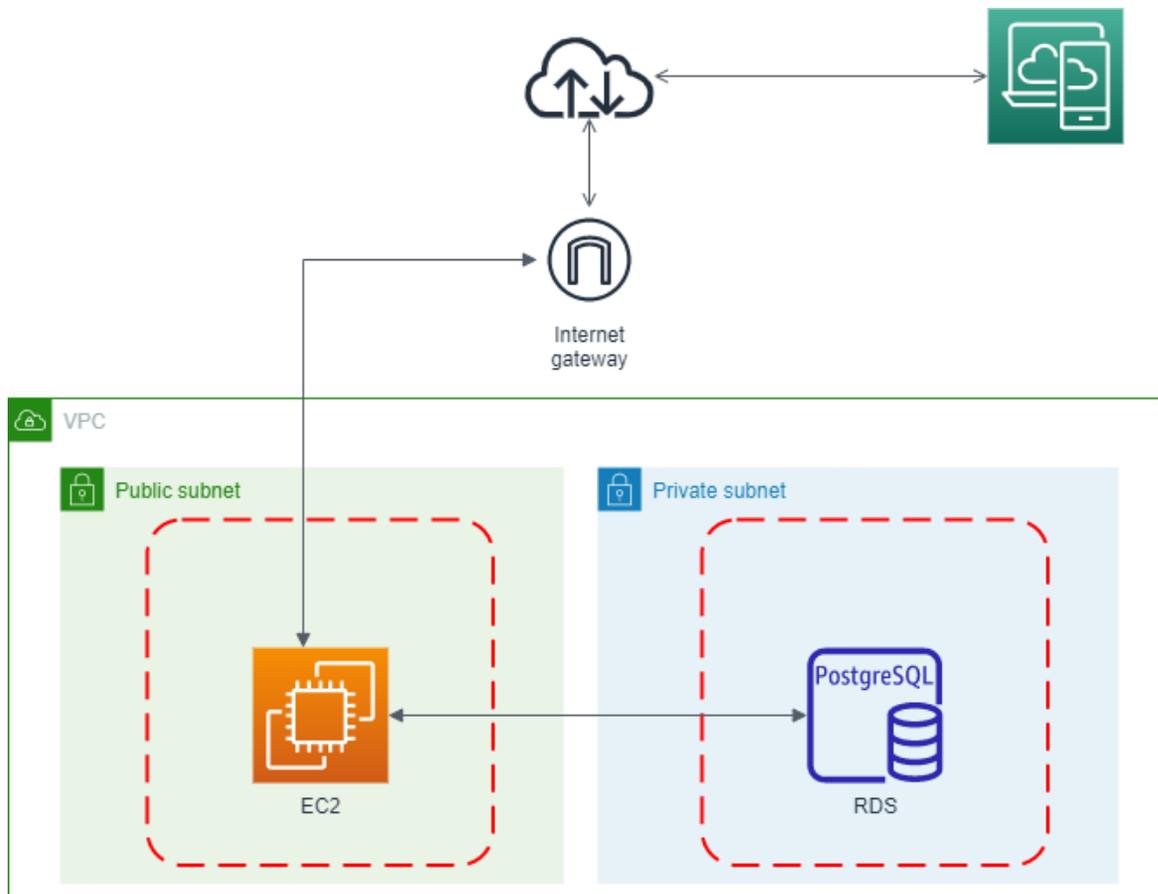


Ilustración 15 - Arquitectura Alto Nivel AWS

## 6. CONSTRUCCIÓN

Para realizar el desarrollo del proyecto, se configuran previamente las herramientas, creación de máquinas y distintos componentes necesarios para ejecutar el proyecto. Dado que el desarrollo está planteado como capas, se realiza la división en dos subproyectos denominados BackEnd y FrontEnd, a continuación, se describe una etapa de preparación.

## 6.1. INSTALACIÓN Y CONFIGURACIÓN

La construcción de software fue realizada con las herramientas AWS de Amazon.

Se utilizó un recurso de Amazon EC2 y RDS.

