

**Sistema de Información para Seguimiento y Control de los Procesos  
Administrativos de la Asociación Asoagroproleche del Municipio de Carmen de Carupa  
AsoVapp**

Yeimy Constanza Ángel Sánchez y Jairo Orlando Pascagaza Barriga

Programa de ingeniería de sistemas, Universidad de Cundinamarca

Trabajo de grado

Director

Cristian Eduardo Cano López

Junio 2, 2023

## **Dedicatoria**

Dedicamos este proyecto a nuestras familias quienes con su presencia, amor y guía han sido nuestra inspiración y fortaleza a lo largo de este proceso. Agradecemos porque ellos fueron nuestros primeros maestros de vida que con su amor, dedicación y sacrificio han sido un faro que nos ha guiado en cada momento. Gracias por enseñarnos el valor del trabajo duro, la honestidad y la perseverancia. Han sido un apoyo incondicional en cada paso que hemos dado, creyendo en nuestros sueños y animándonos a alcanzarlos.

A esos seres que ya no nos acompañan en la vida terrenal pero que siempre los llevaremos en el corazón y nos inspiraron a cumplir esta etapa de nuestras vidas *15*.

## **Agradecimiento**

En primer lugar, damos gracias a Dios por permitirnos culminar esta meta.

Queremos agradecer a nuestros padres, quienes han sido nuestra mayor fuente de inspiración y apoyo incondicional. Su amor, paciencia, consejos y sacrificios han sido fundamentales en nuestras vidas. Gracias por creer en nosotros y por alentarnos a seguir nuestros sueños. Su constante estímulo y confianza en nuestras habilidades han sido la fuerza e impulso detrás de este logro.

Queremos expresar nuestro sincero agradecimiento a todas las personas que han contribuido de alguna u otra manera en la realización de este proyecto. Su apoyo, orientación y colaboración han sido fundamentales para la culminación de este.

Además, agradecemos a nuestros compañeros y amigos que nos han brindado su apoyo. Sus palabras de aliento y colaboración han sido clave para superar los obstáculos y mantenernos enfocados en la meta.

## Índice

1.	Introducción .....	11
2.	Resumen.....	13
3.	Definición del problema .....	15
3.1.	Descripción de la situación problemática .....	15
3.1.1.	Formulación del problema .....	17
4.	Justificación .....	18
5.	Objetivos .....	20
5.1.	Objetivo general.....	20
5.2.	Objetivos específicos .....	20
6.	Alcances y limitaciones .....	21
6.1.	Alcances .....	21
6.2.	Limitaciones.....	22
7.	Marco de referencia .....	23
7.1.	Estado del arte.....	23
7.1.1.	Internacional .....	23
7.1.2.	Nacional .....	24
7.1.3.	Local .....	25
7.2.	Marco teórico .....	25
7.2.1.	Sistemas de información.....	25
7.2.2	Sistema de información gerencial .....	26

7.2.2.	Sistemas de planificación de recursos (ERP).....	27
7.2.3.	Teoría general de sistemas .....	28
7.2.4.	Software .....	28
7.2.5.	Desarrollo ágil.....	29
7.2.6.	ISO/IEC 25000.....	30
7.3.	Marco conceptual.....	30
7.3.1.	Base De Datos.....	30
7.3.2.	Sistema Gestor De Bases De Datos .....	31
7.3.3.	Modelo Vista Controlador .....	32
7.3.4.	Lenguaje SQL.....	32
7.3.5.	Lenguaje De Programación.....	32
7.3.6.	TIC .....	33
7.4.	Marco Legal .....	34
8.	Marco metodológico .....	37
8.1.	Metodología de trabajo en equipo.....	37
8.1.1.	Product Backlog.....	38
8.1.2.	Sprint Planning.....	39
8.1.3.	Diagrama de burn down.....	43
8.2.	Metodología de investigación.....	73
8.3.	Ingeniería del proyecto .....	77
8.3.1.	Requerimientos Funcionales.....	77
8.3.2.	Requerimientos No Funcionales .....	88

8.3.3. Diagramas de UML del sistema.....	93
8.4. Metodología de desarrollo .....	97
8.4.1. Diseño de Logo .....	98
8.4.2. Diseño de mockups .....	100
9. Pruebas.....	102
9.1. Pruebas unitarias .....	102
9.1.1. Módulo inicio de sesión .....	102
9.2. Pruebas de integración .....	103
9.2.1. Módulo de presidente.....	104
9.2.2. Módulo de administradora .....	106
9.2.3. Módulo de socios .....	109
9.3. Pruebas de rendimiento.....	111
9.4. Pruebas de aceptación .....	118
10. Cronograma.....	120

## Índice de tablas

<b>Tabla 1</b> Asignación de roles .....	37
<b>Tabla 2</b> Product Backlog .....	39
<b>Tabla 3</b> Actividades que se planean en reunión .....	40
<b>Tabla 4</b> Tabla de Born Down .....	43
<b>Tabla 5</b> Historia de usuario – Visita asociación.....	46
<b>Tabla 6</b> Historia de usuario – Elaboración entrevistas.....	47
<b>Tabla 7</b> Historia de usuario –Análisis de la información .....	50
<b>Tabla 8</b> Historia de usuario - Requerimientos.....	51
<b>Tabla 9</b> Historia de usuario - Mockups .....	52
<b>Tabla 10</b> Historia de usuario – Base de datos .....	53
<b>Tabla 11</b> Historia de usuario – Módulo inicio sesión .....	54
<b>Tabla 12</b> Historia de usuario – Listado socios .....	56
<b>Tabla 13</b> Historia de usuario - Ajustes .....	59
<b>Tabla 14</b> Historia de usuario – Registro producción.....	61
<b>Tabla 15</b> Historia de usuario- Cálculos de pagos.....	63
<b>Tabla 16</b> Historia de usuario – Cargue de archivos .....	64
<b>Tabla 17</b> Historia de usuario – Panel de control .....	66
<b>Tabla 18</b> Historia de usuario – Registros materiales.....	68
<b>Tabla 19</b> Historia de usuario – Registro animales .....	69
<b>Tabla 20</b> Historia de usuario – Visualización de producción .....	70
<b>Tabla 21</b> Historia de usuario – Visualización de archivos .....	72
<b>Tabla 22</b> Fases de la metodología mixta .....	73

<b>Tabla 23</b>	Requerimiento Funcional 1 .....	77
<b>Tabla 24</b>	Requerimiento Funcional 2 .....	79
<b>Tabla 25</b>	Requerimiento Funcional 3 .....	80
<b>Tabla 26</b>	Requerimiento Funcional 4 .....	81
<b>Tabla 27</b>	Requerimiento Funcional 5 .....	81
<b>Tabla 28</b>	Requerimiento Funcional 6 .....	83
<b>Tabla 29</b>	Requerimiento Funcional 7 .....	84
<b>Tabla 30</b>	Requerimiento Funcional 8 .....	84
<b>Tabla 31</b>	Requerimiento Funcional 9 .....	85
<b>Tabla 32</b>	Requerimiento Funcional 10 .....	87
<b>Tabla 33</b>	Requerimiento No Funcional 1 .....	88
<b>Tabla 34</b>	Requerimiento No funcional 2 .....	89
<b>Tabla 35</b>	Requerimiento No Funcional 3 .....	90
<b>Tabla 36</b>	Requerimiento No Funcional 4 .....	91
<b>Tabla 37</b>	Requerimiento No Funcional 5 .....	92
<b>Tabla 38</b>	Requerimiento No Funcional 6 .....	93
<b>Tabla 39</b>	Prueba unitaria .....	104
<b>Tabla 40</b>	Segunda prueba unitaria .....	105
<b>Tabla 41</b>	Tercera prueba unitaria .....	106
<b>Tabla 42</b>	Cuarta prueba unitaria .....	106
<b>Tabla 43</b>	Quinta Prueba Unitaria .....	108
<b>Tabla 44</b>	Sexta prueba unitaria .....	109
<b>Tabla 45</b>	Séptima prueba unitaria .....	110

<b>Tabla 46</b> Octava prueba unitaria .....	110
<b>Tabla 47</b> Cronograma de actividades AsoVapp.....	120

## Índice de figuras

<b>Figura 1</b>	Registros de producción lechera.....	16
<b>Figura 2</b>	Diagrama Burn Down.....	44
<b>Figura 3</b>	Metodología.....	45
<b>Figura 4</b>	Evidencia de respuesta.....	76
<b>Figura 5</b>	Diagrama de contexto.....	94
<b>Figura 6</b>	Caso de uso Inicio sesión .....	95
<b>Figura 7</b>	Diagrama de casos de Usos de presidente .....	96
<b>Figura 8</b>	Diagrama caso de uso socio.....	97
<b>Figura 9</b>	Logo de AsoVapp.....	99
<b>Figura 10</b>	Diseño de la base de datos .....	100
<b>Figura 11</b>	Mockups de AsoVapp .....	101
<b>Figura 12</b>	Prueba unitaria de Inicio de Sesión .....	102
<b>Figura 13</b>	Prueba Unitaria del registro.....	103
<b>Figura 14</b>	Prueba de velocidad al sistema AsoVapp.....	112
<b>Figura 15</b>	Tiempo de carga .....	113
<b>Figura 16</b>	Actuación del sistema.....	114
<b>Figura 17</b>	Estructura del sistema.....	115
<b>Figura 18</b>	Gráfico de cascada del sistema.....	116
<b>Figura 19</b>	Historia del sistema .....	117
<b>Figura 20</b>	Evidencia .....	119

## **Introducción**

La transformación digital y tecnológica ha llegado a todos los sectores económicos del mundo donde las Asociaciones ganaderas no se quedan por fuera, ya que se han implementado herramientas que ayudan al mejoramiento productivo de la actividad lechera, y a su vez aportan al crecimiento del sector económico. Sin embargo, el acceso a estas tecnologías es una posibilidad que no tienen todas las Asociaciones en las regiones colombianas, debido a los elevados costos y las pocas facilidades de obtención.

En muchas de las empresas o entidades lecheras los ganaderos dependen de modelos empíricos para controlar sus procesos. Sin embargo, la falta de un sistema organizado y estructurado limita su capacidad para planificar, gestionar y administrar eficientemente su producción. Esto puede resultar en la pérdida de beneficios y dificultades para alcanzar objetivos claros. Implementar un sistema de control de procesos adecuado permitiría una mejor planificación, gestión de recursos, control de calidad y análisis de datos, lo que maximizaría la eficiencia y rentabilidad en la producción de leche.

El uso de sistemas de información en varios casos ayuda a aumentar la productividad de los empleados y encargados de las organizaciones o asociaciones permitiendo llevar un registro y control de la información y datos que se generan, produciendo con mayor eficiencia el desarrollo de actividades administrativas y económicas que dicha entidad plantee.

Según el empresario Andrés René Gutiérrez Tiuso columnista en la Revista Empresarial y

Laboral:

Las organizaciones de acuerdo con sus necesidades pueden contar con aplicaciones web y móviles para gestionar operaciones propias. Cada una de ellas

debe encontrar la mejor manera para optimizar sus procesos y cumplir con las crecientes demandas de sus clientes o asociados, y muchas veces la mejor manera de hacerlo es con software desarrollado a la medida. (René, 2020)

Para el desarrollo de este proyecto se tendrán en cuenta diferentes metodologías y marcos de trabajo con el fin de hacer un proceso ágil y de alta calidad, se tendrá en cuenta la metodología de investigación mixta para los procesos de recolección y análisis de datos que permitirán identificar los problemas principales a los cuales se les dará solución. Se utilizará el marco de trabajo Scrum, para organizar las actividades necesarias para llevar a cabo el proyecto, y finalmente se utilizará la metodología de desarrollo XP para ejecutar las actividades de desarrollo de software requeridas para la elaboración del producto, en este caso un sistema de información.

El sistema de información contará con tres roles, el presidente, que podrá consultar datos de cada uno de los socios como la información de las fincas y la producción lechera y registrar los proyectos en los que participan como asociación. La administradora de tanque a la cual le permitirá registrar la cantidad de leche producida por los socios y la recolección de litros de leche vendidos a las empresas recolectoras, por último, el rol del socio, donde podrá consultar su producción y registrar cada uno de sus animales con sus respectivas características, lo que permitirá tener una mejor organización y control de sus actividades diarias.

## **Resumen**

Se realizó un sistema de información el cual permitió a los usuarios realizar seguimiento y control a las actividades administrativas que desempeña dentro de la asociación de ganaderos Asoagroproleche, mediante el registro de los datos que recolecta o genera al realizarla, facilitando así la organización de cada uno de los procesos. Teniendo en cuenta que la asociación está conformada por diferentes roles (presidente, administrador, ganadero), el sistema cuenta con funciones personalizadas para cada uno de estos, los cuales les permitió registrar y tener un historial de la leche recolectada, almacenar información acerca de los proyectos propuestos para la asociación y los pagos realizados a los productores lecheros, además les condescendió a los ganaderos conocer la cantidad de producto entregado y la información de sus animales en sus fincas.

En el presente documento se muestra el proceso que se llevó a cabo para el desarrollo del proyecto, se describe el problema a solucionar dentro de la asociación, y metodologías de trabajo, investigación y desarrollo que permitieron plantear, obtener y organizar las actividades requeridas para cumplir con los objetivos del proyecto.

### **Palabras clave**

Producción lechera, sistema de información, datos de ganaderos, recolección de información, registro de socios.

### **Abstract**

An information system was developed that allowed users to monitor and control the administrative activities carried out within the Asoagroproleche cattlemen's association by recording the data collected or generated by the association, thus facilitating the organization of each of the processes. Taking into account that the association is made up of different roles (president, administrator, farmer), the system has customized functions for each of them, which allowed them to record and have a history of the milk collected, store information about the projects proposed for the association and the payments made to dairy farmers, and also allowed farmers to know the amount of product delivered and the information of their animals on their farms.

This document shows the process that was carried out for the development of the project, describes the problem to be solved within the association, and the work, research and development methodologies that made it possible to plan, obtain and organize the activities required to meet the project's objectives.

### **Key Words**

Dairy production, information system, farmer data, data collection, membership registration.

## **Definición del problema**

### **3.1. Descripción de la situación problemática**

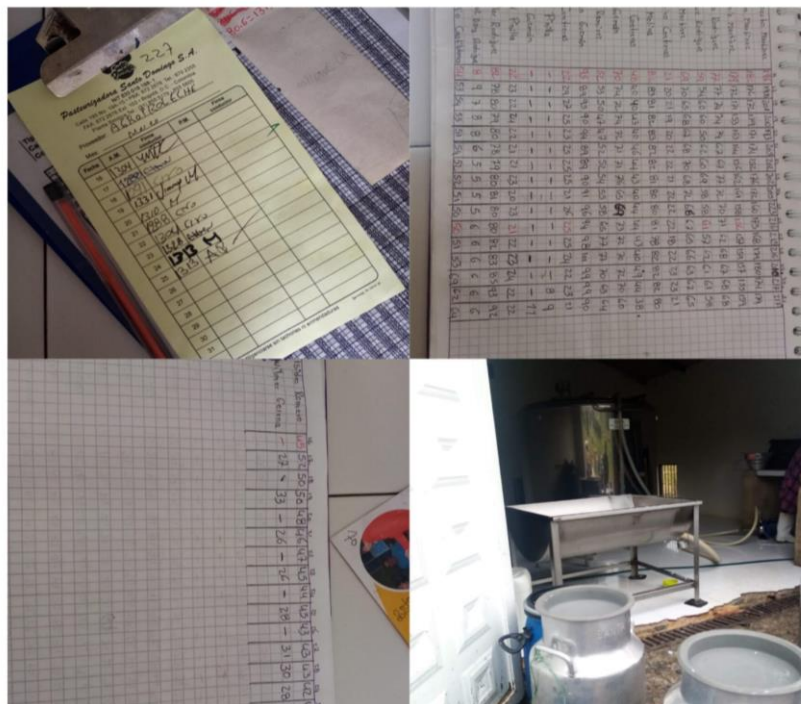
La asociación de ganaderos denominada Asoagroproleche es una empresa sin ánimo de lucro ubicada en la vereda Hatico y Eneas del municipio de Carmen de Carupa que se dedica al recaudo o acopio de producción lechera que los socios o ganaderos obtienen en sus fincas, para luego realizar el proceso de enfriamiento y venta a las empresas de la región que se dedican a elaborar derivados lácteos.

En la actualidad, esta entidad realiza seguimiento y control a cada una de las actividades que desarrolla de manera manual, por medio de cuadernos u hojas de Excel, lo que hace que se puedan generar pérdidas de información tanto de la producción lechera como los datos de cada uno de los socios o ganaderos que la conforman, a su vez la seguridad de la información es mínima, ya que al encontrarse de manera física cualquiera puede acceder a esta y manipularla.

En cuanto a las técnicas en los cálculos de pago de la producción lechera de cada uno de los socios, se realizan utilizando calculadora, papel y lápiz, lo que hace que este proceso sea tedioso y demorado, llegando a causar errores de digitación, lo que lleva a resultados erróneos que pasan desapercibidos y pueden generar pérdida de dinero en el total de la cantidad de litros producidos. Debido a los cálculos de pago proclives a errores, la asociación se ha visto afectada económicamente, pues debe reponer el dinero que se pierde debido a los procesos mencionados anteriormente.

## Figura 1

### *Registros de producción lechera*



*Nota.* Evidencia el uso de cuadernos y libretas para el registro de la información.

Debido a que la asociación de ganaderos realiza labores de recolección de leche, es necesario contar con una serie de documentación legal exigida por entidades de control como el ICA, dicha documentación es administrada por medio de aplicaciones de mensajería instantánea, lo que ha generado confusión en el proceso de almacenamiento al no existir un espacio adecuado para el manejo de dicha documentación, debido a lo mencionado anteriormente, se ha generado pérdida de información que es de suma importancia para la asociación.

Por otro lado, los ganaderos no cuentan con un método que les permita conocer la producción lechera que han ingresado a la asociación, por lo tanto, deben manejar registros paralelos a los de la entidad, causando duplicidad en la información recolectada, además debido a

que sus técnicas de almacenamiento de información se basan principalmente en cuadernos o libretas que deben cambiar periódicamente, donde la información tiende a perderse. Para ver el árbol de problemas [Ver anexos](#)

### **3.1.1. *Formulación del problema***

¿Cómo optimizar los procesos administrativos en la asociación de ganaderos Asoagroproleche ubicada en el municipio de Carmen de Carupa a través de software a la medida?

## **Justificación**

Las diferentes asociaciones ganaderas representan a los pequeños y medianos productores de leche en la región, por medio de estas pueden recibir apoyo en planes y proyectos que permitan el desarrollo social y económico dentro de sus actividades, beneficiando así al gremio de ganaderos.

La empresa lechera Asoagroproleche, lleva diferentes registros como el de la producción lechera, proyectos en los que participan, el personal que hace parte de la asociación como su estado y comportamiento dentro de la entidad, propiedades de la leche producida en cada una de las fincas ganaderas vinculadas a esta empresa. De manera que la falta de herramientas y tecnologías que se acoplen a las necesidades de la entidad hace que lleven el seguimiento de su información de manera manual haciendo que se vean afectados los usuarios por el uso de métodos poco eficientes y tediosos, por tal motivo la ejecución del sistema de información y de nuevas tecnologías permita que los datos de la asociación estén seguros, y de esta forma tener claridad y transparencia en las tareas que se realizan.

El sistema facilita los procesos administrativos de la asociación logrando buenas metodologías de organización y éxito para la toma de decisiones en esta entidad.

Este proyecto se enfoca a las líneas de investigación de desarrollo de la gestión administrativa, económica, financiera de producción de operaciones regionales, locales y a la gestión de tecnología aplicada a los sectores agropecuarios donde se relaciona con el sistema de información que va dirigido para el seguimiento de procesos administrativos de la asociación Asoagroproleche. En cuanto a los objetivos de desarrollo sostenible, se aporta al del trabajo decente y crecimiento económico, donde el sistema permite tener una estructura concreta y organizada de la asociación, lo que genera beneficios a todos los integrantes, desde los directivos

a los ganaderos debido a que ayuda en la economía a tener ganancias sostenibles que concuerden con las entradas y salidas de la producción lechera y así poder llevar buenos balances en cuanto al dinero y el producido de la leche para el progreso de esta población.