Para la instancia EC2 se utilizó una versión de la capa gratuita de tipo t2.micro, la cual viene cubierta por las siguientes características:

- Plataforma: Ubuntu – Linux
- Volumen: 20 Gb

**Resumen de instancia de i-08a7472d076aff084 (BackEnd)** Información

Se ha actualizado hace less than a minute

Conectar Estado de la instancia Acciones

ID de la instancia i-08a7472d076aff084 (BackEnd)	Dirección IPv4 pública 35.172.179.106   <a href="#">dirección abierta</a>	Direcciones IPv4 privadas 172.31.21.238
Dirección IPv6 -	Estado de la instancia <span style="color: green;">✔ En ejecución</span>	DNS de IPv4 pública ec2-35-172-179-106.compute-1.amazonaws.com   <a href="#">dirección abierta</a>
Tipo de nombre de anfitrión Nombre de IP: ip-172-31-21-238.ec2.internal	Nombre DNS de IP privada (solo IPv4) ip-172-31-21-238.ec2.internal	Direcciones IP elásticas -
Responder al nombre DNS de recurso privado IPv4 (A)	Tipo de instancia t2.micro	Hallazgo de AWS Compute Optimizer <a href="#">Suscribirse a AWS Compute Optimizer para recibir recomendaciones.</a> <a href="#">Más información</a>
Dirección IP asignada automáticamente 35.172.179.106 [IP pública]	ID de VPC vpc-0dbcaab3323279815	Nombre del grupo de Auto Scaling -
Rol de IAM -	ID de subred subnet-0e3a19581d441f081	

Ilustración 16 - Instancia Back-End

Para el acceso a la máquina virtual se utiliza una llave de seguridad que se originó durante la creación del servidor, adicionalmente en el mismo grupo de seguridad se le dio acceso a los puertos de conexión segura para realizarla a través de SSH.

En cuanto a la RDS se utilizó un tipo de motor PostgreSQL y un tamaño de db.t3.micro bajo la instancia de base de datos gratuita. la base de datos para mejorar el nivel de seguridad esta instancia únicamente es accesible por el servidor backend

The screenshot shows the AWS RDS console for the instance 'db-app-test'. The breadcrumb navigation is 'RDS > Databases > db-app-test'. There are buttons for 'Modificar' and 'Acciones'. Below is a 'Resumen' section with the following details:

Identificador de base de datos	CPU	Estado	Clase
db-app-test	3.37%	Disponible	db.t3.micro
Rol	Actividad actual	Motor	Región y AZ
Instancia	10 Conexiones	PostgreSQL	us-east-1f

### Ilustración 17 - Instancia RDS

Para ambas instancias se creó un grupo de seguridad determinado por permisos a través de puertos. El grupo de seguridad será compartido para la etapa de pruebas y desarrollo, una vez terminado el desarrollo es posible ajustar la aplicación a las necesidades del negocio. A continuación, se muestran las reglas básicas de las instancias EC2 y RDS.

Tipo	Protocolo	Intervalo de puertos	Origen	Descripción
TCP personalizado	TCP	8080	0.0.0.0/0	Acceso 8080
HTTPS	TCP	443	0.0.0.0/0	Acceso HTTPS
PostgreSQL	TCP	5432	0.0.0.0/0	Postgres ANY
SSH	TCP	22	0.0.0.0/0	Acceso SSH
HTTP	TCP	80	0.0.0.0/0	Acceso 80

### Ilustración 18 - Grupo Seguridad Instancias

Otro aspecto a tener en cuenta en la configuración del proyecto para el desarrollo y pruebas fueron las redes. Se realizó la parametrización de un VPC (Amazon Virtual Private Cloud) en la cual se asignan las direcciones IP dentro de la misma VPC.

Detalles <a href="#">Información</a>			
ID de la VPC vpc-0dbcaab3323279815	Estado Available	Nombres de host DNS Habilitado	Resolución de DNS Habilitado
Tenencia Default	Conjunto de opciones de DHCP dopt-0038ce7a469b2563e	Tabla de enrutamiento principal rtb-065a3499b226879b9	ACL de red principal acl-08231ef11c3ecc59b
VPC predeterminada Sí	CIDR IPv4 172.31.0.0/16	Grupo IPv6 -	CIDR IPv6 (grupo de bordes de red) -
Métricas de unidad de mapeo de red Desactivado	Grupos de reglas del firewall de DNS de Route 53 Resolver -	ID de propietario 925766377217	

Ilustración 19 - Instancias VPC

## 6.2. BACK-END

El enfoque dado al desarrollo consiste en una arquitectura por capas, se prepara un Back-End construido en Java Spring Boot 2.7.4. La arquitectura de proyecto se muestra en la siguiente figura.



Ilustración 20 - Arquitectura Back-End

El proyecto tiene unas capas determinadas las cuales son; capa de restController, capa de implementación de interfaz, capa de dominio, cada de entidades o datos y capa de base de datos. A continuación, se muestra una estructura breve de cada una de ellas:

- Capa RestController



Ilustración 21 - Back-End - Capa RestController

- Capa de base de datos



Ilustración 22 - Back-End - Capa Base de datos

- Capa implementación de interfaz

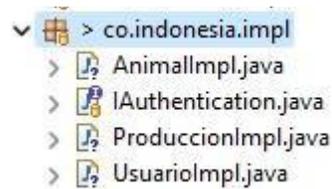


Ilustración 23 - Back-End - Capa Interfaz

Con las capas organizadas se expone un API Rest para el consumo de los servicios los cuales son consumidos por el FRONT-END.

### 6.3. FRONT-END

Para la capa de presentación se utiliza Java y se consumen los servicios expuestos por el Back-End. Se utiliza la herramienta de Android Studio. La herramienta más utilizada para la elaboración de este tipo de proyectos y compatible con Java y Kotlin.