Según el periódico El Tiempo Cundinamarca es el segundo departamento más productor de leche del país, con 4.800.000 litros diarios y de esta actividad dependen más de 70.000 familias, la provincia de Ubaté que ha sido caracterizada como la capital lechera de Colombia debido a que se estima que diariamente se producen más 70 mil litros de leche que alimentan a empresas muy importantes en ese sector industrial. (El Tiempo, 2019)

A partir de lo anteriormente mencionado se plantea el desarrollo de un sistema con los mejores estándares y flexibilidad en la funcionalidad y calidad que permita optimizar los procesos realizados en la asociación Asoagroproleche ubicada en el municipio de Carmen de Carupa.

## **Objetivos**

### **5.1. Objetivo general**

Desarrollar un Sistema de información para mejorar los procesos administrativos en la asociación de ganaderos Asoagroproleche ubicada en el municipio de Carmen de Carupa.

### **5.2. Objetivos específicos**

Identificar las características de los procesos administrativos de la asociación Asoagroproleche para definir las necesidades de esta.

Diseñar el sistema de información para el apoyo administrativo de la entidad, obteniendo cada uno de los módulos del sistema.

Validar la efectividad del sistema de información por medio de mediciones de indicadores que permitan determinar el grado de optimización de los procesos administrativos.

## Alcances y limitaciones

### 6.1. Alcances

El sistema de información AsoVapp contará con tres roles, el presidente que es el encargado del manejo de la información de los usuarios y proyectos en los cuales participa la entidad lechera.

Le permitirá consultar estadísticas de producción lechera general e individual de los socios, y registrar información y documentación acerca de los proyectos.

La administradora de tanque es la persona encargada de registrar la cantidad de litros de leche entrantes y salientes de cada uno de los socios y de las empresas recolectoras. El sistema le permitirá realizar los cálculos correspondientes a estas actividades.

El socio es el encargado de suministrar la producción lechera de la asociación y el sistema le permitirá conocer la cantidad de litros de leche que ha ingresado, además de permitirle registrar los animales que se encuentren en sus fincas.

El sistema permitirá a la organización mejorar la toma de decisiones al proporcionar datos en tiempo real sobre la producción de leche, lo que ayudará a la organización a identificar oportunidades para mejorar la eficiencia.

El sistema permitirá a la organización mejorar la seguridad de los datos al garantizar que la información personal y comercial de los socios y las empresas recolectoras se almacene de manera segura y se proteja de posibles violaciones de seguridad.

## **6.2. Limitaciones**

El sistema se desarrollará específicamente para la asociación de ganaderos Asoagroproleche ubicada en la vereda Hatico y Eneas del Municipio de Carmen de Carupa para el manejo y seguimiento de cada uno de los procesos y actividades administrativas que realiza cada integrante, dependiendo sus funciones dentro de la entidad.

Para hacer uso de la plataforma se requiere de conexión a internet y de un dispositivo capaz de desplegar el sistema.

El sistema no se encargará de labores contables, solo realizará cálculos acerca de la producción lechera de la asociación.

El presupuesto con el que se cuenta es limitado, con lo cual es necesario considerar el uso de soluciones de software de código abierto en lugar de soluciones comerciales de pago para reducir los costos. Además, es necesario priorizar las características y funcionalidades esenciales del sistema para no incluir funciones secundarias que generen más costos.

## **Marco de referencia**

En este marco se estableció una previa investigación acerca de los antecedentes, teorías como proyectos que se han desarrollado para el manejo de procesos administrativos en las empresas lecheras a nivel internacional, nacional y local lo que permite conocer los que se tendrán como referencia para la elaboración del sistema de información.

Se realizó una consulta acerca de los aspectos legales que se deberán tener en cuenta para el desarrollo de los diferentes objetivos que permitirán llevar a cabo el cumplimiento de los requerimientos inicialmente planteados.

### **7.1. Estado del arte**

Se realizó una compilación de investigaciones provenientes de proyectos que ya se desarrollaron para el manejo de los procesos que se realizan como ayuda para la toma de decisiones que generaron un impacto en la sociedad o entorno en el que se llevó a cabo.

#### **7.1.1. *Internacional***

El proyecto EasyTambo desarrollado por alumnos de la Universidad ORT de Uruguay, es un sistema para el apoyo a la toma de decisiones en establecimientos de producción de leche. Este contribuye al desarrollo del sistema ya que ambos trabajan en base a las necesidades identificadas en el sector agropecuario en especial el sector lechero ya que ha crecido a gran escala en producción y derivados, lo que hace que se recurra al uso de nuevas tecnologías, sistemas móviles y web para estos lugares del campo y herramientas que permitan servir de apoyo para la toma de decisiones y el progreso de las organizaciones y así aumentar las ganancias y productividad de este sector.

En efecto, el 70% de las exportaciones del Uruguay son de origen agropecuario o agroindustrial. Dentro del sector agropecuario, la lechería ocupa un rol muy importante en la estructura económica de Uruguay, siendo uno de los sectores de la industria que genera mayor valor agregado. Por otra parte, el desarrollo del proyecto representó un gran desafío para el equipo puesto que se utilizaron tecnologías con las cuales no tenía experiencia. Sin embargo, esto le permitió al equipo incorporar conocimiento acerca de tecnologías requeridas actualmente en el mercado laboral. (Medina, 2022)

También EasyTambo maneja la metodología de trabajo Scrum para el desarrollo del prototipo donde trabajaron de la mano con el cliente definiendo los requerimientos funcionales y no funcionales para la construcción de los diferentes sprints es decir cada una de las tareas que se plantean como equipo para lograr el objetivo de realizar una herramienta innovadora.

### **7.1.2. Nacional**

Otro de los sistemas de información encontrados es el Sistema De Información Granja El Tíbar, es un sistema de información web creado para la granja el Tíbar de la Universidad de Cundinamarca que permite el registro de animales bovinos, procesar datos los cuales se pueden mostrar en cualquier dispositivo con conexión a internet este ayuda a la toma de decisiones al encargado de administrar.

El código fuente del software queda abierto para poder implementar todas las explotaciones de la granja y así mismo poder seguir mejorándolo, dando a conocer la granja el Tíbar de la Universidad de Cundinamarca seccional Ubaté ante la comunidad. (Vargas Parra, 2016)

### **7.1.3. Local**

Finalmente, está el Sistema De Información Para La Gestión De La Actividad Lechera Del Hato San Fernando Ubicado En El Municipio De Fusagasugá. Este va dirigido al sector lechero y a los procesos que se ejecutan en las fincas de los ganaderos que hacen uso de un software donde registran las características de los animales como su reproducción, nacimientos para poder llevar un control y seguimiento de la salud y bienestar de sus vacas garantizando una mejor producción de leche y beneficios para todos.

Colombia es uno de los países más importantes de la agroindustria, por su gran geografía se tiene presente que el sector lechero está cubierto a nivel nacional, sin embargo, el área andina que está a más de 2000 m.s.n.m producen un gran porcentaje que está destinada a las grandes industrias del país. Hay que aclarar que, en la mayoría de los casos el control de procesos que se realizan por los modelos empíricos que los ganaderos implementan, limitan la planificación, gestión y administración de las producciones de leche y eso puede hacer que pierdan beneficios que pueden aprovechar, debido que al no llevar un buen sistema de organización y gestión puede hacer que sus procesos sean más lentos haciendo que el orden se pierda junto con unos resultados, objetivos y metas realmente claros. (Munevar Rativa, 2021)

## **7.2. Marco teórico**

### **7.2.1. Sistemas de información**

Para las diferentes organizaciones es indispensable contar con sistemas que les ayuden con los procesos administrativos, es decir, a recopilar, procesar y distribuir información, ya que

les permite tener un mayor control y conocimiento de los procesos de la organización, además de ayudar a incrementar sus ganancias y mejorar los servicios con los clientes, a partir de esta necesidad es que surgen los sistemas de información, pues según Ralph Stair y George Reynolds “un sistema de información es un conjunto de elementos o componentes interrelacionados que recaban (entrada), manipulan (proceso), almacenan y distribuyen (salida) datos e información y proporciona una reacción correctiva (mecanismo de retroalimentación) si no se ha logrado cumplir un objetivo.” (Stair, 1996)

Para el presente proyecto se toman en cuenta los fundamentos principales de los sistemas de información, pues la asociación a la cual va dirigido, no cuenta con herramientas que le permitan gestionar y procesar la información que se genera en ésta.

### ***7.2.2 Sistema de información gerencial***

El marco teórico del "Sistema de Información Gerencial" se puede aplicar al proyecto para proporcionar una base sólida para el diseño y la implementación del sistema. Según Kenneth C. Laudon y Jane P. Laudon en su libro "Sistemas de Información Gerencial", un SIG es "un conjunto integrado de componentes que recopila, almacena, procesa y distribuye información para apoyar la toma de decisiones, la coordinación, el control, el análisis y la visualización en una organización". (Kenneth C. Laudon, 2016)

El SIG se utiliza para facilitar la toma de decisiones, aumentar la eficiencia y mejorar la eficacia de la gestión de la organización.

En el contexto del proyecto, el SIG puede ayudar a la organización a recopilar, procesar y distribuir información relevante sobre la producción de leche, lo que puede mejorar la toma de

decisiones y aumentar la eficiencia en la gestión de los procesos de recolección y facturación.

Además, el SIG puede ayudar a la organización a mejorar la eficacia de su gestión al proporcionar una solución centralizada para el almacenamiento y gestión de datos, lo que puede ayudar a reducir errores y redundancias en los procesos.

### **7.2.2. *Sistemas de planificación de recursos (ERP)***

Los procesos administrativos de las empresas y organizaciones han sufrido cambios en el tiempo, y en la actualidad la tecnología ha servido para automatizar y organizar la información en sistemas complejos. Estos sistemas se conocen como software administrativo o ERP (Enterprise Resource Planning). Según Davenport “un sistema ERP es un paquete de software comercial que integra toda la información que fluye a través de la compañía: información financiera y contable, información de recursos humanos, información de la cadena de abastecimiento e información de clientes” (Davenport, 1998).

Estos softwares se encargan de diferentes actividades y Holland y Light” un ERP automatiza las actividades corporativas nucleares, tales como: manufactura, recursos humanos, finanzas y gestión de la cadena de abastecimiento, incorporando las mejores prácticas para facilitar la toma de decisiones rápida, la reducción de costes y el mayor control directivo” (Holland Ben Light, 1999).

El presente proyecto al estar enfocado en una asociación en específico toma como base los procesos y funciones propias de los ERP, pues la gestión de la información tanto financiera como de recursos humanos de la asociación se gestionará con el sistema de información propuesto en este proyecto.

### **7.2.3. Teoría general de sistemas**

La Teoría General de Sistemas tiene una gran aplicación en el proyecto ya que permite analizar el proyecto en su conjunto, considerando los diferentes elementos y subsistemas que lo componen, desde los usuarios hasta la infraestructura tecnológica. De esta forma, se pueden identificar las interacciones entre los diferentes integrantes de la asociación de ganaderos. Por ejemplo, la interacción entre el rol del presidente, socio y el de la administradora de tanque puede ser analizada a través de la TGS, para identificar cómo se relacionan y cómo afecta la información que maneja cada uno de ellos al resto del sistema.

Según Ludwig von Bertalanffy, la TGS "son un conjunto de conceptos, proposiciones, métodos y técnicas que se han unificado para estudiar problemas específicos y cuya aplicación se extiende a diversas disciplinas" (Bertalanffy, 1968). La TGS también permite aplicar un enfoque de pensamiento sistémico en la planificación, desarrollo y evaluación del proyecto. Esto implica considerar al proyecto como un sistema complejo, donde cada elemento y proceso es interdependiente y se influyen mutuamente. De esta forma, se pueden identificar las posibles consecuencias y efectos secundarios de cada acción y decisión tomada en el proyecto, lo que permite tomar decisiones informadas y minimizar los riesgos. Por ejemplo, se puede utilizar la TGS para evaluar cómo la implementación de un nuevo módulo en el sistema puede afectar a otros procesos ya existentes, considerando cómo se relacionan entre sí y cómo pueden afectarse mutuamente.

### **7.2.4. Software**

El software de computadora es el producto que construyen los programadores profesionales y al que después le dan mantenimiento durante un largo tiempo. Incluye

programas que se ejecutan en una computadora de cualquier tamaño y arquitectura, contenido que se presenta a medida que se ejecutan los programas de cómputo e información descriptiva tanto en una copia dura como en formatos virtuales que engloban virtualmente a cualesquiera medios electrónicos. La ingeniería de software está formada por un proceso, un conjunto de métodos (prácticas) y un arreglo de herramientas que permite a los profesionales elaborar software de cómputo de alta calidad. (Pressman, 2010)

El desarrollo de software es una de las etapas principales de este proyecto, pues el sistema planteado para la gestión de la información de la asociación, es una página web, por lo tanto, requiere de una serie de procedimientos descritos por autores como Pressman, en los cuales se definen las principales características y requisitos que se deben tener en cuenta para el desarrollo de este software.

#### **7.2.5. *Desarrollo ágil***

La ingeniería de software ágil combina una filosofía con un conjunto de lineamientos de desarrollo. La filosofía pone el énfasis en: la satisfacción del cliente y en la entrega rápida de software incremental, los equipos pequeños y muy motivados para efectuar el proyecto, los métodos informales, los productos del trabajo con mínima ingeniería de software y la sencillez general en el desarrollo. Los lineamientos de desarrollo enfatizan la entrega sobre el análisis y el diseño (aunque estas actividades no se desalientan) y la comunicación activa y continua entre desarrolladores y clientes. (Pressman, 2010)

La productividad es esencial en el desarrollo de proyectos, cada integrante del equipo de trabajo debe tener una serie de responsabilidades que contribuyan en la ejecución del proyecto, y para este apartado se tomara como base las metodologías ágiles, pues permiten generar entornos

de comunicación constante y roles de trabajo bien definidos que permiten optimizar el tiempo de desarrollo del proyecto sin perder la calidad del producto.

### **7.2.6. ISO/IEC 25000**

Asegurar la calidad de los sistemas web es de suma importancia, para ello se cuenta con diferentes herramientas que permiten medir el rendimiento del sistema, pero a la vez se cuenta con una normativa que indica los aspectos a tener en cuenta al momento de desarrollar un software. En el caso del presente proyecto se tomará en cuenta la familia de normas ISO/IEC 25000, conocida como SQuaRE (System and Software Quality Requirements and Evaluation), es una familia de normas que tiene por objetivo la creación de un marco de trabajo común para evaluar la calidad del producto software. (Standars, 2019)

El desarrollo de software es uno de los objetivos principales de este proyecto, y para medir la calidad del producto que se está desarrollando es necesario tener en cuenta normas internacionales como la mencionada anteriormente.

## **7.3. Marco conceptual**

El desarrollo de este proyecto sirve de apoyo para la asociación de ganaderos “Asoagroproleche” ya que, por medio de este, se realiza seguimiento y control a cada uno de los procesos que se llevan a cabo en dicha organización por medio de los administradores, recolectores y asociados. A continuación, se presenta la definición y contextualización de algunos conceptos que se relacionan directamente con el sistema.

### **7.3.1. Base De Datos**

El sistema de información tiene en su desarrollo una base de datos la cual permite almacenar cantidad de datos relacionados y estructurados acerca de información de los

integrantes de la asociación “Asoagroproleche” como la producción lechera, los proyectos en los que se benefician y de los animales de las fincas de los socios.

En una base de datos típica, los datos se almacenan en un formato que hace fácil acceder, ya sea para consultas individuales o grandes reportes. En un entorno de base de datos integrada, todos los datos se mantienen en un solo repositorio llamado base de datos, y se gestiona mediante el administrador de base de datos (ABD). Todo acceso a la base de datos es a través del sistema de gestión de base de datos (DBMS), un paquete de software que configura estructuras de almacenamiento carga datos, proporciona acceso a programas y usuarios interactivos, formatea datos recuperados, oculta ciertos datos, realiza actualizaciones, controla concurrencia y efectúa respaldos y recuperación para la base de datos. Los datos en una base de datos están integrados, son autodescriptivos y se comparten de manera concurrente mediante muchos usuarios. ( Bases de datos. Up.ac.pa., 2016)

### **7.3.2. Sistema Gestor De Bases De Datos**

El sistema gestor de base de datos de AsoVapp es MySQL de administración relacional, es decir, datos estructurados en tablas que son compuestas por cada uno de los atributos que sean ingresados por el usuario permitiendo guardar la información que se maneja en la asociación Asoagroproleche de manera organizada.

Según Definimos un Sistema Gestor de Bases de Datos o SGBD, también llamado DBMS (Data Base Management System) como una colección de datos relacionados entre sí, estructurados y organizados, y un conjunto de programas que acceden y gestionan esos datos. La colección de esos datos se denomina Base de Datos o BD, (DB Data Base). (Modrego, 2006)

### **7.3.3. *Modelo Vista Controlador***

Es una arquitectura de software que se trabaja por capas que facilitan el flujo de la aplicación, donde se separan los datos que se manejan en el sistema (Modelo), la interfaz que interactúa con el usuario final (Vista) y la respectiva lógica que es la que se encarga de gestionar la transformación de los datos (Controlador). Este concepto puede tener un impacto significativo en el proyecto, ya que puede mejorar la eficiencia, escalabilidad, mantenibilidad y seguridad del sistema, mediante la separación de la lógica de negocio, la presentación al usuario y la gestión de las interacciones del usuario.

Es un patrón en el diseño de software comúnmente utilizado para implementar interfaces de usuario, datos y lógica de control. Enfatiza una separación entre la lógica de negocios y su visualización. Esta "separación de preocupaciones" proporciona una mejor división del trabajo y una mejora de mantenimiento. (Mozilla.org, 2022)

### **7.3.4. *Lenguaje SQL***

Es el lenguaje de consulta estructurado que se usa como interfaz que permite crear y manipular con mayor rapidez y facilidad cualquier operación de acceso y manejo de información almacenada en la base de datos. Con ayuda de las sentencias de SQL como create, select, drop, insert, delete entre otros permiten que usar condiciones para realizar cualquier acción en una tabla o campos específicos indicando que se muestren solamente los registros que se soliciten.

### **7.3.5. *Lenguaje De Programación***

Para realizar el desarrollo se usan diferentes lenguajes como PHP, HTML, JavaScript y CSS los cuales permiten describir diferentes series de instrucciones y comandos específicos a el

equipo del desarrollador realizar cada uno de los módulos dándoles el respectivo diseño y funcionamiento logrando ser interactivo y amigable con el usuario final del sistema.

Un lenguaje de programación es un conjunto de símbolos y reglas sintácticas y semánticas que definen su estructura y el significado de sus elementos y expresiones, y es utilizado para controlar el comportamiento físico y lógico de una máquina.

Un lenguaje de programación permite especificar de manera precisa sobre qué datos debe operar una computadora, cómo estos datos deben ser almacenados o transmitidos y qué acciones debe tomar bajo una variada gama de circunstancias. Todo esto, a través de un lenguaje que intenta estar relativamente próximo al lenguaje humano o natural, tal como sucede con el lenguaje Léxico. (Modificación, 2008)

#### **7.3.6. TIC**

Las TIC son las tecnologías las cuales permiten guardar, procesar y mostrar información al tener cualquier tipo de comunicación y navegación por medio de teléfonos, televisores, computadores entre otros equipos. El sistema AsoVapp para que se despliegue necesita de mínimo un celular o computadora con acceso a internet para que permita al usuario final registrar sus datos para ser guardados de forma segura y eficaz en una base de datos para poder mostrar un análisis de graficas con la información ingresada por el presidente, administradora de tanque y el ganadero.

Las TIC “sirvieron de soporte a familias, empresas y sociedad en general para mantenerse comunicadas, informadas e intentar continuar con sus vidas en medio de la ‘nueva normalidad’”, dice Ismael Arenas, presidente de la Asociación Colombiana de Ingenieros

(Aciem). Para Samuel Hoyos, presidente del gremio de operadores Asomóvil, si el personaje de 2020 fuera un sector, sería el de las TIC (Medina., 2021)

#### **7.4. Marco Legal**

Para el desarrollo del sistema de información AsoVapp se establece la respectiva normatividad que se requiere con el manejo de datos personales. La constitución política de Colombia en la Ley 1581 de 2012 establece una serie de directrices que se deben tener en cuenta al momento de la recolección y tratamiento de datos personales, debido a que el sistema de información que se quiere implementar en la asociación de ganaderos Asoagropoleche, utiliza bases de datos para almacenar información sensible y datos personales de los ganaderos, es necesario aplicar la normatividad vigente acerca de estos temas.