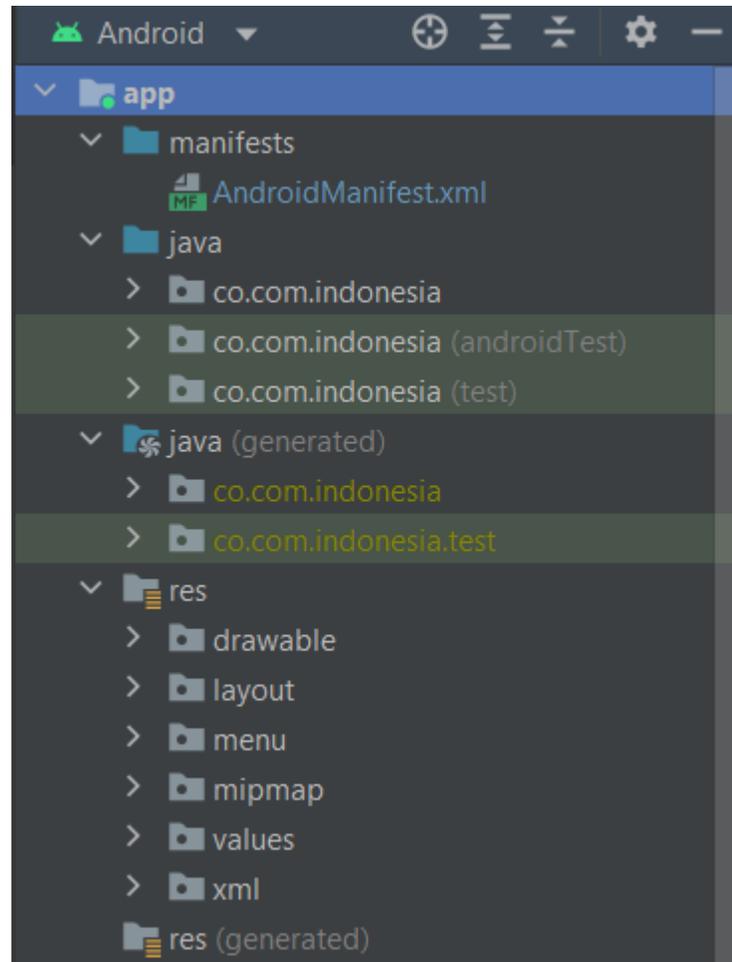


Ilustración 24 - Arquitectura Front-End

Para el desarrollo de la aplicación Front se utilizan los siguientes complementos, los cuales son pilares para la adopción de la aplicación:

```
dependencies {  
    implementation 'androidx.appcompat:appcompat:1.5.1'  
    implementation 'com.google.android.material:material:1.4.0'  
    implementation 'androidx.constraintlayout:constraintlayout:2.1.4'  
    testImplementation 'junit:junit:4.13.2'  
    androidTestImplementation 'androidx.test.ext:junit:1.1.3'  
    androidTestImplementation 'androidx.test.espresso:espresso-core:3.4.0'  
  
    /* Retrofit */  
    implementation 'com.squareup.retrofit2:retrofit:2.9.0'  
    implementation 'com.squareup.retrofit2:converter-gson:2.9.0'  
    implementation 'com.squareup.okhttp3:logging-interceptor:4.10.0'  
}
```

Ilustración 25 - Front-End – Complementos

#### 6.4. PRINCIPIOS DE SEGURIDAD

Aplicando los principios de seguridad, se utiliza seguridad de Token con JWT con una duración de 15 minutos y acceso por roles. Se estipula usuario y contraseña para el acceso a la aplicación y al generarse un token debe ser utilizado para el resto de las operaciones dentro de la aplicación. A continuación, un breve detalle de la operación de autenticación.

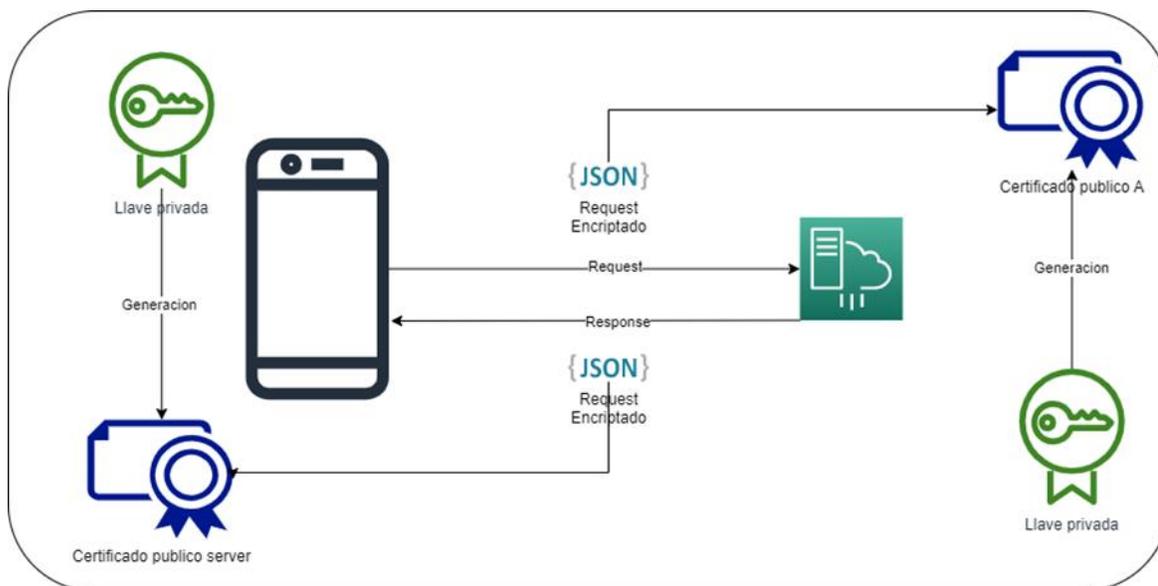


Ilustración 26 - Seguridad - Conexión

Adicional de agregar un intercambio de llaves la cual tiene como propósito encriptar las tramas (json) de petición y de respuesta entre la aplicación y las api's.

## 7. APLICACIÓN

A continuación, se muestran pantallas de la aplicación desarrollada. Cada una de las secciones están mostradas dando claridad de su funcionamiento.