La Constitución Política de Colombia en su artículo 15 de la ley 1581 de 2012 establece que:

Todas las personas tienen derecho a su intimidad personal y familiar y a su buen nombre, y el Estado debe respetarlos y hacerlos respetar. De igual modo, tienen derecho a conocer, actualizar y rectificar las informaciones que se hayan recogido sobre ellas en bancos de datos y en archivos de entidades públicas y privadas. (Constitución Política de Colombia, 1991)

De igual forma en el artículo 4 de la ley 1581 de 2012 se establecen una serie de principios que se deben cumplir al momento de almacenar y tratar información sensible en bases de datos, la constitución política define lo siguiente:

- a) . Principio de legalidad en materia de Tratamiento de datos: El Tratamiento a que se refiere la presente ley es una actividad reglada que debe sujetarse a lo establecido en ella y en las demás disposiciones que la desarrollen;

- b) . Principio de finalidad: El Tratamiento debe obedecer a una finalidad legítima de acuerdo con la Constitución y la Ley, la cual debe ser informada al Titular;
- c) . Principio de libertad: El Tratamiento sólo puede ejercerse con el consentimiento, previo, expreso e informado del Titular. Los datos personales no podrán ser obtenidos o divulgados sin previa autorización, o en ausencia de mandato legal o judicial que releve el consentimiento;
- d) . Principio de veracidad o calidad: La información sujeta a Tratamiento debe ser veraz, completa, exacta, actualizada, comprobable y comprensible. Se prohíbe el Tratamiento de datos parciales, incompletos, fraccionados o que induzcan a error;
- e) . Principio de transparencia: En el Tratamiento debe garantizarse el derecho del Titular a obtener del Responsable del Tratamiento o del encargado del Tratamiento, en cualquier momento y sin restricciones, información acerca de la existencia de datos que le conciernan;
- f) . Principio de acceso y circulación restringida: El Tratamiento se sujeta a los límites que se derivan de la naturaleza de los datos personales, de las disposiciones de la presente ley y la Constitución. En este sentido, el Tratamiento sólo podrá hacerse por personas autorizadas por el Titular y/o por las personas previstas en la presente ley.

Los datos personales, salvo la información pública, no podrán estar disponibles en Internet u otros medios de divulgación o comunicación masiva, salvo que el acceso sea técnicamente controlable para brindar un conocimiento restringido sólo a los Titulares o terceros autorizados conforme a la presente ley;

g) . Principio de seguridad: La información sujeta a Tratamiento por el Responsable del Tratamiento o Encargado del Tratamiento a que se refiere la presente ley, se deberá manejar con las medidas técnicas, humanas y administrativas que sean necesarias para otorgar seguridad a los registros evitando su adulteración, pérdida, consulta, uso o acceso no autorizado o fraudulento;

h). Principio de confidencialidad: Todas las personas que intervengan en el Tratamiento de datos personales que no tengan la naturaleza de públicos están obligadas a garantizar la reserva de la información, inclusive después de finalizada su relación con alguna de las labores que comprende el Tratamiento, pudiendo sólo realizar suministro o comunicación de datos personales cuando ello corresponda al desarrollo de las actividades autorizadas en la presente ley y en los términos de la misma. (Ministerio de Ambiente y Desarrollo Sostenible, 2022)

## Marco metodológico

### 8.1. Metodología de trabajo en equipo

Para el desarrollo de este proyecto se utiliza la metodología Scrum con el fin de llevar a cabo un conjunto de tareas de forma regular con el objetivo principal de trabajar de manera colaborativa para obtener un sistema de alta calidad. Con este método de trabajo lo que se pretende es alcanzar el mejor resultado del proyecto determinando las prácticas que se aplican.

Se realiza la implementación de este marco en el sistema teniendo en cuenta que, esta es una metodología ágil y flexible para gestionar el desarrollo de software, que se basa en construir primero la funcionalidad de mayor valor para el cliente y en los principios de inspección continua, adaptación, autogestión e innovación. Principios de inspección continua, adaptación, autogestión e innovación.

Para este proyecto se presentan los roles para realizar cada una de las actividades que se definen en esta metodología. A continuación, se muestra una descripción de los roles, las tareas y funciones de cada integrante del equipo:

**Tabla 1**

*Asignación de roles*

<i>Roles</i>	<i>Integrantes</i>
<i>Scrum Máster</i>	Jairo Orlando Pascagaza
<i>Development team</i>	Jairo Pascagaza y Yeimy Ángel
<i>Product owner</i>	Ingeniero Cristian Cano

*Nota.* Se muestra la asignación de los roles en la metodología Scrum responsables de que se cumplan las actividades.

**Scrum Máster:** Para este rol se asignó al estudiante Jairo Orlando Pascagaza quien tiene la labor de que se cumplan las actividades según el cronograma que se estableció al iniciar

el proyecto.

**Development team:** Este rol está a cargo de los estudiantes Jairo Pascagaza y Yeimy Ángel quienes son los encargados de realizar el desarrollo de cada uno de los módulos que componen el sistema, teniendo en cuenta los requerimientos que se definieron con los estudios y entrevistas realizadas a la asociación.

**Product owner:** Este rol lo desempeña el Ingeniero Cristian Cano que es el responsable de definir, priorizar los requisitos del producto y de colaborar con el equipo de desarrollo para asegurar que se cumpla con el desarrollo del producto final.

### **8.1.1. *Product Backlog***

En este se tiene la gestión de cada uno de los requisitos en el desarrollo del sistema de información utilizando la metodología Scrum. Este representa la lista priorizada de todas las funcionalidades, características, que se le implementaron al producto final en este caso el sistema para la asociación Asoagroproleche.

**Tabla 2***Product Backlog*

<i>Item</i>	<i>Descripción</i>
Registro de usuarios	Los usuarios podrán registrasen en el sistema de información
Inicio de sesión	Podrán iniciar sesión con un correo y la respectiva contraseña.
Recuperación de contraseña	Con el correo el cual se registraron podrán recuperar la cuenta.
Gestionar socios	Los usuarios(presidente ,administradora ) podrán consultar o modificar el estado de los socios.
Registro de documentos	Tendrán la opción de registrar documentos de proyectos, registros de finca, certificados de vacuna.
Consultar estadísticas	El presidente y la administradora de tanque podrán consultar estadísticas de la producción lechera.
Ajuste de perfil de usuario	Los usuarios tendrán la opción de configurar el perfil.
Registrar producción lechera	La administradora de tanque registra la cantidad de producción lechera.
Registrar empresas recolectoras	La administradora registrara las empresas recolectoras a las que les venden la producción.
Predecir producción lechera	El presidente podrá predecir la producción lechera que obtendrá en tantos días.
Gestionar inventario	La administradora de tanque podrá registrar los elementos que usan en las instalaciones.

*Nota.* Se muestran las funcionalidades que los usuarios podrán realizar en el sistema de información.

**8.1.2. *Sprint Planning***

Este permitió al equipo de desarrollo establecer las metas claras y definir las tareas que se llevaron a cabo durante los Sprint. Esta nos benefició en que cada actividad o tarea se promueve la colaboración, la transparencia y la eficiencia en el trabajo y desarrollo del sistema.

**Tabla 3***Actividades que se planean en reunión***Objetivo:** Identificar los procesos administrativos de la asociación

<b>Tarea</b>	<b>Responsable</b>	<b>Estimación (Días)</b>
Diseñar entrevista para cada uno de los integrantes (presidente, administradora de tanque, socio)	Yeimy Angel , Jairo Pascagaza	2
Visitar las instalaciones de la asociación	Yeimy Angel , Jairo Pascagaza	1
Aplicar las entrevistas	Yeimy Angel , Jairo Pascagaza	1
Realizar análisis de los datos recolectados	Jairo Pascagaza	2
Identificar requerimientos	Yeimy Angel	3

*Nota.* Se muestran las características del Sprint Planning que se realizaron la reunión.**Tabla 4***Actividades a desarrollar***Objetivo:** Modelar diagramas UML

<b>Tarea</b>	<b>Responsable</b>	<b>Estimación (Días)</b>
Consultar la estructura de los diagramas UML	Yeimy Angel , Jairo Pascagaza	2
Identificar las acciones que realizara cada usuario en el sistema	Yeimy Angel , Jairo Pascagaza	3
Realizar diagrama de casos de uso	Yeimy Angel , Jairo Pascagaza	3
Realizar diagrama de contexto	Jairo Pascagaza	1
construir cada uno de los diagramas	Yeimy Angel	3

*Nota.* Se muestran las características e ítems de los Sprint Planning que se realizaron en esta reunión.

**Tabla 5***Actividades para la configuración*

<b>Objetivo:</b> Configurar el ecosistema y entorno de desarrollo			
<b>Tarea</b>	<b>Responsable</b>	<b>Estimación (Días)</b>	
Consultar las herramientas que se van a usar	Yeimy Angel , Jairo Pascagaza	2	
Instalar las herramientas a usar	Yeimy Angel , Jairo Pascagaza	2	
Aplicar una arquitectura para su mejor organización	Yeimy Angel , Jairo Pascagaza	1	
Configurar base de datos	Jairo Pascagaza	1	
Configurar a GitHub	Yeimy Angel	3	

*Nota.* Se muestra cada una de las actividades planteadas en la reunión.

**Tabla 6***Tareas planteadas*

<b>Objetivo:</b> Codificación Frontend y Backend del sistema			
<b>Tarea</b>	<b>Responsable</b>	<b>Estimación (Días)</b>	
Consultar plantillas de diseño en Bootstrap	Yeimy Angel	10	
Desarrollar los modelos(entidades, atributos y relaciones)	Yeimy Angel , Jairo Pascagaza	40	
Desarrollar las vistas (formularios, botones ,menús)	Yeimy Angel	20	
Realizar los respectivos controladores (consultas para acceder y manipular los respectivos datos)	Jairo Pascagaza	45	
Configurar a la base de datos	Jairo Pascagaza	20	

*Nota.* Se muestran cada una de las tareas a desarrollar para el próximo encuentro.

**Tabla 7***Proyección de tareas*

<b>Objetivo:</b> Realización de pruebas y subir a servidor		
<b>Tarea</b>	<b>Responsable</b>	<b>Estimación (Días)</b>
Consultar las pruebas finales que se le hacen a un sistema.	Yeimy Angel , Jairo Pascagaza	2
Realizar las pruebas unitarias , por módulos y de rendimiento	Yeimy Angel , Jairo Pascagaza	6
Hacer uso de herramientas para las pruebas de rendimiento.	Yeimy Angel , Jairo Pascagaza	2
Identificar los resultados obtenidos de las pruebas	Jairo Pascagaza	8
Realizar la mejoría de los aspectos con menor porcentaje de rendimiento.	Yeimy Angel	4

*Nota.* Se muestran las características e ítems de los Sprint Planning que se realizaron en las reuniones.

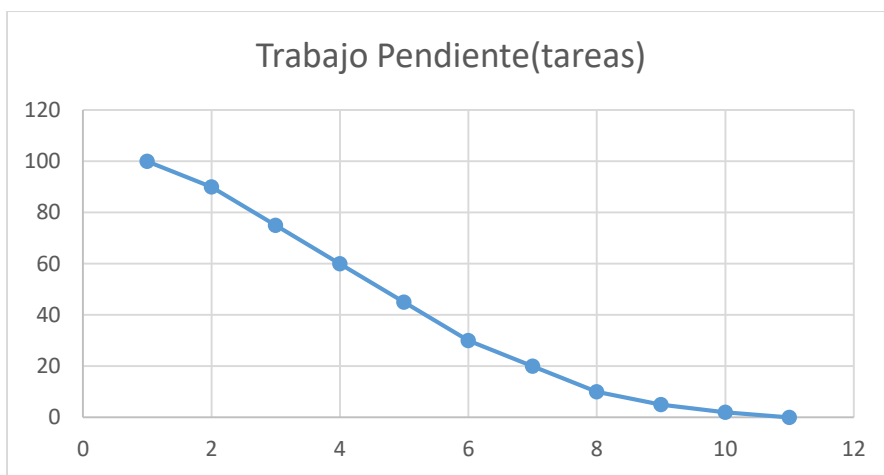
### 8.1.3. Diagrama de burn down

**Tabla 8**

*Tabla de Born Down*

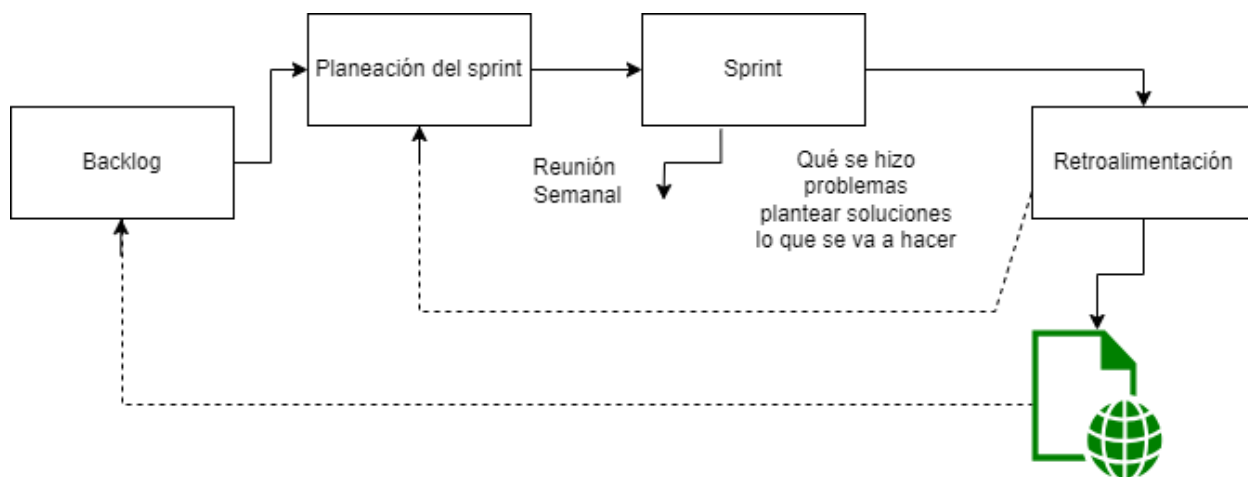
<i>Día</i>	<i>Trabajo Pendiente(tareas)</i>
<i>1</i>	100
<i>2</i>	90
<i>3</i>	75
<i>4</i>	60
<i>5</i>	45
<i>6</i>	30
<i>7</i>	20
<i>8</i>	10
<i>9</i>	5
<i>10</i>	2
<i>11</i>	0

*Nota.* Se muestra la cantidad de trabajo que queda por hacer al final de cada día. La cantidad de trabajo pendiente se actualiza diariamente a medida que se completan tareas y se avanza en el desarrollo del sistema de información.

**Figura 2***Diagrama Burn Down*

*Nota.* Se muestra el tiempo de las tareas pendientes que van haciendo falta para la finalización del proyecto.

A medida que los días avanzan, se espera que la cantidad de trabajo pendiente disminuya hasta llegar a cero al finalizar el proyecto. El equipo de desarrollo de AsoVapp puede registrar el progreso diario, lo que permite tener una visión clara de cómo se está desarrollando el sistema de información a lo largo del tiempo.

**Figura 3***Metodología*

*Nota.* Se muestra la ilustración de la metodología de trabajo Scrum.

Para la organización y desarrollo de las actividades, se realizarán reuniones semanales en las cuales se planteará una retroalimentación acerca de las labores realizadas por los integrantes del equipo, en dichas reuniones se discutirá las dificultades que se hayan presentado y se plantearan soluciones para abordar dichos problemas.

**Desarrollo Sprint 1**

Para este sprint se acordó realizar una visita a la asociación de ganaderos “Asoagropoleche”, con el fin de recolectar información acerca de los procesos y funcionamiento de esta.

La siguiente tabla contiene la información acerca de las características principales que se definieron para la realización del sprint.

**Tabla 9***Historia de usuario – Visita asociación*


---

**Historia de usuario 01**

---

**Número: 01**                      **Usuario:** presidente, administradora de tanque, socio

**Nombre de historia:** Visita a la asociación

**Prioridad:** Alta                      **Riesgo en desarrollo:** Medio

**Programador responsable:** Jairo Pascagaza, Yeimy Ángel

**Descripción:** Recolección de información de la asociación de ganaderos

Asoagroproleche

---

**Validación:**

- Se debe establecer contacto con los miembros administrativos de la asociación.
- El lenguaje utilizado debe ser entendible fácilmente.

---

*Nota.* Se muestra los datos que se requieren para poder ser usuario de AsoVapp.

**Resumen sprint 1 (review, retrospective, planning)**

Al final del primer sprint se logró recolectar información relevante de los procesos administrativos actuales con los cuales se maneja la asociación, además se establecieron los primeros contactos con integrantes de la asociación que no pertenecen a la rama administrativa.

**Desarrollo Sprint 2**

Para este sprint se acordó redactar, diseñar y aplicar una entrevista a los diferentes integrantes de la asociación de ganaderos Asoagroproleche.

La siguiente tabla contiene la información acerca de las características principales que se definieron para la realización del sprint.

**Tabla 10***Historia de usuario – Elaboración entrevistas*


---

<b>Historia de usuario 02</b>	
<b>Número: 02</b>	<b>Usuario:</b> Presidente, administradora de tanque, socio
<b>Nombre de historia:</b> Elaboración de entrevistas	
<b>Prioridad:</b> Alta	<b>Riesgo en desarrollo:</b> Medio
<b>Programador responsable:</b> Jairo Pascagaza, Yeimy Ángel	
<b>Descripción:</b> Diseño y aplicación de las entrevistas para conocer la opinión y preferencia de los integrantes de la asociación con respecto a los procesos administrativos que se llevan, además de la opinión de estos acerca del uso de la tecnología.	
<b>Validación:</b>	
<ul style="list-style-type: none"> <li>• Se debe establecer contacto con todos los integrantes de la asociación.</li> <li>• El lenguaje utilizado debe ser entendible fácilmente.</li> <li>• La entrevista debe ser realizada de manera virtual</li> <li>• Las entrevistas deben ser para cada rol de la asociación (Presidente, Administradora de tanque, Socio)</li> </ul>	

---

*Nota.* Se muestra los datos que se requieren para poder ser usuario de AsoVapp.

**Resumen sprint 2 (review, retrospective, planning)**

Al final del segundo sprint se lograron realizar las entrevistas a todos los integrantes de la asociación de ganaderos Asoagroproleche, se aplicó de manera virtual a través de la herramienta “Google forms”, y se aplicó una entrevista diferente para cada rol dentro de la asociación.

**Desarrollo Sprint 3**

Para este sprint se acordó redactar, diseñar y aplicar una entrevista a los diferentes integrantes de la asociación de ganaderos Asoagroproleche.

La siguiente tabla contiene la información acerca de las características principales que se definieron para la realización del sprint.

**Tabla 11***Historia de usuario –Análisis de la información*


---

<b>Historia de usuario 03</b>	
<b>Número: 03</b>	<b>Usuario:</b> Presidente, administradora de tanque, socio
<b>Nombre de historia:</b> Análisis de la información	
<b>Prioridad:</b> Alta	<b>Riesgo en desarrollo:</b> Medio
<b>Programador responsable:</b> Jairo Pascagaza, Yeimy Ángel	
<b>Descripción:</b> Análisis de las entrevistas realizadas a cada uno de los miembros de la asociación	
<b>Validación:</b>	
<ul style="list-style-type: none"> <li>• Se debe extraer la información de las entrevistas</li> <li>• La información debe tabularse</li> <li>• Se deben generar diagramas que permitan realizar un análisis.</li> </ul>	

---

*Nota.* Se muestra los datos que se requieren para poder ser usuario de AsoVapp.

### **Resumen sprint 3 (review, retrospective, planning)**

Al final del tercer sprint se logró obtener la información y tabulación de los datos, con lo cual se pudo obtener los porcentajes de cada una de las preguntas formuladas, y posteriormente se hizo un análisis de esa información con el fin de establecer y sacar conclusiones para continuar con la formulación de los requerimientos.

### **Desarrollo Sprint 4**

Para este sprint se acordó realizar el proceso de levantamiento de requerimientos funcionales y no funcionales a partir del análisis y la información recolectada en las entrevistas y visitas a la asociación.