Ilustración 27 - App Presentación



Ilustración 28 - App Inicio Sesión



Ilustración 29 - App Menú Principal

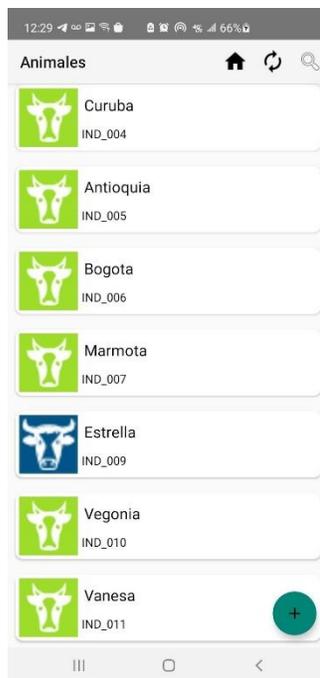
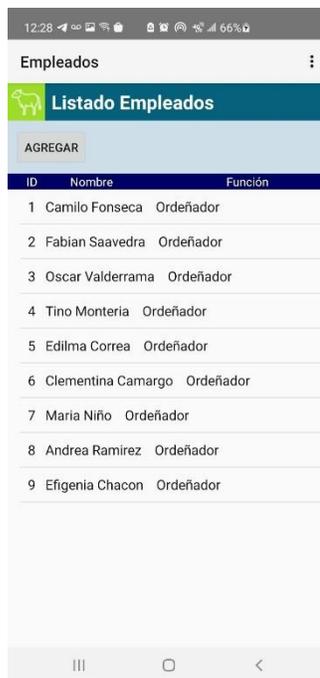


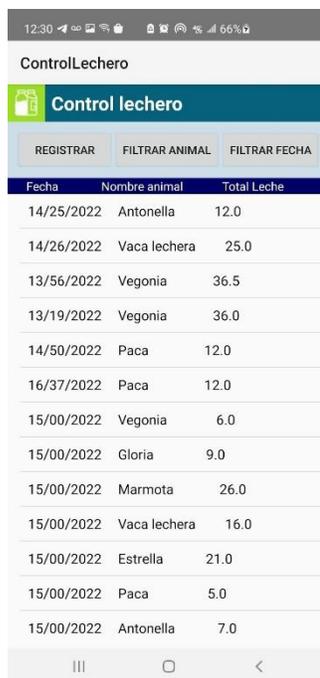
Ilustración 30 - App Listado Animales



The screenshot shows a mobile application interface for 'Empleados'. At the top, there is a status bar with the time 12:28 and battery level at 66%. Below the status bar, the app title 'Empleados' is displayed. A green header bar contains a cow icon and the text 'Listado Empleados'. Below this, there is a button labeled 'AGREGAR'. The main content is a table with three columns: 'ID', 'Nombre', and 'Función'. The table lists nine employees, all with the function 'Ordeñador'.

ID	Nombre	Función
1	Camilo Fonseca	Ordeñador
2	Fabian Saavedra	Ordeñador
3	Oscar Valderrama	Ordeñador
4	Tino Montería	Ordeñador
5	Edilma Correa	Ordeñador
6	Clementina Camargo	Ordeñador
7	Maria Niño	Ordeñador
8	Andrea Ramirez	Ordeñador
9	Efigenia Chacon	Ordeñador

Ilustración 31 - App Listado Empleados



The screenshot shows a mobile application interface for 'ControlLechero'. At the top, there is a status bar with the time 12:30 and battery level at 66%. Below the status bar, the app title 'ControlLechero' is displayed. A green header bar contains a cow icon and the text 'Control lechero'. Below this, there are three buttons: 'REGISTRAR', 'FILTRAR ANIMAL', and 'FILTRAR FECHA'. The main content is a table with three columns: 'Fecha', 'Nombre animal', and 'Total Leche'. The table lists 15 records of milk production for various animals over time.