La siguiente tabla contiene la información acerca de las características principales que se definieron para la realización del sprint.

**Tabla 12**

*Historia de usuario - Requerimientos*

---

<b>Historia de usuario 04</b>	
<b>Número: 04</b>	<b>Usuario:</b> Presidente, administradora de tanque, socio
<b>Nombre de historia:</b> Levantamiento de requerimientos	
<b>Prioridad:</b> Alta	<b>Riesgo en desarrollo:</b> Medio
<b>Programador responsable:</b> Jairo Pascagaza, Yeimy Ángel	
<b>Descripción:</b> Elaboración de los requerimientos del sistema.	
<b>Validación:</b>	
<ul style="list-style-type: none"> <li>• Se debe utilizar conforme con el estándar IEEE Std 830-1998.</li> <li>• Se deben definir requerimientos funcionales y no funcionales.</li> <li>• Se deben diseñar a partir de la información recolectada en las entrevistas.</li> </ul>	

---

*Nota.* Se muestra los datos que se requieren para poder ser usuario de AsoVapp.

### **Resumen sprint 4 (review, retrospective, planning)**

Al final del cuarto sprint se logró realizar el levantamiento de requerimientos, en los cuales se definieron las funciones y las directrices con las cuales se debe realizar el desarrollo del sistema de información.

### **Desarrollo Sprint 5**

Para este sprint se acordó el diseño de los mockups para los diferentes roles que se manejan en la asociación, con los cuales se llevara a cabo el proceso de desarrollo teniéndolos como base para la maquetación.

La siguiente tabla contiene la información acerca de las características principales que se definieron para la realización del sprint.

**Tabla 13**

*Historia de usuario - Mockups*

---

<b>Historia de usuario 05</b>	
<b>Número: 05</b>	<b>Usuario:</b> Presidente, administradora de tanque, socio
<b>Nombre de historia:</b> Mockups	
<b>Prioridad:</b> Alta	<b>Riesgo en desarrollo:</b> Medio
<b>Programador responsable:</b> Jairo Pascagaza, Yeimy Ángel	
<b>Descripción:</b> Diseños preliminares de los diferentes módulos que componen el sistema.	
<b>Validación:</b>	
<ul style="list-style-type: none"> <li>• Se debe usar principalmente colores blanco y verde.</li> <li>• El lenguaje utilizado debe ser entendible fácilmente.</li> <li>• Se debe utilizar el logo y fotografías acerca de producción lechera.</li> </ul>	

---

*Nota.* Se muestra los datos que se requieren para poder ser usuario de AsoVapp.

### **Resumen sprint 5 (review, retrospective, planning)**

Al final del quinto sprint se logró diseñar los diferentes módulos que debe contener el sistema, esto gracias al software balsamiq, que brinda una serie de herramientas que simulan interacciones básicas del usuario con la plataforma. Además, se logró el desarrollo de la página principal donde se muestra información acerca del sistema y sus funciones.

## Desarrollo Sprint 6

Para este sprint se acordó el diseño de la base de datos teniendo en cuenta las entidades y relaciones definidas para el sistema y sus diferentes roles.

La siguiente tabla contiene la información acerca de las características principales que se definieron para la realización del sprint.

**Tabla 14**

*Historia de usuario – Base de datos*

---

<b>Historia de usuario 06</b>	
<b>Número: 06</b>	<b>Usuario:</b> presidente, administradora de tanque, socio
<b>Nombre de historia:</b> Base de datos	
<b>Prioridad:</b> Alta	<b>Riesgo en desarrollo:</b> Medio
<b>Programador responsable:</b> Jairo Pascagaza, Yeimy Ángel	
<b>Descripción:</b> Diseño de la base de datos a partir de las entidades y relaciones identificadas en la asociación.	
<b>Validación:</b>	
<ul style="list-style-type: none"> <li>• Se debe usar lenguaje SQL.</li> <li>• Se debe parametrizar la base de datos.</li> <li>• Se debe tener en cuenta las llaves principales y foráneas.</li> </ul>	

---

*Nota.* se muestra los datos que se requieren para poder ser usuario de AsoVapp.

## Resumen sprint 6 (review, retrospective, planning)

Al final del sexto sprint se logró modelar la base de datos SQL, tomando como referencia las entidades y relaciones que se lograron identificar en la asociación de ganaderos, este proceso

se realizó con el lenguaje SQL y la utilización de phpmyadmin para la creación de la base de datos.

### **Desarrollo Sprint 7**

Para este sprint se acordó la creación de las interfaces principales correspondientes al registro e inicio de sesión, en las cuales se deben validar los datos y roles que se definen en el diseño del sistema, y con los cuales los usuarios tendrán designadas una serie de funciones dependiendo de su rol dentro de la asociación. Además del desarrollo de los submódulos de consulta de socios y estadísticas individuales correspondiente al rol de presidente.

La siguiente tabla contiene la información acerca de la historia de usuario en la cual se especifican las características a tener en cuenta para el desarrollo de los módulos.

**Tabla 15***Historia de usuario – Módulo inicio sesión*


---

<b>Historia de usuario 07</b>	
<b>Número: 07</b>	<b>Usuario:</b> Presidente, administradora de tanque, socio
<b>Nombre de historia:</b> Módulos de inicio de sesión y registro	
<b>Prioridad:</b> Alta	<b>Riesgo en desarrollo:</b> Medio
<b>Programador responsable:</b> Jairo Pascagaza, Yeimy Ángel	
<b>Descripción:</b> Módulos que permitan el registro de datos correspondientes a los miembros de la asociación de ganaderos.	
<b>Validación:</b>	
<ul style="list-style-type: none"> <li>• Se debe usar principalmente colores blanco y verde.</li> <li>• El lenguaje utilizado debe ser entendible fácilmente.</li> <li>• La información debe ser almacenada de manera segura.</li> <li>• Todos los campos de los formularios deben se validados.</li> </ul>	

---

*Nota.* Se muestra los datos que se requieren para poder ser usuario de AsoVapp.

En la siguiente historia de usuario se describen las validaciones y características que se tomaron en cuenta para la ejecución del desarrollo planteado.

**Tabla 16***Historia de usuario – Listado socios*


---

<b>Historia de usuario 07</b>	
<b>Número: 07</b>	<b>Usuario:</b> Presidente
<b>Nombre de historia:</b> Listado de socios e información individual	
<b>Prioridad:</b> Alta	<b>Riesgo en desarrollo:</b> Medio
<b>Programador responsable:</b> Jairo Pascagaza, Yeimy Ángel	
<b>Descripción:</b> Visualizar una lista de los socios registrados en el sistema y seleccionar de manera individual, para consultar sus estadísticas de producción y documentos reglamentarios de la finca.	
<b>Validación:</b>	
<ul style="list-style-type: none"> <li>• Se debe configurar el menú de opciones para el rol de presidente.</li> <li>• La lista de los socios y la interfaz en general debe ser adaptable a diferentes dispositivos.</li> <li>• La información individual de cada socio debe contener los datos personales, la producción y los certificados de vacunación de los animales.</li> <li>• Las estadísticas deben mostrar un gráfico a partir de un rango de fechas seleccionadas por el presidente</li> <li>• Los documentos que se pueden mostrar deben ser cargados previamente por el rol de socio.</li> </ul>	

---

*Nota.* Se muestra los datos que se requieren para poder ser usuario de AsoVapp.

### **Resumen sprint 7 (review, retrospective, planning)**

Se logró desarrollar los módulos de inicio de sesión y registro teniendo en cuenta el MVC (Modelo, Vista, Controlador), utilizando como editor de código a Visual Studio Code, y sincronizándolo con git, de esta manera se trabajó de forma independiente de acuerdo a los horarios disponibles por los integrantes del equipo de desarrollo. Además, se definieron los roles de usuario y los permisos de cada uno de estos al ingresar al sistema.

## **Desarrollo Sprint 8**

Para este sprint se acordó el desarrollo de las funciones del presidente, específicamente las de registrar información de proyectos, y los ajustes del sistema.

En la siguiente historia de usuario se muestra la información en la cual se especifican las características a tener en cuenta para el desarrollo de las funciones descritas anteriormente incluyendo las acciones que tiene el usuario y las validaciones que realiza el sistema para ejecutar las funciones correctamente.

**Tabla 8***Historia de usuario- Registro de proyectos*


---

**Historia de usuario 08**

---

<b>Número: 08</b>	<b>Usuario:</b> Presidente
<b>Nombre de historia:</b> Registrar proyecto	
<b>Prioridad:</b> Alta	<b>Riesgo en desarrollo:</b> Medio
<b>Programador responsable:</b> Jairo Pascagaza, Yeimy Ángel	
<b>Descripción:</b> Sección que permita ingresar información de los proyectos en los cuales participa la asociación y permita cargar documentos de soporte.	

---

**Validación:**

- Para el frontend se debe usar principalmente colores blanco y verde.
- Se debe visualizar un listado de los proyectos registrados
- Al registrar un proyecto se deben diligenciar varios campos obligatorios
- El cargue de documentos es opcional
- Se debe mostrar un submenú que permita editar o eliminar el proyecto.

---

*Nota.* Se muestra los datos que se requieren para poder ser usuario de AsoVapp.

En la siguiente historia de usuario se describen las validaciones y características que se tomaron en cuenta para la ejecución del desarrollo planteado.

**Tabla 17***Historia de usuario - Ajustes*


---

<b>Historia de usuario 08</b>	
<b>Número: 08</b>	<b>Usuario:</b> Presidente
<b>Nombre de historia:</b> Ajustes del perfil	
<b>Prioridad:</b> Alta	<b>Riesgo en desarrollo:</b> Medio
<b>Programador responsable:</b> Jairo Pascagaza, Yeimy Ángel	
<b>Descripción:</b> Acceder a la información del perfil y tener la posibilidad de actualizar o modificar los datos registrados, además de poder cambiar la contraseña.	
<b>Validación:</b>	
<ul style="list-style-type: none"> <li>• Se mostrará la información que está registrada en el sistema con la opción de editarla</li> <li>• Para hacer una actualización de la información se debe proporcionar la contraseña.</li> <li>• Para cambiar la contraseña se debe ingresar dos veces para verificarla, además se debe proporcionar la contraseña actual</li> </ul>	

---

*Nota.* Se muestra los datos que se requieren para poder ser usuario de AsoVapp.

### **Resumen sprint 8(review, retrospective, planning)**

En el sprint ocho se logró desarrollar varias funciones para el rol de presidente, en las cuales puede ingresar información, eliminar información, subir documentos y editar información, debido a la complejidad en el desarrollo de algunas funciones, este sprint tomó un poco más de tiempo que los anteriores, por lo que el scrum master redujo la cantidad de tareas para el siguiente sprint.

### **Desarrollo Sprint 9**

Para este sprint se acordó el desarrollo de las funciones del administrador de tanque, específicamente las de generar una lista de los socios pertenecientes a Asoagroproleche y la función de registrar la producción de cada socio.

En la siguiente historia de usuario se muestra la información en la cual se especifican las características a tener en cuenta para el desarrollo de las funciones descritas anteriormente incluyendo las acciones que tiene el usuario y las validaciones que realiza el sistema para ejecutar las funciones correctamente.

**Tabla 18***Historia de usuario – Registro producción*


---

<b>Historia de usuario 09</b>	
<b>Número: 09</b>	<b>Usuario:</b> Administrador
<b>Nombre de historia:</b> Listado de socios y registro de producción	
<b>Prioridad:</b> Alta	<b>Riesgo en desarrollo:</b> Medio
<b>Programador responsable:</b> Jairo Pascagaza, Yeimy Ángel	
<b>Descripción:</b> Visualizar una lista de los socios registrados en el sistema y seleccionar de manera individual, para registrar la cantidad de producción lechera de cada socio.	
<b>Validación:</b>	
<ul style="list-style-type: none"> <li>• Se debe configurar el menú de opciones para el rol de administrador.</li> <li>• La lista de los socios y la interfaz en general debe ser adaptable a diferentes dispositivos.</li> <li>• La información individual de cada socio debe contener los datos personales y la producción</li> <li>• Los datos de registro de producción deben ser controlados</li> <li>• Los colores de la interfaz deben ser blanco y verde</li> </ul>	

---

*Nota.* Se muestra los datos que se requieren para poder ser usuario de AsoVapp.

**Resumen sprint 9(review, retrospective, planning)**

En el noveno sprint se logró desarrollar varias funciones para el rol de administrador de tanque, en las cuales puede ingresar o registrar la información correspondiente a la producción individual de cada socio, eliminar información, editar información, además se realizó una reutilización de código para el desarrollo del submódulo que se encarga de listar los socios.

### **Desarrollo Sprint 10**

Para este sprint se acordó el desarrollo de las funciones del administrador de tanque, específicamente la de mostrar la información de producción de cada socio, y el cálculo de pago que se le debe realizar a cada uno.

En la siguiente historia de usuario se muestra la información en la cual se especifican las características a tener en cuenta para el desarrollo de las funciones descritas anteriormente incluyendo las acciones que tiene el usuario y las validaciones que realiza el sistema para ejecutar las funciones correctamente.

**Tabla 19***Historia de usuario- Cálculos de pagos*


---

<b>Historia de usuario 10</b>	
<b>Número: 10</b>	<b>Usuario:</b> Administrador
<b>Nombre de historia:</b> Registros y cálculo de pagos	
<b>Prioridad:</b> Alta	<b>Riesgo en desarrollo:</b> Medio
<b>Programador responsable:</b> Jairo Pascagaza, Yeimy Ángel	
<b>Descripción:</b> Visualizar una lista con la información individual de cada socio donde se pueda observar los registros de producción a partir de un rango de fechas.	
<b>Validación:</b>	
<ul style="list-style-type: none"> <li>• Se debe configurar el menú de opciones para el rol de administrador.</li> <li>• La lista de registros y la interfaz en general debe ser adaptable a diferentes dispositivos.</li> <li>• Los datos de registro de producción deben ser controlados</li> <li>• Los colores de la interfaz deben ser blanco y verde</li> <li>• Se deben configurar opciones para exportar la información</li> </ul>	

---

*Nota.* Se muestra los datos que se requieren para poder ser usuario de AsoVapp.

### **Resumen sprint 10(review, retrospective, planning)**

En el décimo sprint se logró desarrollar varias funciones para el rol de administrador de tanque, en las cuales puede hacer consultas de registros de producción a partir de un rango de fechas establecidas por el usuario, y a partir de esos registros se hace el proceso de cálculo de pago por la producción.

### **Desarrollo Sprint 11**

Para este sprint se acordó el desarrollo de las funciones del administrador de tanque, específicamente registrar las entregas de producción lechera que realiza la asociación a las empresas recolectoras, además se acordó el desarrollo de una sección específicamente para el rol de administrador de tanque en la cual se puedan subir o cargar archivos.

En la siguiente historia de usuario se muestra la información en la cual se especifican las características a tener en cuenta para el desarrollo de las funciones descritas anteriormente incluyendo las acciones que tiene el usuario y las validaciones que realiza el sistema para ejecutar las funciones correctamente.

**Tabla 20**

*Historia de usuario – Cargue de archivos*

---

<b>Historia de usuario 11</b>	
<b>Número: 11</b>	<b>Usuario:</b> Administrador
<b>Nombre de historia:</b> Registro de entregas y espacio para cargue de archivos	
<b>Prioridad:</b> Alta	<b>Riesgo en desarrollo:</b> Medio
<b>Programador responsable:</b> Jairo Pascagaza, Yeimy Ángel	
<b>Descripción:</b> Se debe proporcionar un formulario para registrar la información correspondiente a las entregas de producción que hace la asociación a las empresas recolectoras, además se debe habilitar un espacio para cargar archivos relacionados con cada empresa recolectora.	
<b>Validación:</b>	
<ul style="list-style-type: none"> <li>• Se debe validar el tipo de datos que se registran desde el formulario, con el fin de evitar errores y conflictos con otras funciones del sistema</li> <li>• Para cargar archivos es necesario verificar la extensión para que solo permita .pdf o .doc, esto con el fin de evitar que se carguen archivos peligrosos o de origen dudoso que puedan afectar la seguridad del sistema</li> <li>• Los formularios deben estar alineados con el diseño que se ha venido utilizando en el resto del sistema.</li> </ul>	

*Nota.* Se muestra los tipos de datos que se requieren en los formularios.

### **Resumen sprint 11(review, retrospective, planning)**

En el undécimo sprint se desarrolló de manera satisfactoria las tareas planteadas en el inicio, se logró crear la interfaz para que el administrador de tanque registre la información correspondiente, y el cargue de archivos. Se tuvo en cuenta las validaciones para la correcta

integración con las demás funcionalidades del sistema

### **Desarrollo Sprint 12**

Para este sprint se acordó el desarrollo de las funciones del administrador de tanque, específicamente la creación de un panel principal en el cual se muestre un resumen de la información correspondiente a la producción general de la asociación, se deben mostrar los datos correspondientes a un rango de fechas que el usuario puede manipular.

En la siguiente historia de usuario se muestra la tanto la descripción como las verificaciones que se tuvieron en cuenta para el correcto desarrollo de las funcionalidades planteadas anteriormente.

**Tabla 21***Historia de usuario – Panel de control*


---

<b>Historia de usuario 12</b>	
<b>Número:</b> 12	<b>Usuario:</b> Administrador
<b>Nombre de historia:</b> Panel de control	
<b>Prioridad:</b> Alta	<b>Riesgo en desarrollo:</b> Medio
<b>Programador responsable:</b> Jairo Pascagaza, Yeimy Ángel	
<b>Descripción:</b> Se debe desarrollar una interfaz en la cual se muestre un resumen de la producción e ingresos de la asociación, se debe tener en cuenta los litros recolectados y los litros entregados a las empresas recolectoras, así como el aporte de producción en porcentaje de cada uno de los socios.	
<b>Validación:</b>	
<ul style="list-style-type: none"> <li>• Se debe validar el rango de fechas iniciales y finales con el fin de obtener la información de la base de datos de forma correcta.</li> <li>• Se debe tener en cuenta una serie de datos que son de suma importancia para la asociación como la producción y entrega a empresas recolectoras.</li> <li>• El diseño debe ajustarse al utilizado en el resto del sistema respetando colores y tipo de fuente.</li> <li>• Se debe tener en cuenta que el panel sea adaptable a diferentes pantallas, debido a que puede ser visto desde dispositivos móviles.</li> </ul>	

---

*Nota.* Se muestra historia de la administradora de tanque.

### **Resumen sprint 12(review, retrospective, planning)**

En el duodécimo sprint se tuvieron inconvenientes al integrar la información en una sola pantalla, especialmente para la versión móvil, sin embargo, se lograron solucionar utilizando bootstrap para adaptarlo de manera correcta. Debido a dificultades mencionadas anteriormente, se asignó el doble de tiempo para esta actividad con el fin de realizarla de manera correcta.

En general se logró el desarrollo del panel propuesto al inicio del sprint con el fin de

mostrar información relevante al administrador de tanque.

### **Desarrollo Sprint 13**

Para este sprint se acordó el desarrollo de las funciones del administrador de tanque, principalmente para el registro y visualización de los implementos y herramientas que se utilizan en las labores diarias dentro de las instalaciones de la asociación, como pueden ser cantinas, tanque, medidores etc.

En la siguiente historia de usuario se muestra la tanto la descripción como las verificaciones que se tuvieron en cuenta para el correcto desarrollo de las funcionalidades planteadas anteriormente.

**Tabla 22***Historia de usuario – Registros materiales*


---

<b>Historia de usuario 13</b>	
<b>Número: 13</b>	<b>Usuario:</b> Administrador
<b>Nombre de historia:</b> Registro de recursos materiales	
<b>Prioridad:</b> Alta	<b>Riesgo en desarrollo:</b> Medio
<b>Programador responsable:</b> Jairo Pascagaza, Yeimy Ángel	
<b>Descripción:</b> Se debe desarrollar una interfaz que permita registrar y visualizar información acerca de los recursos materiales de la asociación.	
<b>Validación:</b>	
<ul style="list-style-type: none"> <li>• Es necesario verificar los datos que ingresa el usuario con el fin de evitar errores en el sistema.</li> <li>• Es necesario habilitar un espacio para subir imágenes de cada recurso material que se registre.</li> <li>• El diseño debe ajustarse al utilizado en el resto del sistema respetando colores, tipo de fuente y adaptabilidad a diferentes pantallas.</li> </ul>	

---

*Nota.* Se muestra como se debe desarrollar el registro de materiales de la asociación.