Fecha	Nombre animal	Total Leche
14/25/2022	Antonella	12.0
14/26/2022	Vaca lechera	25.0
13/56/2022	Vegonia	36.5
13/19/2022	Vegonia	36.0
14/50/2022	Paca	12.0
16/37/2022	Paca	12.0
15/00/2022	Vegonia	6.0
15/00/2022	Gloria	9.0
15/00/2022	Marmota	26.0
15/00/2022	Vaca lechera	16.0
15/00/2022	Estrella	21.0
15/00/2022	Paca	5.0
15/00/2022	Antonella	7.0

Ilustración 32 - App Listado Control Lácteo



Ilustración 33 - App Informes

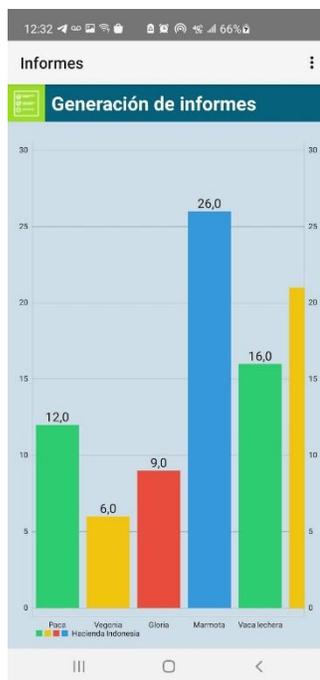


Ilustración 34 - App Reportes



Ilustración 35 - App Filtro Fecha

## 8. PRUEBAS

Para las métricas de calidad del software, como fallas, errores, code smells, cobertura y duplicación de código, el backend desarrollado con Spring Boot utiliza esta vez la herramienta Sonarcloud, que se conecta a un repositorio local con la capacidad para escanear código en nuestro escenario de calidad dándole mantenimiento con sus capacidades de análisis y pruebas. Se creó un proyecto en la administración de sonarqube como lo demuestra la Ilustración N ZZ donde se incrusta el repositorio backend que evaluaremos. Luego como lo demuestra la IlustracionNZZ ejecutamos el comando de análisis de proyecto en lenguaje java

```

INFO: CPD Executor Calculating CPD for 26 files
INFO: CPD Executor CPD calculation finished (done) | time=70ms
INFO: Analysis report generated in 258ms, dir size=232.4 kB
INFO: Analysis report compressed in 785ms, zip size=111.7 kB
INFO: Analysis report uploaded in 62ms
INFO: ANALYSIS SUCCESSFUL, you can find the results at: http://localhost:9000/dashboard?id=indonesia-backend
INFO: Note that you will be able to access the updated dashboard once the server has processed the submitted analysis report
INFO: More about the report processing at http://localhost:9000/api/ce/task?id=AVQB13LrSeY91w1SNsNE
INFO: Analysis total time: 15.119 s
INFO: -----
INFO: EXECUTION SUCCESS
INFO: -----
INFO: Total time: 17.600s
INFO: Final Memory: 21M/77M
INFO: -----
C:\Users\User\Documents\UAN\indonesia\app\IndonesiaApp\indonesia-backend>