### **Resumen sprint 13(review, retrospective, planning)**

En este sprint se logró desarrollar la funcionalidad propuesta sin demasiadas complicaciones, debido a que anteriormente se había desarrollado algo similar que se logró adaptar a lo requerido en este sprint. Es necesario recalcar que se tuvieron en cuenta las diferentes validaciones necesarias para la correcta integración con el sistema.

### **Desarrollo Sprint 14**

Para este sprint se acordó el inicio del desarrollo del módulo de socio, se debe

implementar la funcionalidad para que cada socio registre los animales de su finca. Los registros deben visualizarse en una interfaz diferente a la del registro.

En la siguiente historia de usuario se muestra la tanto la descripción como las verificaciones que se tuvieron en cuenta para el correcto desarrollo de las funcionalidades planteadas anteriormente.

### Tabla 23

#### *Historia de usuario – Registro animales*

---

<b>Historia de usuario 14</b>	
<b>Número: 14</b>	<b>Usuario:</b> Socio
<b>Nombre de historia:</b> Registro de animales	
<b>Prioridad:</b> Alta	<b>Riesgo en desarrollo:</b> Medio
<b>Programador responsable:</b> Jairo Pascagaza, Yeimy Ángel	
<b>Descripción:</b> Se debe desarrollar una interfaz que permita registrar y visualizar información acerca de los animales que cada socio tiene en su finca.	
<b>Validación:</b>	
<ul style="list-style-type: none"> <li>• Es necesario verificar los datos que ingresa el usuario con el fin de evitar errores en el sistema.</li> <li>• Es necesario habilitar un espacio para subir imágenes de cada animal que se registre.</li> <li>• Es necesario que el software calcule la edad de cada animal a partir de la fecha de nacimiento proporcionada por el usuario.</li> <li>• Es necesario que los registros y fotos se puedan eliminar y editar.</li> <li>• Es necesario reducir el peso de las imágenes subidas por el usuario con el fin de no utilizar demasiado almacenamiento, además para agilizar la carga de la plataforma.</li> <li>• El diseño debe ajustarse al utilizado en el resto del sistema respetando colores, tipo de fuente y adaptabilidad a diferentes pantallas.</li> </ul>	

---

*Nota.* Se muestra información que se requiere para poder ser usuario de AsoVapp.

**Resumen sprint 14(review, retrospective, planning)**

En este sprint se logró desarrollar la funcionalidad propuesta sin demasiadas complicaciones, debido a que anteriormente se había desarrollado algo similar (Historia de usuario número 14) que se logró adaptar a lo requerido en este sprint. Es necesario recalcar que se tuvieron en cuenta las diferentes validaciones necesarias para la correcta integración con el sistema.

**Desarrollo Sprint 15**

Para este sprint se acordó desarrollo de las funcionalidades para mostrar la información acerca de la producción que se ha aportado a la asociación.

En la siguiente historia de usuario se muestra la tanto la descripción como las verificaciones que se tuvieron en cuenta para el correcto desarrollo de las funcionalidades planteadas anteriormente.

**Tabla 24**

Historia de usuario – Visualización de producción

---

<b>Historia de usuario 15</b>	
<b>Número:</b> 15	<b>Usuario:</b> Socio
<b>Nombre de historia:</b> Visualización de producción.	
<b>Prioridad:</b> Alta	<b>Riesgo en desarrollo:</b> Medio
<b>Programador responsable:</b> Jairo Pascagaza, Yeimy Ángel	
<b>Descripción:</b> Se debe desarrollar una interfaz que permita visualizar los registros de producción que haya hecho la asociación de ganaderos para un socio en específico, se debe tener en cuenta que el usuario proporcionará un rango de fechas las cuales servirán como filtro para obtener la información correspondiente.	
<b>Validación:</b>	
<ul style="list-style-type: none"> <li>• Es necesario verificar las fechas proporcionadas por el usuario con el fin de evitar errores en el sistema</li> <li>• Se debe limitar la acción del usuario a solo visualización sin posibilidad de editar o eliminar registros debido a que estas acciones solo puede realizarlas el administrador de tanque.</li> </ul>	

---

*Nota.* Se muestra información que se requiere para el módulo del socio.

**Resumen sprint 15(review, retrospective, planning)**

En este sprint se logró desarrollar de manera satisfactoria las funcionalidades propuestas anteriormente, debido a que previamente se desarrolló una funcionalidad similar para el rol de administrador de tanque los tiempos se redujeron considerablemente y el tiempo de sobra se utilizó para corrección de errores de algunas funcionalidades del sistema.

**Desarrollo Sprint 16**

Para este sprint se acordó el desarrollo de las funcionalidades para realizar el cargue de

archivos por parte del socio o ganadero, correspondiente a los certificados de vacunación de los animales y al registro de finca. Estos documentos deben ser visualizados por el rol de presidente.

En la siguiente historia de usuario se muestra la tanto la descripción como las verificaciones que se tuvieron en cuenta para el correcto desarrollo de las funcionalidades planteadas anteriormente.

## Tabla 25

### *Historia de usuario – Visualización de archivos*

---

<b>Historia de usuario 16</b>	
<b>Número: 16</b>	<b>Usuario:</b> Socio
<b>Nombre de historia:</b> Cargue y visualización de archivos	
<b>Prioridad:</b> Alta	<b>Riesgo en desarrollo:</b> Medio
<b>Programador responsable:</b> Jairo Pascagaza, Yeimy Ángel	
<b>Descripción:</b> Se debe desarrollar una interfaz que permita al usuario cargar archivos correspondientes a la vacunación de los animales y al registro de finca.	
<b>Validación:</b>	
<ul style="list-style-type: none"> <li>• Es necesario verificar los archivos subidos por el usuario con el fin de evitar que se suban archivos que comprometan la seguridad del sistema</li> <li>• Es necesario que los registros se puedan editar o eliminar</li> <li>• Es necesario mostrar una barra de carga que permita visualizar el porcentaje subido del archivo.</li> <li>• El diseño que se ha tenido en cuenta para los demás módulos se debe respetar, incluyendo la versión móvil.</li> </ul>	

---

*Nota.* Se muestra información que se requiere para el módulo del socio.

### **Resumen sprint 16(review, retrospective, planning)**

En este sprint se logró desarrollar de manera satisfactoria las funcionalidades propuestas anteriormente, debido a que previamente se desarrolló una funcionalidad similar para el rol de administrador de tanque los tiempos se redujeron considerablemente y el tiempo de sobra se utilizó para agregar pantallas de carga en las funcionalidades que requieren la subida de documentos o archivos, con el fin de brindar una ayuda gráfica a los usuarios.

### **8.2. Metodología de investigación**

Para este proyecto se utiliza la metodología mixta la cual se aplican de manera simultánea los métodos cualitativo y cuantitativo, se realiza mediante el proceso secuencial en la modalidad de la derivativa que es donde se mezclan los resultados que se adquieren de la asociación Asoagroproleche, se hizo una visita a las instalaciones del lugar donde se usaron los instrumentos como la de entrevista al presidente, la administradora de tanque y a los socios para saber que se requiere para el desarrollo del sistema AsoVapp, después se prepararon preguntas abiertas para recibir las opiniones y sugerencias de cómo les gustaría que se construyera el sistema de información y en el cuantitativo se realizaron preguntas abiertas que permiten conocer cuántos son los integrantes de la asociación y saber que datos y cifras se manejan en el recaudo de la leche.

**Tabla 26***Fases de la metodología mixta*

<b>Fase I</b>	Planteamiento del problema	Desconocen los procesos
<b>Fase II</b>	Diseño de investigación	Realizan una sección de preguntas para entrevistas.
<b>Fase III</b>	Muestreo	Se identifican las estadísticas de los datos.
<b>Fase IV</b>	Recolección de datos	Se realiza las respectivas entrevistas.
<b>Fase V</b>	Análisis de los datos y resultados	Se identifican los requerimientos para realizar el sistema.

*Nota.* Se muestran las fases en las que se realizó la investigación en la asociación Asoagroproleche.

Debido a que no se conocen los procesos o actividades que realiza Asoagroproleche se plantea inicialmente la elaboración de unas entrevistas que constan tanto de preguntas abiertas como de selección para luego aplicarlas a cada uno de los integrantes como lo es el presidente, administradora y cada uno de los socios, para así obtener cada uno de los datos e información de cómo es el uso de las tecnologías en el ámbito de las labores diarias tanto de los administrativos como de los ganaderos y también de los efectos que llevarían a la de la ejecución de un sistema de información para centralizar los datos pertinentes para así lograr beneficio mutuo en esta empresa, finalmente poder conseguir cada uno de los requerimientos que se necesitan para el desarrollo del sistema.

Teniendo en cuenta la información inicial recolectada en la entrevista se realizaron tres entrevistas en Google Forms cada una definida a una sección diferente de la entidad con el fin de conocer la información necesaria para plantear el desarrollo y funcionamiento de cada uno de los

módulos principales del sistema y a su vez poder identificar los requerimientos específicos para cada uno.

La primera entrevista se realizó al presidente de esta organización ya que se identificó que él se encarga del manejo de los proyectos en lo que participan como asociación, para esta entrevista se plantearon cinco preguntas las cuales brindan la información requerida para el desarrollo del módulo inicial del sistema denominado manejo de proyectos e información de los asociados.

La segunda entrevista fue realizada a la encargada de administrar el tanque recolector de leche en el acopio de la asociación, ya que esta es la sección principal de la empresa en la que se realiza el proceso de recolectar el producto y venderlo a una empresa productora y comercializadora de derivados lácteos, se diseñaron cinco preguntas las cuales permiten obtener los datos requeridos por los desarrolladores para realizar el módulo denominado Administración del tanque.

Por último, la tercera entrevista se realizó a los productores de leche asociados a la entidad, es decir, los ganaderos los cuales tienen un proceso interno de administración y manejo de cada una de sus fincas y sus animales, Para esta entrevista se realizaron siete preguntas en las cuales se recolecto la información esencial para diseñar y desarrollar el módulo del sistema denominado Ganaderos

Recopilados los resultados de las entrevistas estadísticas, se realizó un plan de análisis de datos, en el cual, se clasifica, controla y analiza la información, para de esta manera verificar que el aplicativo sea aceptable por los integrantes de la asociación, luego se procede a tabular los datos obtenidos en las diferentes entrevistas para obtener los porcentajes por cada una de las preguntas formuladas. Se validó la calidad de los datos para minimizar el sesgo y optimizar el

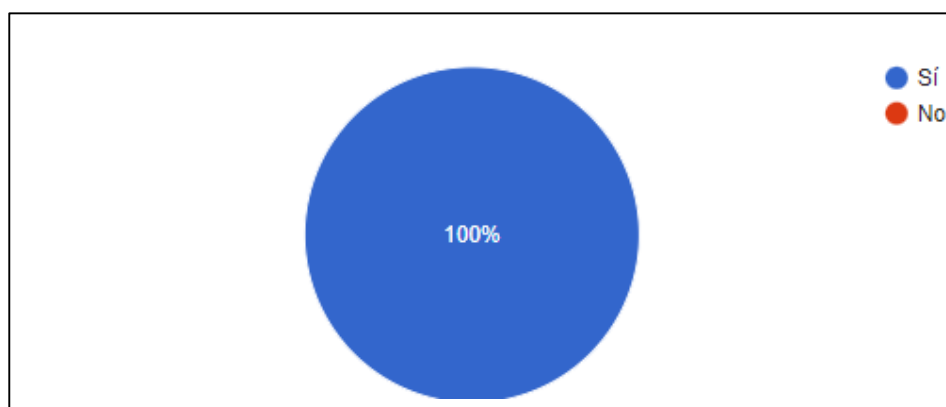
análisis de estos, finalmente se llevó a cabo el proceso de sacar conclusiones para continuar con la formulación de requerimientos y desarrollo del proyecto.

Una de las preguntas claves que se realizaron fue:

¿Cree que la tecnología le aportaría al desarrollo de la asociación?

#### **Figura 4**

*Evidencia de respuesta*



*Nota.* Se muestra la evidencia de la respuesta del presidente.

Para visualizar más de los resultados de cada una de las entrevistas ver anexos.

El sistema de información en la asociación ganadera de acopio de leche Asoagroproleche permite mejorar la gestión de cada uno de los datos, control de calidad, organización de cada uno de los elementos que usan y apoyar en la toma de decisiones basadas en datos concretos. Esto contribuye a un trabajo más eficiente, con mayor control y una gestión estratégica.

### **8.3. Ingeniería del proyecto**

A continuación, se contará con una clasificación y descripción de los distintos requerimientos esenciales tanto funcionales como no funcionales, identificando en su desarrollo como se han implementado porque es fundamental mantener el proyecto en un alto alcance cumpliendo todas sus exigencias y expectativas del usuario final.

Se diseñaron los requerimientos funcionales por cada módulo del sistema de información y los no funcionales del sistema en general.

#### **8.3.1. *Requerimientos Funcionales***

Los requerimientos funcionales se dividen por cada uno de los módulos con los que cuenta el aplicativo. Para el levantamiento de estos requerimientos se realizaron una cantidad de entrevistas a los miembros de la asociación para identificar las funciones y características que debe tener el sistema.

Tabla 27

*Requerimiento Funcional 1*

<b>Identificación del requerimiento:</b>	RF01
<b>Nombre del Requerimiento:</b>	Autenticación de Usuario.
<b>Características:</b>	Los usuarios deberán proporcionar un usuario y una contraseña para poder ingresar al sistema.
<b>Descripción del requerimiento:</b>	El sistema solo permitirá acceso a los módulos y funcionalidades dependiendo del rol al que pertenezca el usuario.
<b>Requerimiento NO funcional:</b>	<ul style="list-style-type: none"> <li>• RNF01</li> <li>• RNF02</li> <li>• RNF05</li> <li>• RNF08</li> </ul>
<b>Prioridad del requerimiento:</b>	Alta

*Nota.* Evidencia de la ficha del requerimiento de autenticación del usuario del sistema AsoVapp.

Tabla 28

*Requerimiento Funcional 2*

<b>Identificación del requerimiento:</b>	RF02
<b>Nombre del Requerimiento:</b>	Registrar Usuarios.
<b>Características:</b>	Los usuarios deberán registrarse en el sistema para acceder a cualquier parte del sistema.
<b>Descripción del requerimiento:</b>	El sistema permitirá al usuario (Presidente, socio y Administrador) registrarse. El usuario debe proporcionar datos como: Nombre, Apellido, Teléfono, Correo electrónico y Contraseña.
<b>Requerimiento NO funcional:</b>	<ul style="list-style-type: none"> <li>• RNF01</li> <li>• RNF02</li> <li>• RNF05</li> <li>• RNF08</li> </ul>
<b>Prioridad del requerimiento:</b>	Alta

*Nota.* Evidencia de los requisitos para el registro de usuarios dependiendo el rol dentro del sistema.

Tabla 29

*Requerimiento Funcional 3*

<b>Identificación del requerimiento:</b>	RF03
<b>Nombre del Requerimiento:</b>	Consulta de estadísticas
<b>Características:</b>	El sistema ofrecerá al usuario (presidente) información estadística acerca de la producción general de la asociación.
<b>Descripción del requerimiento:</b>	Muestra los datos estadísticos de la producción lechera de la asociación, definiendo un rango de fechas para analizar, y muestra un gráfico de línea de las fechas y la producción de cada una.
<b>Requerimiento NO funcional:</b>	<ul style="list-style-type: none"> <li>• RNF01</li> <li>• RNF02</li> <li>• RNF05</li> <li>• RNF08</li> </ul>
<b>Prioridad del requerimiento:</b>	Alta

*Nota.* Se muestra las especificaciones del requisito para la consulta de estadísticas.

**Tabla 30***Requerimiento Funcional 4*

<b>Identificación</b>	RF04
<b>del requerimiento:</b>	
<b>Nombre del</b>	Consultar Información.
<b>Requerimiento:</b>	
<b>Características:</b>	El sistema ofrecerá al usuario (Presidente) información general acerca de los datos personales de cada socio.
<b>Descripción del requerimiento:</b>	<b>Consultar Instrucción del socio:</b> Muestra información general sobre los socios registrados en el sistema, los documentos de finca y vacunación de animales, además de una representación estadística de la producción individual de cada socio.
<b>Requerimiento</b>	<ul style="list-style-type: none"> <li>• RNF01</li> </ul>
<b>NO funcional:</b>	<ul style="list-style-type: none"> <li>• RNF02</li> </ul>
<b>Prioridad del requerimiento:</b>	
	Alta

*Nota.* Evidencia de las características requeridas para consultar información de cualquier socio.

**Tabla 31***Requerimiento Funcional 5*

<b>Identificación</b>	RF05
<b>del requerimiento:</b>	
<b>Nombre del Requerimiento:</b>	Registrar proyectos
<b>Características:</b>	El sistema ofrecerá al usuario(presidente) la funcionalidad de registrar proyectos en los cuales participa la asociación.
<b>Descripción del requerimiento:</b>	<b>Registrar proyectos:</b> Permite registrar la información pertinente a proyectos en los cuales participa la asociación, además brinda un espacio para subir la documentación y soportes que esto genera.
<b>Requerimiento</b>	<ul style="list-style-type: none"> <li>• RNF01</li> </ul>
<b>NO funcional:</b>	<ul style="list-style-type: none"> <li>• RNF02</li> </ul>
<b>Prioridad del requerimiento:</b>	Alta

*Nota.* Especificación de requisitos para registrar proyectos en los que ha participado la asociación.

**Tabla 32***Requerimiento Funcional 6*

<b>Identificación</b>	RF06
<b>del requerimiento:</b>	
<b>Nombre del Requerimiento:</b>	Consulta y edición de proyectos
<b>Características:</b>	El sistema ofrecerá al usuario(presidente) la funcionalidad de consultar y editar los proyectos en los cuales participa la asociación registrados previamente en el sistema.
<b>Descripción del requerimiento:</b>	<b>Registrar proyectos:</b> Permite consultar un listado con los proyectos que se registraron en el sistema, y la documentación perteneciente a cada uno de estos, además de modificar o complementar la información de cada uno de estos.
<b>Requerimiento NO funcional:</b>	<ul style="list-style-type: none"> <li>• RNF01</li> <li>• RNF02</li> </ul>
<b>Prioridad del requerimiento:</b>	Alta

*Nota.* Se muestra la descripción del requisito para consultar y editar proyectos en los que ha participado la entidad.

**Tabla 33***Requerimiento Funcional 7*

<b>Identificación del requerimiento:</b>	RF07
<b>Nombre del Requerimiento:</b>	Modificar.
<b>Características:</b>	El sistema permitirá al administrador, presidente y ganadero modificar los datos personales.
<b>Descripción del requerimiento:</b>	<b>Modificar datos:</b> Permite a los usuarios del sistema modificar los datos personales registrados en este.
<b>Requerimiento NO funcional:</b>	<ul style="list-style-type: none"> <li>• RNF01</li> <li>• RNF02</li> </ul>
<b>Prioridad del requerimiento:</b>	Alta

*Nota.* Se evidencia las características para modificar datos personales.

Tabla 34

*Requerimiento Funcional 8*

<b>Identificación del requerimiento:</b>	RF08
<b>Nombre del Requerimiento:</b>	Registrar producción
<b>Características:</b>	Permitirá el registro de producción lechera.
<b>Descripción del requerimiento:</b>	Permite al administrador ingresar información correspondiente a la producción lechera de cada socio. Los datos que deben ser proporcionados son la cantidad de leche, la fecha y el volumen.
<b>Requerimiento NO funcional:</b>	<ul style="list-style-type: none"> <li>• RNF01</li> <li>• RNF02</li> <li>• RNF05</li> </ul>
<b>Prioridad del requerimiento:</b>	Alta

*Nota.* Se muestra la descripción para realizar el registro de producción lechera.

**Tabla 35***Requerimiento Funcional 9*

<b>Identificación</b>	RF09
<b>del requerimiento:</b>	
<b>Nombre del Requerimiento:</b>	Consultar producción.
<b>Características:</b>	Permite consultar los registros de producción.
<b>Descripción del requerimiento:</b>	<b>Consultar producción:</b> Permite al administrador una vez que haya accedido con su cuenta al sistema, consultar los registros de producción de cada uno de los socios.
<b>Requerimiento NO funcional:</b>	<ul style="list-style-type: none"> <li>• RNF01</li> <li>• RNF02</li> <li>• RNF05</li> <li>• RNF06</li> <li>• RNF07</li> <li>• RNF08</li> </ul>
<b>Prioridad del requerimiento:</b>	Alta
<i>Nota.</i> Descripción del requisito para consultar la producción lechera que se a recolectado.	

**Tabla 36***Requerimiento Funcional 10*


---

<b>Identificación</b>	RF10
<b>del requerimiento:</b>	
<b>Nombre del Requerimiento:</b>	Registrar animales.
<b>Características:</b>	Permite registrar información de los animales de cada socio.
<b>Descripción del requerimiento:</b>	<b>Registrar Animales</b> El socio o ganadero deberá suministrar información como raza, edad, documentos de vacunación.
<b>Requerimiento NO funcional:</b>	<ul style="list-style-type: none"> <li>• RNF01</li> <li>• RNF02</li> <li>• RNF05</li> <li>• RNF06</li> <li>• RNF07</li> <li>• RNF08</li> </ul>
<b>Prioridad del requerimiento:</b>	
	Alta

---

*Nota.* Se muestra la descripción del requerimiento para el registro de animales.

### 8.3.2. *Requerimientos No Funcionales*

**Tabla 37**

*Requerimiento No Funcional 1*

<b>Identificación del requerimiento:</b>	RNF01
<b>Nombre del Requerimiento:</b>	Interfaz del sistema.
<b>Características:</b>	El sistema presentara una interfaz de usuario sencilla para que sea de fácil manejo a los usuarios del sistema.
<b>Descripción del requerimiento:</b>	La interfaz debe contar con el diseño y estética trabaja en el transcurso de su desarrollo
<b>Prioridad del requerimiento:</b>	Alta

*Nota.* Se muestra la descripción de la interfaz con la que interactúa el usuario final.

**Tabla 38***Requerimiento No funcional 2*

<b>Identificación</b>	RNF02
<b>del requerimiento:</b>	
<b>Nombre del Requerimiento:</b>	Mantenimiento.
<b>Características:</b>	El sistema deberá de tener un manual de instalación y manual de usuario para facilitar los mantenimientos que serán realizados por el administrador.
<b>Descripción del requerimiento:</b>	Flexibilidad ante futuras modificaciones en su desarrollo.
<b>Prioridad del requerimiento:</b>	Alta

*Nota.* Se definen las características del mantenimiento del sistema de información.

**Tabla 39***Requerimiento No Funcional 3*

<b>Identificación</b>	RNF03
<b>del requerimiento:</b>	
<b>Nombre del Requerimiento:</b>	El Diseño de la Interfaz Responsive
<b>Características:</b>	El sistema deberá ser adaptado a todo tipo de dispositivo en el que se despliegue ya sea en un computador o celular.
<b>Descripción del requerimiento:</b>	La interfaz del usuario deberá
<b>Prioridad del requerimiento:</b>	Alta

*Nota.* Se muestran las características del diseño Responsive del sistema de información.

**Tabla 40***Requerimiento No Funcional 4*

<b>Identificación del requerimiento:</b>	RNF04
<b>Nombre del Requerimiento:</b>	Desempeño
<b>Características:</b>	El sistema garantizara a los usuarios un desempeño en cuanto a los datos almacenado en el sistema ofreciéndole una confiabilidad a esta misma.
<b>Descripción del requerimiento:</b>	Garantizar el desempeño del sistema informático a los diferentes usuarios. En este sentido la información almacenada o registros realizados podrán ser consultados y actualizados permanente y simultáneamente.
<b>Prioridad del requerimiento:</b>	Alta

*Nota.* Se muestra la descripción del desempeño del sistema de información.

**Tabla 41***Requerimiento No Funcional 5*

<b>Identificación</b>	RNF05
<b>del</b>	
<b>requerimiento:</b>	
<b>Nombre del</b>	Nivel de Usuario
<b>Requerimiento:</b>	
<b>Características:</b>	Garantizará al usuario el acceso de información de acuerdo al nivel que posee.
<b>Descripción del</b>	Seguridad para un usuario y contraseña contra el acceso, alteración
<b>requerimiento:</b>	y destrucción no autorizada de Información.
<b>Prioridad del requerimiento:</b>	
	Alta

*Nota.* Se muestran las características de la seguridad del sistema de información.

**Tabla 42***Requerimiento No Funcional 6*

<b>Identificación del requerimiento:</b>	RNF06
<b>Nombre del Requerimiento:</b>	Seguridad en información
<b>Características:</b>	La aplicación debe contar con seguridad de usuario, para ofrecer la privacidad necesaria en su desarrollo.
<b>Descripción del requerimiento:</b>	Los datos ingresados por el usuario deben contar con la seguridad por parte de las bases de datos que lo almacenaran y de igual forma ser utilizada esta información para futuras consultas y resultados.
<b>Prioridad del requerimiento:</b>	Alta

*Nota.* Se muestran las características de la seguridad de la información con la que cuenta el usuario.

### 8.3.3. *Diagramas de UML del sistema*

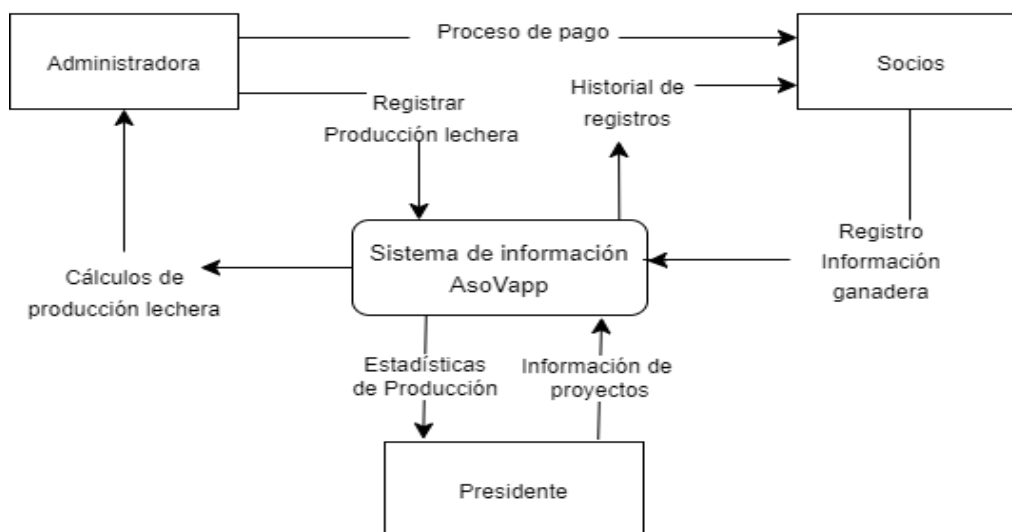
Estos permiten realizar una representación gráfica especificando las características de los objetos, estado y procesos que se realizan en AsoVapp para facilitar la comprensión del sistema por otros. Sirviendo, así como documentación ya que son utilizados para definir las acciones que son posibles de efectuar en el sistema por el usuario final.

A continuación, se presenta cada uno de los diagramas implementados en el desarrollo de este proyecto, cada uno de estos diagramas nos permite ver cómo será la estructura del sistema y el manejo de los datos que este requieren. Este diagrama tiene como finalidad ofrecer una representación visual clara y concisa de las interacciones entre nuestro sistema y su entorno.

Como parte fundamental del análisis y diseño de nuestro sistema de información, ya que nos brinda una visión general de la interacción de los usuarios.

### Figura 5

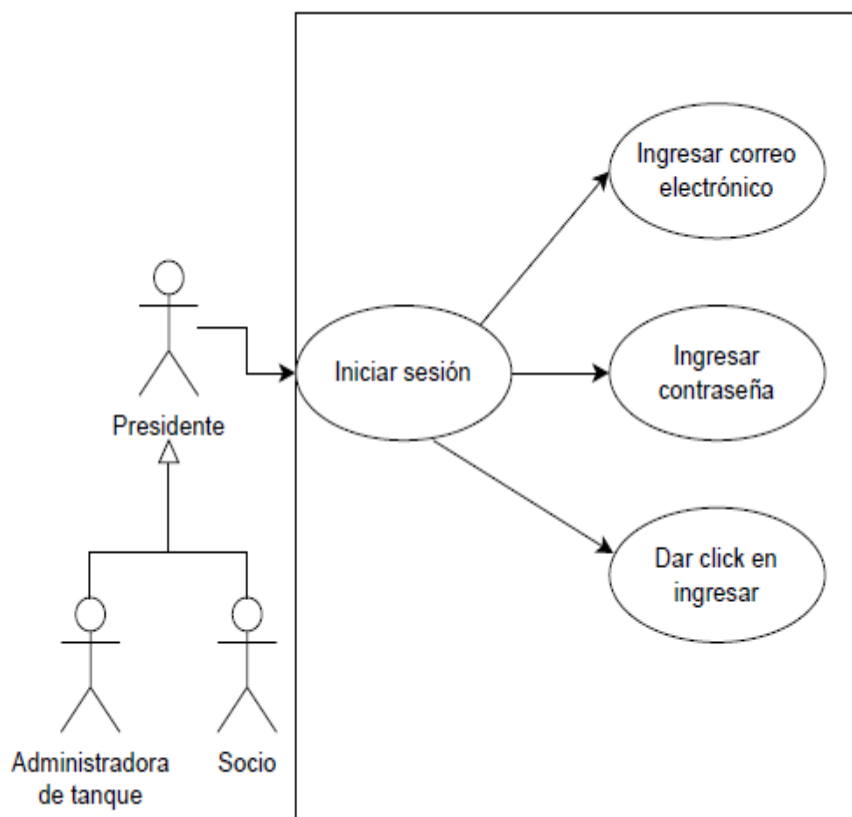
#### Diagrama de contexto



*Nota.* El diagrama de contexto muestra la interacción de los diferentes roles con el sistema, principalmente las funciones más relevantes y la respuesta del sistema a estos.

Estos nos permiten visualizar las interacciones claves entre los actores del sistema como lo es el presidente, la administradora de tanque y los socios es decir cada uno de los ganaderos de la asociación Asoagropuleche.

También nos ayudan a identificar las diversas funcionalidades y acciones que realizarán los usuarios en el sistema de información para la entidad lechera.

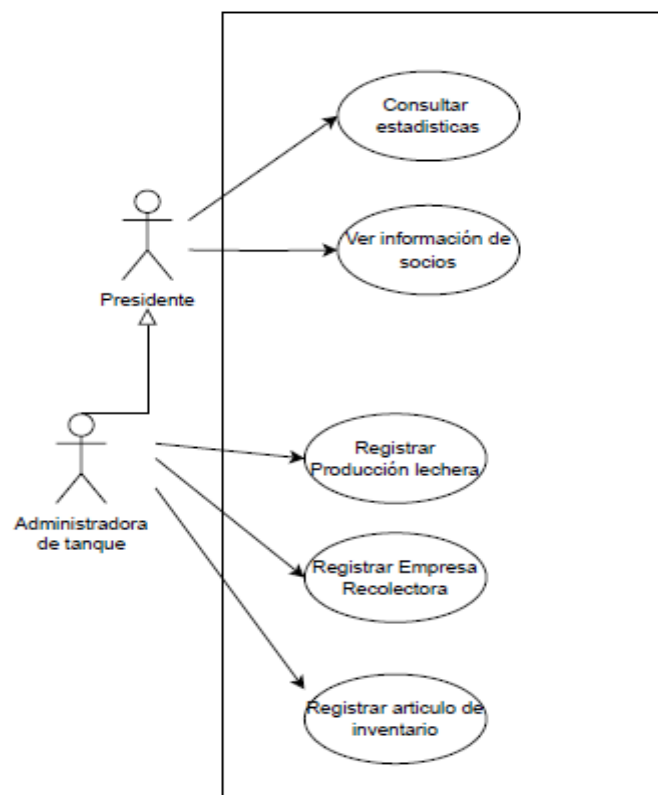
**Figura 6***Caso de uso Inicio sesión*

*Nota.* Se muestra como los usuarios del sistema AsoVapp podrán iniciar sesión.

En el caso de uso anterior muestra la visualización de las acciones que un usuario autenticado, con su identidad en el sistema puede acceder a funcionalidades específicas según el rol. Una vez que el usuario ingresa sus credenciales, el sistema verifica si son válidas y corresponden a una cuenta de usuario existente.

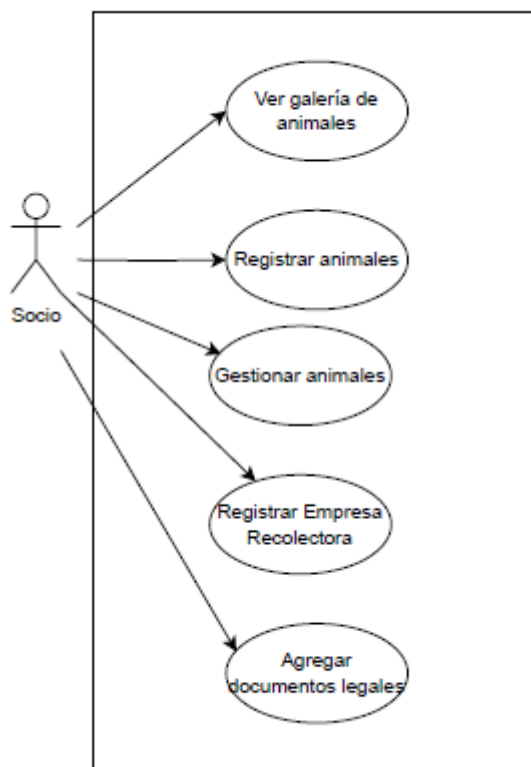
**Figura 7**

*Diagrama de casos de Usos de presidente*



*Nota.* La imagen muestra la interacción con el sistema por parte del presidente de la asociación de ganaderos.

En este se muestra la visualización de los roles del presidente y la administradora de tanque quienes pueden realizar ciertas acciones dentro del sistema permitiéndoles la toma de decisiones en cuanto a la producción lechera que se va suministrando al sistema el cual le permite ver un análisis por medio de estadísticas generando la trayectoria e impacto que se tiene en el transcurrir de los días en la asociación.

**Figura 8***Diagrama caso de uso socio*

*Nota.* La imagen muestra la interacción con el sistema por parte del ganadero o socio de la asociación de ganaderos.

**8.4. Metodología de desarrollo**

La metodología ágil XP se utilizará para el desarrollo del sistema de información AsoVapp, ya que se basa principalmente en la comunicación constante entre los desarrolladores y los clientes teniendo en cuenta cinco valores fundamentales, los cuales son la comunicación, valentía, simplicidad, retroalimentación y respeto. El objetivo de estos valores es que el equipo

de desarrolladores trabaje bajo una mentalidad conjunta para poder colaborar y crear un producto de alto nivel. Esta metodología se compone de cuatro etapas principales las cuales son la planificación, el diseño, la codificación y las pruebas, se caracteriza por generar entregas constantes las cuales permiten tener un ritmo de trabajo sostenido y flexible en el que se puede aportar mejoras continuas al producto. La utilización de esta metodología va encaminada con el uso de Scrum, pues ambas proponen como método de retroalimentación la comunicación constante entre los miembros del equipo, y la distribución de responsabilidades en partes iguales. Las entregas se realizan de manera iterativa en determinados intervalos de tiempo, lo que permite tener un mayor control acerca del proceso de desarrollo que se lleva a cabo.

#### **8.4.1. *Diseño de Logo***

El sector agrícola colombiano tiene un enorme potencial de crecimiento en las nuevas tecnologías, dado que hasta ahora afronta un grave rezago en materia de competitividad. Y es que el 44,7% de la población rural está en pobreza, mientras que 20% de los jóvenes de ese segmento de la población no ha recibido educación formal. Esta problemática se manifiesta en la actividad campesina y se agudiza debido a la falta de apoyo, ya que solo el 9% de los productores cuenta con asistencia técnica. (Semana., 2016)

**Figura 9**

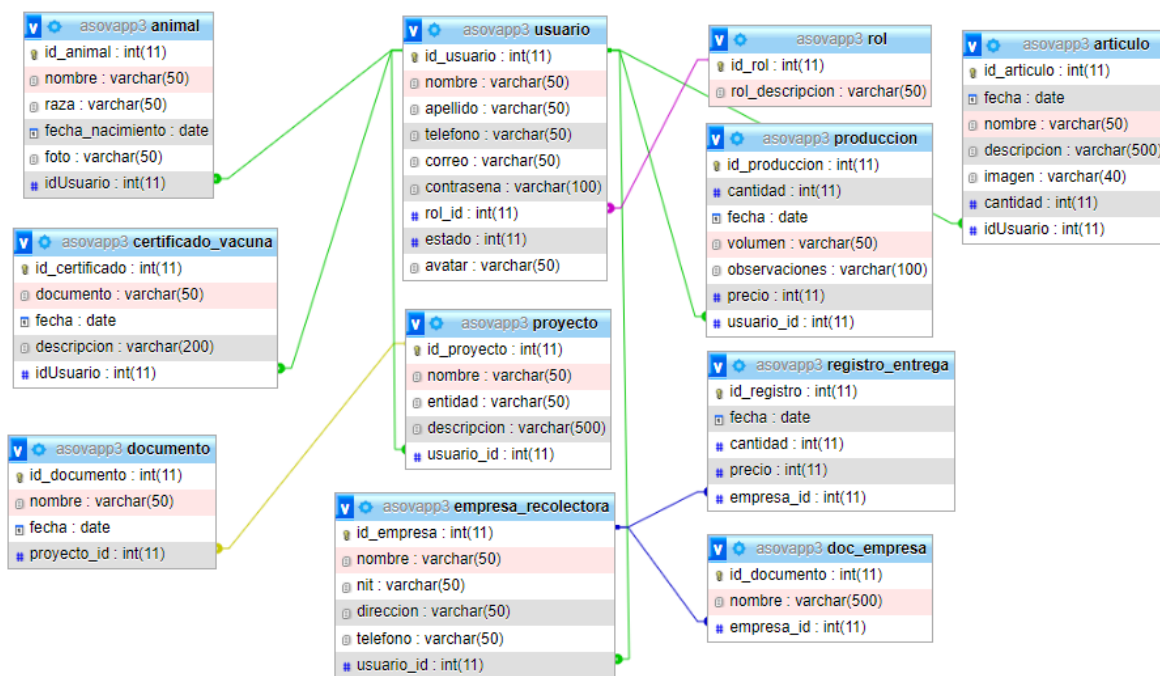
*Logo de AsoVapp*



*Nota.* Diseño realizado por el equipo de trabajo en el cual se evidencia el diseño del logo del sistema.

El logo de AsoVapp lo componen la mano que representa el cursor del mouse lo que transmiten paz y armonía en sus actividades diarias donde son acompañados de las garzas aves que acostumbran a estar cerca de grandes grupos de ganados también el negro color usado representa la elegancia y el poder con el que cuentan los campesinos que son los que trabajan por el sustento de las grandes ciudades.