```

Ilustración 36 - Sonar

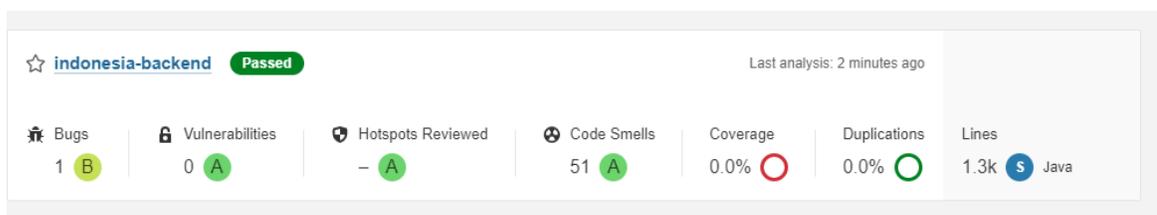


Ilustración 37 - Sonar 2

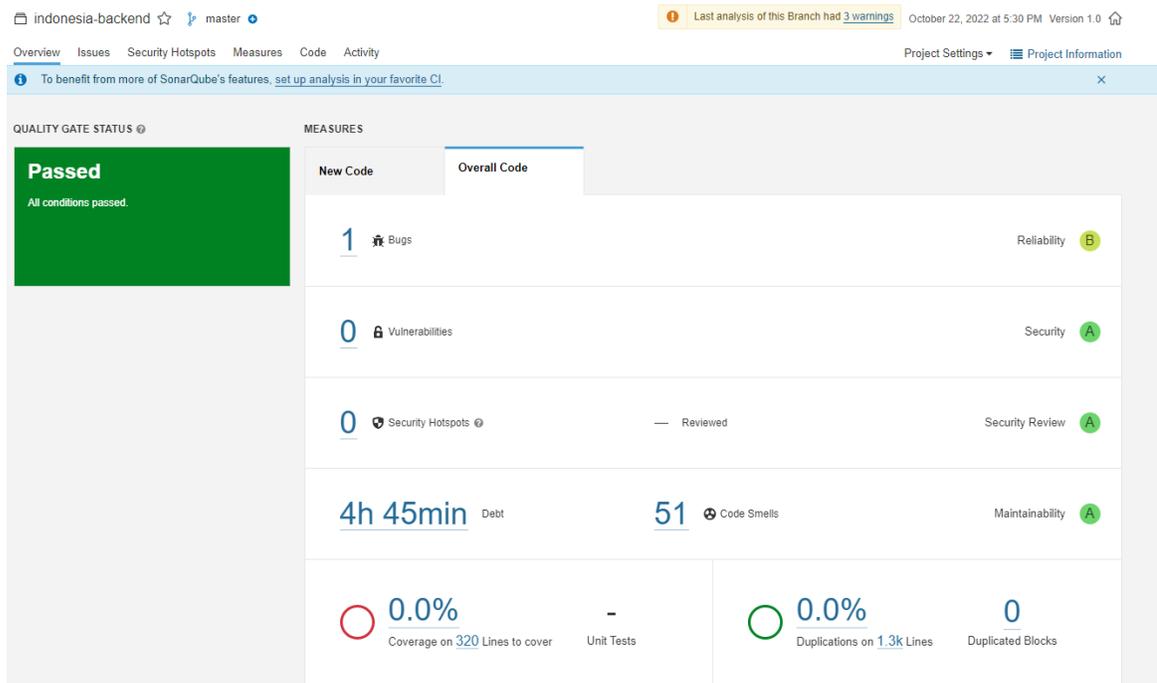


Ilustración 38 - Pruebas de calidad en Backend - Spring Boot

**Fuente: Sonarcloud**

Para concluir navegamos al proyecto previamente creado sitio local del sonarqube donde observamos los resultados del código estático tal como lo muestra la IlustraciónZZ..

## 9. CONCLUSIONES

Se diseña y desarrolla el proyecto exitosamente, cumpliendo los objetivos planteados y llevando un componente de innovación a un campo que lo necesita. Se desarrolla y pone en marcha un aplicativo móvil para plataforma Android, cuyo objetivo es realizar la gestión de animales, una gestión de producción y de empleados. Dentro de esos objetivos se puede concluir que se cumplió lo siguiente:

- La pérdida de registros es casi nula y se cumple con un sistema que permite ver en tiempo real los datos que se están tomando de los registros de producción de leche.
- La información centralizada de la producción de leche, animales y empleados.
- El dueño o administrador de la hacienda tiene información suficiente para la toma de decisiones.
- Es posible guardar y tener a su disposición PDF con la información relevante y/o descargarlos en cualquier momento. Al igual que un gráfico de barras con el total de producción diaria y otro gráfico generado por la producción individual de cada animal.

**ANEXOS**

## REFERENCIAS BIBLIOGRÁFICAS

Android Studio dolphin. (2022, julio 10). Android Developers.

<https://developer.android.com/studio/releases/index.html>

Benítez, L. F. (2015). Concepto De La Aplicación Móvil - Ligo Gratis.