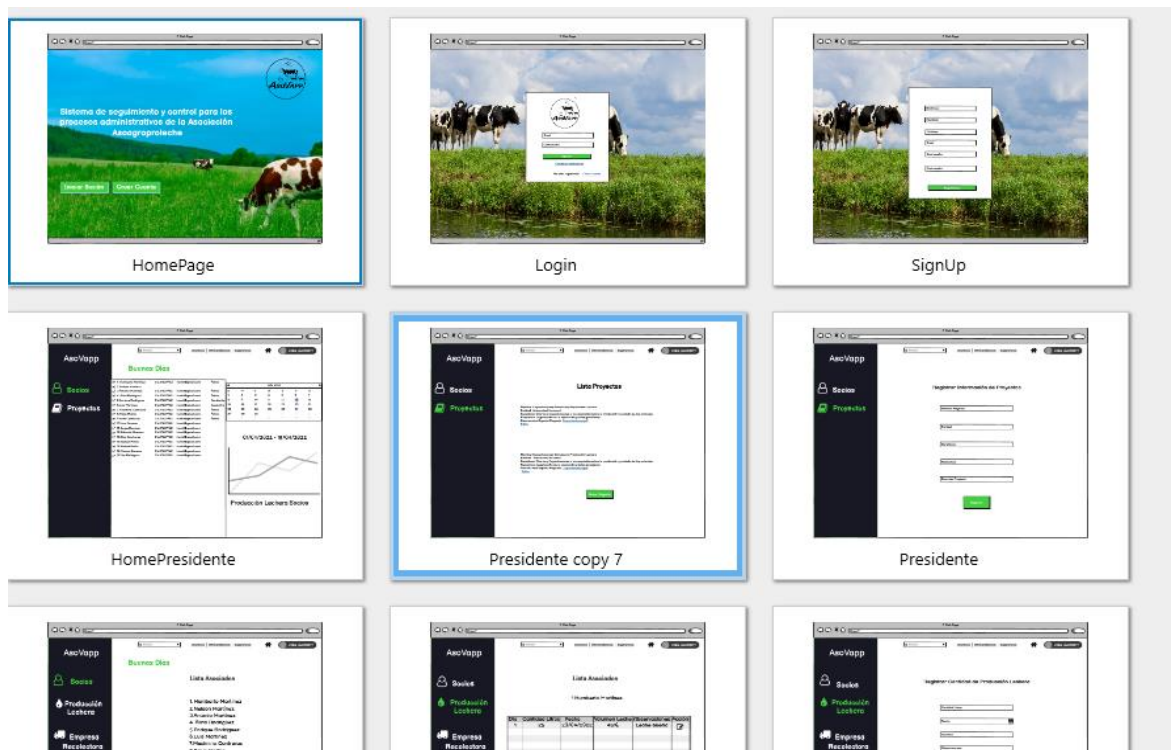
La economía campesina es una forma de producción familiar que utiliza efectivamente el trabajo y los recursos naturales, para asegurar la supervivencia de la unidad familiar y mejorar su calidad de vida.

**Figura 10***Diseño de la base de datos*

*Nota.* Se muestra el modelo relacional de la base de datos del sistema AsoVapp en MySQL.

#### 8.4.2. Diseño de mockups

Cada uno de estos Mockups se realizaron por medio de la aplicación Balsamiq una herramienta de fácil manejo y acceso, ya que ofrece bastantes elementos como menús, barras de navegación, botones entre otros, los cuales permitieron la construcción del diseño del sistema “AsoVapp”. Los cuales contienen cada uno de las opciones, estadísticas, gráficos, formularios e historiales de cada uno de los roles como presidente, administradora de tanque y el socio en el sistema.

**Figura 11***Mockups de AsoVapp*

*Nota.* Se muestran algunos de los módulos con los que contará el sistema.

En estos mockups se realiza el diseño de los campos y requisitos para poder el usuario final registrarse en el sistema con su nombre, celular, usuario, rol y la respectiva contraseña donde será guardado en la base de datos la cual con el login valida cada uno de los datos guardados con su usuario y contraseña podrá iniciar sesión y disponer de cada uno de los servicios que este ofrece. Se realizó el diseño del sistema AsoVapp teniendo en cuenta las respuestas que compartieron cada uno de los integrantes de la Asociación Asoagroproleche” como lo que les gustaría que el sistema realizara y les ayudara para un mejor progreso y manejo de cada una de las actividades y procesos. En el siguiente link esta explicado más detalladamente los mockups que se realizaron. [Ver anexos.](#)

## Pruebas

En la realización de pruebas se evaluaron las características, beneficios de cada uno de los módulos. Además, se presentan las herramientas con las cuales se hicieron algunas pruebas y enfoques útiles para llevarlas a cabo de manera efectiva.

### 9.1. Pruebas unitarias

Estas pruebas se realizan a nivel de cada componente individual del sistema para asegurarse de que cada uno de ellos funciona correctamente.

#### 9.1.1. Módulo inicio de sesión

Se hace la validación de usuario en la base de datos, con el correo erróneo *pepas@gmail.com* un correo que no ha sido registrado lo cual le muestra el mensaje de que el usuario o contraseña son incorrectos.

### Figura 12

#### *Prueba unitaria de Inicio de Sesión*

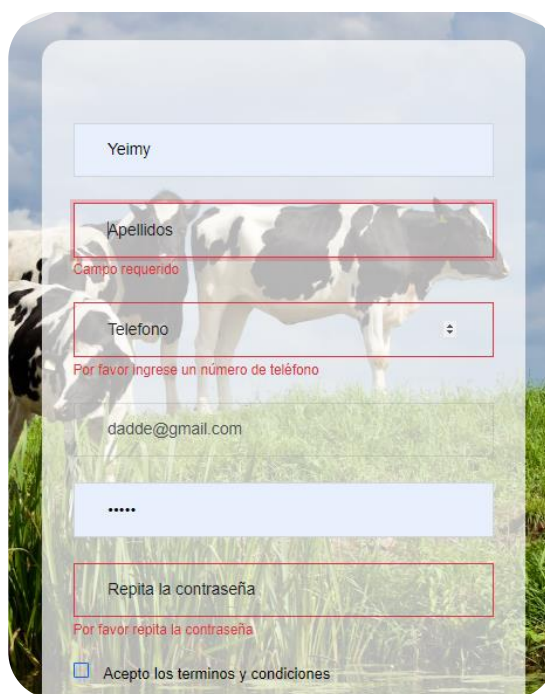


*Nota.* Se muestra la validación de un usuario erróneo, lo cual no le permite el ingreso al sistema.

Al registro también se le realiza la validación de que todos los campos estén llenos en caso de que estén vacíos no le permitirá guardar los cambios y ser registrado.

### Figura 13

#### *Prueba Unitaria del registro*



The image shows a registration form with several fields. The first field contains the name 'Yeimy'. The 'Apellidos' field is empty and has a red border with the error message 'Campo requerido' below it. The 'Telefono' field is empty and has a red border with the error message 'Por favor ingrese un número de teléfono' below it. The email field contains 'dadde@gmail.com'. The password field contains four dots and has a red border with the error message 'Por favor repita la contraseña' below it. At the bottom, there is a checkbox labeled 'Acepto los terminos y condiciones' which is currently unchecked.

*Nota.* Se muestra que al no completar los campos le muestra una alerta de que debe ingresar todos los datos para poder ser registrado en el sistema.

## 9.2. Pruebas de integración

Estas nos permiten garantizar el correcto funcionamiento y la interacción adecuada entre los diferentes autores del sistema como el presidente, administradora de tanque y los socios.

Estas pruebas permiten identificar y solucionar los posibles errores o fallas que se puedan presentar en la integración de cada uno los módulos.

### 9.2.1. Módulo de presidente

**Tabla 43**

*Prueba unitaria*

<b>Nombre del caso de prueba</b>	Administrar los proyectos
<b>Descripción</b>	Ingresar la información sobre los proyectos en los que participa la asociación junto con cada uno de los documentos de soporte. Realizar alguna modificación de los datos que ha ingresado en el momento que el presidente lo desee. Consultar todo el historial de los proyectos que han sido registrados en el sistema
<b>Pasos a seguir</b>	<ul style="list-style-type: none"> <li>• Iniciar sesión como presidente</li> <li>• En el menú de opciones seleccionar proyectos</li> <li>• Agregar cada uno de los datos solicitados por el sistema.</li> <li>• Verificar que se guarde correctamente y se muestre en la pantalla.</li> </ul>
<b>Resultado esperado</b>	El sistema almacenara cada uno de estos datos y los mostrara en una sección organizada junto con los documentos en formato pdf ingresados.
<b>Resultado obtenido</b>	Esta prueba se realiza correctamente debido que al seleccionar la opción de proyectos se mostró el historial de cada proyecto registrado, se logra realizar modificación de cualquier datos, los documentos ingresado se pueden eliminar.

*Nota.* Se muestra la primera prueba del módulo del presidente.

**Tabla 44***Segunda prueba unitaria*

<b>Nombre del caso de prueba</b>	Validar los documentos asociados
<b>Descripción</b>	Consultar que cada uno de los socios hubieran registrado los dos documentos requeridos el de certificado de vacunas que se debe actualizar cada seis meses y el registro de finca.
<b>Pasos a seguir</b>	<ul style="list-style-type: none"> <li>• Iniciar sesión</li> <li>• Seleccionar la opción de socios</li> <li>• Seleccionar el socio que desea revisar</li> <li>• Confirmar que los documentos estén actualizados.</li> </ul>
<b>Resultado esperado</b>	El sistema debe mostrar el historial de los documentos que ha registrado el socio de su finca ganadera.
<b>Resultado obtenido</b>	Esta prueba se realiza correctamente ya que se consultaron todos los documentos que ha registrado el socio desde que se vinculó con la asociación.

*Nota.* Se muestra la segunda prueba unitaria del módulo del presidente.

**Tabla 45***Tercera prueba unitaria*

<b>Nombre del caso de prueba</b>	Consultar estadísticas
<b>Descripción</b>	Permitirá elegir el rango de las fechas de la quincena a la cual desee consultar la cantidad de litros de la producción lechera de todos los socios y el total del pago de esa producción o si lo desea también podrá realizar este proceso con cada uno de los socios.
<b>Pasos a seguir</b>	<ul style="list-style-type: none"> <li>• Iniciar sesión</li> <li>• En la opción de inicio seleccionara las fechas de la quincena que desea consultar.</li> <li>• Seleccionar el botón de mostrar grafico</li> </ul>
<b>Resultado esperado</b>	El usuario deberá consultar estadísticas de la producción lechera de la asociación.
<b>Resultado obtenido</b>	Se realiza correctamente debido a que el cliente del sistema logro consultar las estadísticas de la producción lechera de la entidad.

*Nota.* Se muestra la tercera prueba unitaria del módulo del presidente.

### 9.2.2. *Módulo de administradora*

**Tabla 46***Cuarta prueba unitaria*

<b>Nombre del caso de prueba</b>	Registrar producción lechera
<b>Descripción</b>	En esta prueba la administradora de tanque debe realizar el registro de la cantidad de producción lechera que recibe de los socios y así mismo la que es vendida a la empresa recolectora.
<b>Pasos a seguir</b>	<p>Iniciar sesión</p> <p>Seleccionar la opción de gestionar socios</p> <p>Seleccionar el nombre del socio se desplegará un menú de opciones deberá elegir la de registrar la producción con cada una de las propiedades</p> <p>Confirmar que se guarde los datos ingresados por medio del botón registrar.</p> <p>Podrá exportar esta información en formatos pdf o Excel para compartir con quien requiera.</p>
<b>Resultado esperado</b>	El sistema debe calcular el total de los litros registrados de cada uno de los socios y el total de todos.
<b>Resultado obtenido</b>	Se realiza correctamente ya que el sistema entrega por medio de tablas, gráficos el total de litros de la producción de cada uno de los socios y de todos

*Nota.* Se muestra la primera prueba unitaria del módulo de la administradora.

**Tabla 47***Quinta Prueba Unitaria*

<b>Nombre del caso de prueba</b>	Administrar a los asociados
<b>Descripción</b>	En este caso podrá gestionar cada uno de los datos de los socios ya se modificar su estado activo o inactivo y realizar alguna modificación de los registros de producción
<b>Pasos a seguir</b>	Iniciar sesión  Seleccionar la opción de gestionar socios  Elegir el nombre del socio luego elegir las fechas de la quincena la cual desea verificar los datos de la producción o cambiar estado activo o inactivo.
<b>Resultado esperado</b>	El sistema en caso de que se inactive el socio no debe permitirle realizar ninguna acción dentro del sistema.
<b>Resultado obtenido</b>	Se inactivo el estado de tres socios los cuales ya no llevan la producción lechera a la asociación.

*Nota.* Se muestra la segunda prueba unitaria del módulo de la administradora.

### 9.2.3. Módulo de socios

**Tabla 48**

*Sexta prueba unitaria*

<b>Nombre del caso de prueba</b>	Registrar Animales
<b>Descripción</b>	En esta le permite registrar cada uno de los animales con las respectivas características y fotografías .
<b>Pasos a seguir</b>	<p>Iniciar sesión</p> <p>Seleccionar la opción de animales</p> <p>Ingresar los datos que solicita el formulario</p> <p>Validar que se registre por medio del botón guardar</p>
<b>Resultado esperado</b>	El sistema deberá almacenar estos datos y mostrarlos en una galería organizada por orden de fecha de ingreso del usuario.
<b>Resultado obtenido</b>	El sistema muestra la galería de animales con cada uno de los datos suministrados por el usuario.

*Nota.* Se muestra la primera prueba unitaria del módulo del socio.

**Tabla 49***Séptima prueba unitaria*

<b>Nombre del caso de prueba</b>	Consultar Producción lechera
<b>Descripción</b>	En esta opción se le permitirá al usuario consultar por fechas la cantidad de litros de leche que han sido registrados por la administradora de tanque.
<b>Pasos a seguir</b>	<p>Iniciar sesión</p> <p>Seleccionar la opción de producción lechera</p> <p>Elegir el rango de la quincena que desea consultar</p> <p>Seleccionar el botón de mostrar registros.</p> <p>Podrá exportar esta información en formatos pdf o Excel.</p>
<b>Resultado esperado</b>	El sistema debe mostrar la cantidad de la producción lechera con cada una de las propiedades suministradas por la administradora de tanque.
<b>Resultado obtenido</b>	El usuario logro consultar la producción lechera que ha vendido a la asociación.

*Nota.* Se muestra la segunda prueba unitaria del módulo del socio.

**Tabla 50***Octava prueba unitaria*

<b>Nombre del caso de prueba</b>	Registrar documentos
<b>Descripción</b>	El sistema permite registrar cada uno de los documentos como el de certificación de vacunas y el de registro de fincas, la respectiva descripción y fecha en la que se subieron al sistema.
<b>Pasos a seguir</b>	<p>Iniciar sesión</p> <p>Elegir la opción de subir documentos</p> <p>Seleccionar el botón de agregar documentos</p> <p>Ingresar cada uno de los la información por medio del botón guardar</p>
<b>Resultado esperado</b>	El sistema deberá almacenar el historial de todos los documentos que se ingresen.
<b>Resultado obtenido</b>	Esta prueba se realiza correctamente ya que el socio puede verificar que se realiza el registro exitosamente.

*Nota.* Se muestra la tercera prueba unitaria del módulo del socio.

### 9.3. Pruebas de rendimiento

En estas permite conocer si el sistema se está cargando de manera eficaz, permitiendo al usuario navegar de manera que adquiriera una buena experiencia. Esta se realizó con la herramienta GTMetrix que permite ingresar la URL donde se utiliza para medir el rendimiento, ya que esta herramienta realiza una serie de pruebas en la página y genera un informe que

proporciona información detallada sobre el rendimiento de la página, incluyendo la velocidad de carga, el tamaño de la página y las mejoras recomendadas. A continuación, se muestra el informe generado:

### Figura 14

*Prueba de velocidad al sistema AsoVapp*



*Nota.* Se muestra el rendimiento del sistema a nivel general.

Luego se muestra el tiempo de carga del sistema en el navegador.

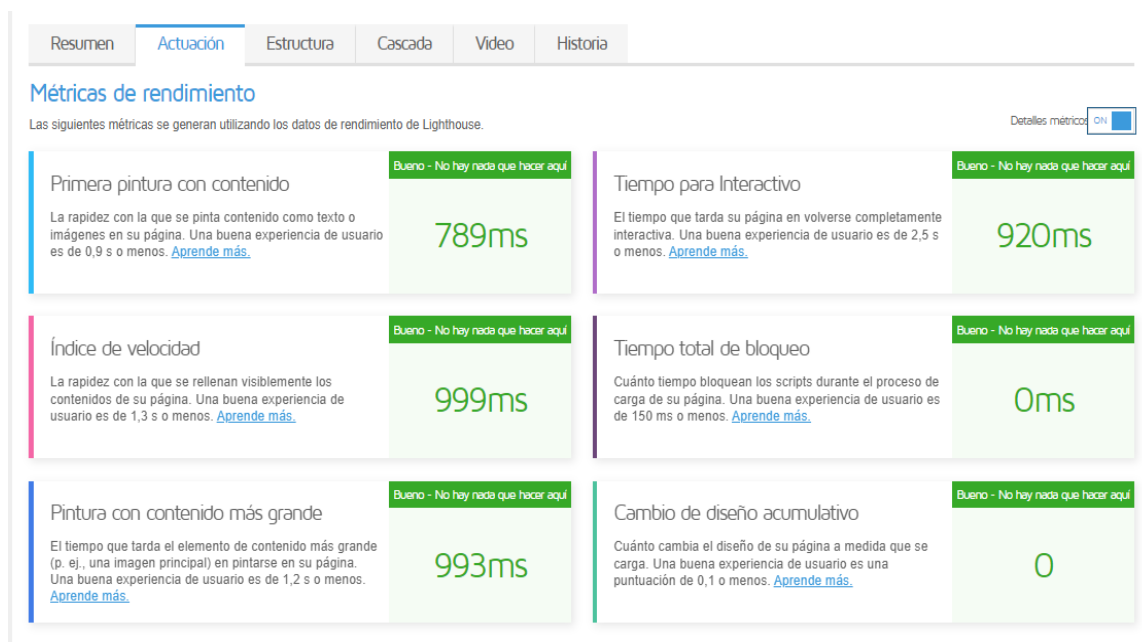
## Figura 15

### Tiempo de carga

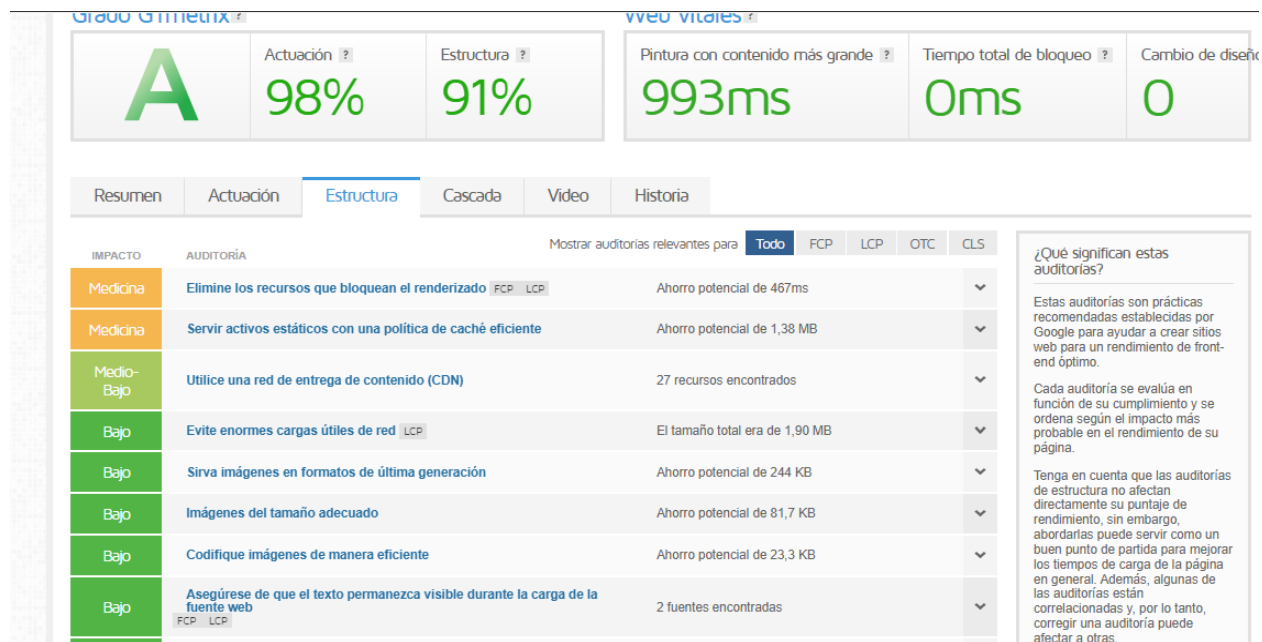


*Nota.* Se muestran los tiempos en que carga por completo el aplicativo.

Otro de los aspectos que entrega el gtmatrix es el de como actúa el sistema en cuanto a la familiarización y experiencia del usuario con el frontend.

**Figura 16***Actuación del sistema*

*Nota.* Se observa que el informe generado por Gtmetrix muestra que el cargue de texto e imágenes se demoran en un tiempo óptimo.