Eumed.net. <https://www.eumed.net/libros-gratis/2016/1539/aplicacion.htm>

Boje, J. (2017). Spring boot: How to get started and build a microservice - second edition. Createspace Independent Publishing Platform.

BovControl - cattle management. (s/f). Google.com. Recuperado el 22 de octubre de 2022, de

<https://play.google.com/store/apps/details?id=com.bovcontrol.bovcontrol>

Bovino. (s/f). Rae.es. Recuperado el 22 de octubre de 2022, de

<https://dle.rae.es/bovino>

BovinoApp. (s/f). Google.com. Recuperado el 22 de octubre de 2022, de

<https://play.google.com/store/apps/details?id=com.bovinoapp>

Control Ganadero. (s/f). Google.com. Recuperado el 22 de octubre de 2022, de

[https://play.google.com/store/apps/details?id=com.grupoarve.cganadero&hl=es\\_CO&gl=US](https://play.google.com/store/apps/details?id=com.grupoarve.cganadero&hl=es_CO&gl=US)

de Planeación y Estudios Económicos, O. (s/f). Balance y perspectivas del sector ganadero colombiano 2021 -2022. [estadisticas.fedegan.org.co](http://estadisticas.fedegan.org.co). Recuperado el 22 de octubre de 2022, de

[https://estadisticas.fedegan.org.co/DOC/download.jsp?pRealName=Balance\\_Perspectivas\\_ganaderia\\_colombiana\\_2021\\_2022\\_.pdf&iIdFiles=817](https://estadisticas.fedegan.org.co/DOC/download.jsp?pRealName=Balance_Perspectivas_ganaderia_colombiana_2021_2022_.pdf&iIdFiles=817)

Informática Básica: ¿Qué es una aplicación móvil? (s/f). Gcfglobal.org.

Recuperado el 22 de octubre de 2022, de

<https://edu.gcfglobal.org/es/informatica-basica/que-es-una-aplicacion-movil/1/>

Jones, M., Bradley, J., & Sakimura, N. (2015). JSON Web Token (JWT). RFC Editor.

Ministerio de comercio, industria y turismo, Información: Perfiles Económicos

Departamentales , Oficina de Estudios Económicos Septiembre 2022.

(2022, septiembre). Gov.co.

<https://www.mincit.gov.co/getattachment/estudios-economicos/perfiles-economicos-por-departamentos/perfiles-economicos-por-departamentos/boyaca/oee-yc-perfil-departamental-boyaca-29sep22.pdf.aspx>

Néstor, E., Daza, C., Secretario, T., & Consejo, N. (s/f). OBSERVATORIO DEL SECTOR LÁCTEO COLOMBIANO. Gov.co. Recuperado el 22 de octubre de 2022, de

[http://uspleche.minagricultura.gov.co/assets/boletin\\_economico\\_sector\\_lacteo\\_2022.pdf](http://uspleche.minagricultura.gov.co/assets/boletin_economico_sector_lacteo_2022.pdf)

PostgreSQL. (2022, octubre 22). PostgreSQL. <https://www.postgresql.org/>

SQLite Home Page. (s/f). Sqlite.org. Recuperado el 22 de octubre de 2022, de

<https://www.sqlite.org/index.html>

Wikipedia contributors. (s/f). Eclipse (software). Wikipedia, The Free Encyclopedia.

[https://es.wikipedia.org/w/index.php?title=Eclipse\\_\(software\)&oldid=145660543](https://es.wikipedia.org/w/index.php?title=Eclipse_(software)&oldid=145660543)

Federación Colombiana de Ganaderos (Fedegan). Balance y perspectivas del sector ganadero colombiano 2021 – 2022.

[https://estadisticas.fedegan.org.co/DOC/download.jsp?pRealName=Balance\\_Perspectivas\\_ganaderia\\_colombiana\\_2021\\_2022\\_.pdf&iIdFiles=817](https://estadisticas.fedegan.org.co/DOC/download.jsp?pRealName=Balance_Perspectivas_ganaderia_colombiana_2021_2022_.pdf&iIdFiles=817)