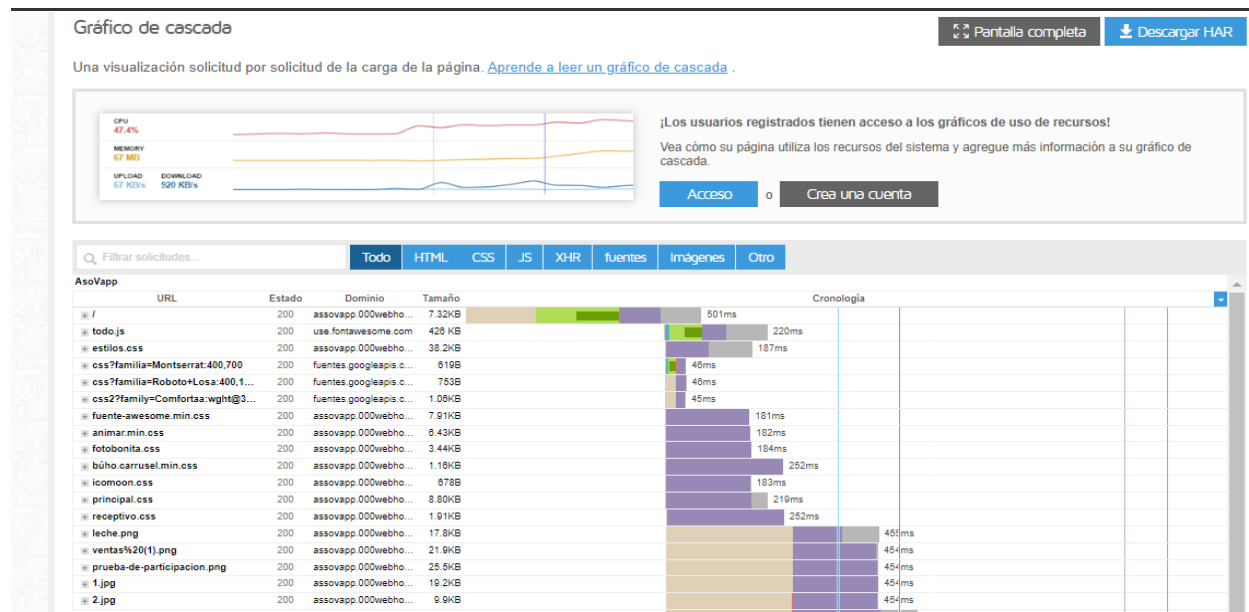
**Figura 17***Estructura del sistema*

*Nota.* Se muestra la puntuación que da el informe en cuanto al impacto que se tiene del sistema dentro del navegador.

En la estructura del sistema se evidencia cómo se compone la página web y cómo se cargan sus elementos, donde proporciona información sobre cómo se organizan y entregan los recursos de la página, como HTML, CSS, JavaScript e imágenes especificando el problema que hay con el fin de mejorarlo o arreglarlo para generar una mejor experiencia con el usuario final.

Figura 18

## Gráfico de cascada del sistema

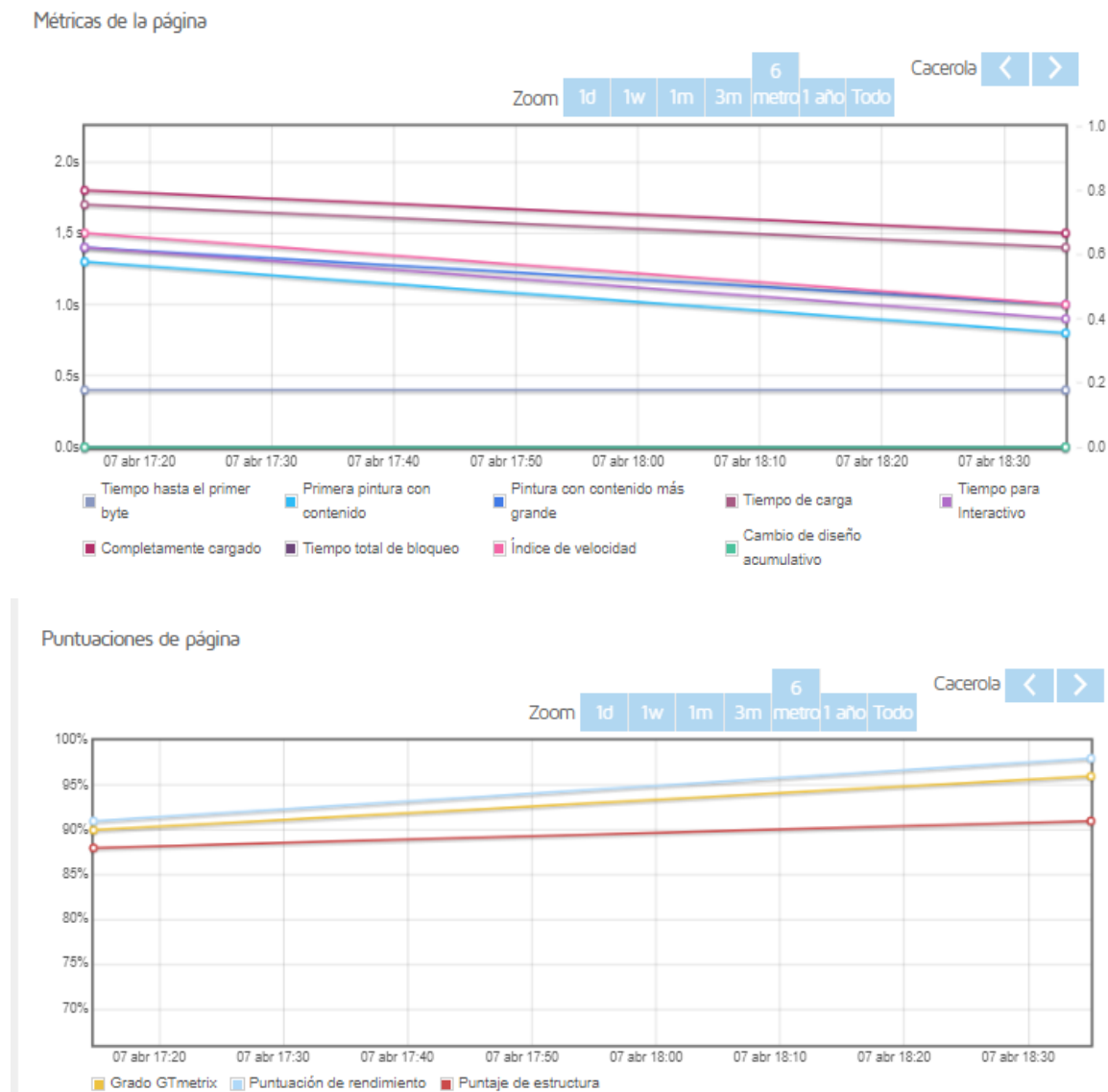


*Nota.* Se muestra cada uno de los recursos y el cargue de cada uno y lo que consume el sistema web dentro del navegador

En el gráfico de cascada que entrega gtmatrix representa todos los recursos que se cargan en una página web en orden cronológico y el tiempo que tarda cada uno en cargarse. Esta permite identificar qué recursos tardan más tiempo en cargarse y cómo se pueden mejorar los tiempos de carga.

## Figura 19

### Historia del sistema



*Nota.* Esta permite comparar diferentes resultados de prueba y ver cómo se han mejorado o empeorado diferentes métricas de rendimiento, como el tiempo de carga de la página y el tamaño de la página.

#### 9.4. Pruebas de aceptación

Estas se realizan para verificar que el sistema de información funciona de acuerdo con las necesidades y expectativas del cliente y que cumple con cada uno de los requisitos que se plantearon inicialmente. Esto incluye probar la funcionalidad, la usabilidad, el rendimiento, la seguridad y la compatibilidad del sistema con el usuario final en este caso con los integrantes de la asociación ganadera Asoagroproleche.

Se realizaron algunas preguntas el día que se realizó la visita a las instalaciones de la asociación donde se le aplicó a tres socios y a la administradora de tanque una muestra de la población objetivo del sistema de información AsoVapp.

- ¿El sistema de información es de fácil uso o encuentra alguna dificultad para utilizarlo?
- ¿Encuentra que el sistema AsoVapp es útil para llevar un registro y seguimiento de la producción lechera?
- ¿Ha encontrado algún problema o error mientras hace uso del sistema?
- ¿El sistema cumple con las funcionalidades que espera? ¿Por qué?

A continuación, se presentan los resultados de las respuestas obtenidas del usuario final: [ver anexos](#)

**Figura 20***Evidencia*

*Nota.* Aplicación de la entrevista a el usuario final.

En la figura 19 se muestra la visita que se realizó a las instalaciones de la asociación donde se aplicaron las entrevistas de pruebas finales del sistema de información el cual nos permitió comprobar cada una de las funcionalidades del sistema con el usuario final, se evidencio que se cumplieron con los requerimientos inicialmente planteados y se obtuvieron resultados satisfactorios por parte de los integrantes de la asociación y el equipo desarrollador.



## Resultados

Para lograr el desarrollo del sistema de información AsoVapp inicialmente se plantean las técnicas de recolección de datos como entrevistas que fueron planteadas y aplicadas a cada uno de los miembros de la asociación como el presidente, administradora de tanque y cada uno de los socios, es decir, los ganaderos los principales proveedores de producción lechera donde por medio de la visita realizada a las instalaciones de la asociación se consiguió aplicar cada una de las entrevistas realizadas y así evidenciar cada uno de los procesos que se realizan en la recolección de la producción lechera como lo es el manejo de técnicas de cálculo, elementos que usan, el registro de la materia prima e historial de producción lechera en cada uno de los años.

### Figura 21

#### *Proceso producción lechera*



*Nota.* Se muestra como es el recaudo y venta de la producción lechera en la asociación Asoagroproleche.

Se identificaron los procesos que manejan en la asociación Asoagroproleche donde luego se plantearon cada uno de los requerimientos, se construyó el diseño y el respectivo desarrollo del sistema el cual unifica y centraliza toda la información de esta empresa para así poder generar informes, reportes estadísticos de la trayectoria de cada uno de los socios y como ha sido el comportamiento de cada una de las variables, es decir, las propiedades de la producción lechera. Permitiendo la toma de decisiones para el beneficio y progreso de la entidad de enfriamiento como también así mismo lograr concebir unas predicciones que se puedan llegar a presenciar más adelante.

Por último, con la realización de las pruebas de integración de módulos y de aceptación permitieron evidenciar que el sistema cumple con los requisitos establecidos inicialmente con los usuarios finales en este caso los integrantes de la asociación Asoagroproleche donde quedaron satisfechos con la ejecución del sistema, el manejo de los procesos y actividades que allí realizan. También implicaron la validación de los casos de uso del sistema donde ayudo a asegurar que el sistema es capaz de realizar las funciones requeridas por los usuarios.

El link de acceso al sistema de información: <https://asovapp.tech/>

## Conclusiones

Se identificaron las características de los procesos administrativos de la asociación Asoagroproleche obteniendo un análisis que proporciono una base sólida para diseñar el sistema las cuales fueron fundamentales para comprender cada uno de los procesos y la situación que estos realizaban y así se definieron las necesidades específicas de esta entidad.

La plataforma web del sistema de información ha demostrado su eficacia y funcionalidad en el entorno de trabajo de la asociación. A lo largo del desarrollo y las pruebas, hemos podido verificar que el sistema cumple con los objetivos planteados, proporcionando una solución integral y eficiente para las necesidades de gestión de información de la asociación.

Durante las pruebas, se evaluaron diferentes aspectos clave de la plataforma, como su usabilidad, rendimiento, seguridad y capacidad de respuesta. Los resultados obtenidos han sido muy positivos, evidenciando una interfaz intuitiva que facilita la interacción de los usuarios con el sistema. Además, la plataforma ha demostrado ser estable y rápida, permitiendo un acceso fluido y ágil a la información requerida.

El uso del sistema de información ha logrado una significativa optimización de los tiempos requeridos en las labores administrativas de la asociación como los registros de producción, cálculo de pagos a socios y manejo de archivos. Mediante la automatización de tareas y la centralización de la información en una sola plataforma, se ha logrado reducir considerablemente los tiempos dedicados a actividades administrativas repetitivas y de baja complejidad. El sistema ha permitido una mayor eficiencia en la gestión de datos, simplificando el ingreso, actualización y consulta de información por parte del personal de la asociación.

El sistema permitirá a los diferentes roles de la asociación llevar un registro detallado de las actividades que realizan y de la información recolectada, lo que facilitará la toma de decisiones y la gestión eficiente de cada uno de los procesos que se realizan allí.

El desarrollo del proyecto se llevó a cabo mediante metodologías de trabajo y de investigación que permitieron garantizar la calidad del sistema y la satisfacción de las necesidades de la asociación.

## Referencias

- Ángel, M. G. (s.f.). Obtenido de METODOLOGÍA INVESTIGACIÓN MIXTA TESIS  
ARQUITECTURA 1. :  
<http://repository.unipiloto.edu.co/bitstream/handle/20.500.12277/10852/Trabajo%20de%20Grado.pdf?sequence=1>
- Bases de datos. Up.ac.pa. (2016). En C. M. Ricardo. México, D. F.: McGRAW-HILL  
INTERAMERICANA EDITORES, S.A. de C.V. Obtenido de <https://doi.org/978-970-10-7275-2>
- Bertalanffy, L. V. (1968). *Teoría General de Sistemas: fundamentos, desarrollo, aplicaciones*.  
*Constitución Política de Colombia*. (1991). Obtenido de  
<https://www.mincit.gov.co/ministerio/normograma-sig/procesos-estrategicos/gestion-de-informacion-y-comunicacion/constitucion-politica/derechos/articulo-15.aspx>
- El Tiempo*. (12 de Julio de 2019). Obtenido de  
<https://www.eltiempo.com/justicia/investigacion/cronologia-del-caso-por-el-que-fue-condenado-andres-felipe-arias-38758>
- Holland Ben Light, C. (1999).
- Kenneth C. Laudon, J. P. (2016). *Sistemas de Información Gerencial*. México: PEARSON  
EDUCACIÓN.
- Medina, Á. M. (17 de Marzo de 2022). *Universidad ORT Uruguay Facultad de Ingeniería*.  
Obtenido de  
<https://dspace.ort.edu.uy/bitstream/handle/20.500.11968/3409/Material%20completo.pdf?sequence=-1&isAllowed=y>

Medina., M. A. (7 de Enero de 2021). *ELESPECTADOR.COM*. Obtenido de

<https://www.elespectador.com/economia/las-tic-en-2020-un-servicio-esencial-en-tiempos-de-pandemia-article/>

Ministerio de Ambiente y Desarrollo Sostenible. (13 de Octubre de 2022). Obtenido de

<https://www.minambiente.gov.co/wp-content/uploads/2023/03/DS-E-GET-01.pdf>

Modificación, Ú. (2008). *Lenguajes de programación*. Obtenido de

[https://guimi.net/descargas/Monograficos/G-Lenguajes\\_de\\_programacion.pdf](https://guimi.net/descargas/Monograficos/G-Lenguajes_de_programacion.pdf)

Modrego, J. M. (2006). *SISTEMAS GESTORES DE BASES*. Madrid: McGraw-

Hill/Interamericana de España, S.A.U. Obtenido de

[https://d1wqtxts1xzle7.cloudfront.net/44649113/Sistemas\\_Gestores\\_de\\_Bases\\_de\\_Datos-with-cover-page-](https://d1wqtxts1xzle7.cloudfront.net/44649113/Sistemas_Gestores_de_Bases_de_Datos-with-cover-page-v2.pdf?Expires=1650651275&Signature=dt3J~xDmzAEzIXn58xWQMScP0jRiHdInfvRuDv168ppUGr3R-j2N25QLo2yP3RvIVGMmwXmYFSrJl6h-Ax2noGzyJWjiz8wR~WXzIsohWyibl8H6D~kplRKb)

[s-with-cover-page-](https://d1wqtxts1xzle7.cloudfront.net/44649113/Sistemas_Gestores_de_Bases_de_Datos-with-cover-page-v2.pdf?Expires=1650651275&Signature=dt3J~xDmzAEzIXn58xWQMScP0jRiHdInfvRuDv168ppUGr3R-j2N25QLo2yP3RvIVGMmwXmYFSrJl6h-Ax2noGzyJWjiz8wR~WXzIsohWyibl8H6D~kplRKb)

[v2.pdf?Expires=1650651275&Signature=dt3J~xDmzAEzIXn58xWQMScP0jRiHdInfv](https://d1wqtxts1xzle7.cloudfront.net/44649113/Sistemas_Gestores_de_Bases_de_Datos-with-cover-page-v2.pdf?Expires=1650651275&Signature=dt3J~xDmzAEzIXn58xWQMScP0jRiHdInfvRuDv168ppUGr3R-j2N25QLo2yP3RvIVGMmwXmYFSrJl6h-Ax2noGzyJWjiz8wR~WXzIsohWyibl8H6D~kplRKb)

[RuDv168ppUGr3R-j2N25QLo2yP3RvIVGMmwXmYFSrJl6h-](https://d1wqtxts1xzle7.cloudfront.net/44649113/Sistemas_Gestores_de_Bases_de_Datos-with-cover-page-v2.pdf?Expires=1650651275&Signature=dt3J~xDmzAEzIXn58xWQMScP0jRiHdInfvRuDv168ppUGr3R-j2N25QLo2yP3RvIVGMmwXmYFSrJl6h-Ax2noGzyJWjiz8wR~WXzIsohWyibl8H6D~kplRKb)

[Ax2noGzyJWjiz8wR~WXzIsohWyibl8H6D~kplRKb](https://d1wqtxts1xzle7.cloudfront.net/44649113/Sistemas_Gestores_de_Bases_de_Datos-with-cover-page-v2.pdf?Expires=1650651275&Signature=dt3J~xDmzAEzIXn58xWQMScP0jRiHdInfvRuDv168ppUGr3R-j2N25QLo2yP3RvIVGMmwXmYFSrJl6h-Ax2noGzyJWjiz8wR~WXzIsohWyibl8H6D~kplRKb)

Munevar Rativa, S. &. (2021). *Ucundinamarca.edu.co*. Obtenido de

<https://doi.org/http://hdl.handle.net/20>.

Pressman, R. S. (2010).

René, A. (4 de Octubre de 2020). *Revista Empresarial & Laboral; Revista Empresarial &*

*Laboral*. . Obtenido de [https://revistaempresarial.com/tecnologia/la-importancia-de-las-](https://revistaempresarial.com/tecnologia/la-importancia-de-las-aplicaciones-web-y-moviles-)

[aplicaciones-web-y-moviles-](https://revistaempresarial.com/tecnologia/la-importancia-de-las-aplicaciones-web-y-moviles-)

Semana., R. (19 de Enero de 2016). Obtenido de

[https://www.semana.com/actualidad/articulo/tecnologia-para-mejorar-la-productividad-](https://www.semana.com/actualidad/articulo/tecnologia-para-mejorar-la-productividad-del-campo-colo)

[del-campo-colo](https://www.semana.com/actualidad/articulo/tecnologia-para-mejorar-la-productividad-del-campo-colo)

Standars, I. O. (2019). *Portal ISO 25000*. Obtenido de Portal ISO 25000:

<https://iso25000.com/index.php/normas-iso-25000>

Vargas Parra, A. F. (5 de diciembre de 2016). *repositorio.ucundinamarca*. Obtenido de

<https://repositorio.ucundinamarca.edu.co/bitstream/handle/20.500.12558/299/Sistema%20de%20Informaci%20Web%20para%20la%20administraci%20de%20los%20datos%20del%20ganado%20bovino%20en%20la%20Granja%20el%20.pdf?sequence=1&isAllowed=y>

## **Manual de Usuario**

Yeimy Constanza Ángel Sánchez y Jairo Orlando Pascagaza Barriga

Programa de ingeniería de sistemas, Universidad de Cundinamarca

Trabajo de grado

Director

Cristian Eduardo Cano López

Junio 2 ,2023

## Manual de Usuario

- **Objetivo**

Mostrar detalladamente cada uno de los pasos que deberá seguir para poder acceder y disfrutar de cada uno de los servicios que ofrece el sistema de información para la asociación.

- **Requerimientos para el uso del sistema**

Para poder hacer uso del Sistema de información deberá contar con dispositivo ya sea computador o teléfono con acceso a internet con mínimas características:



- **Página de inicio del sistema**

Se compartirá el link donde podrá acceder por el navegador de su preferencia. Primero podrá navegar para leer información general del sistema de información.

Link de acceso:

<http://asovapp.tech/>

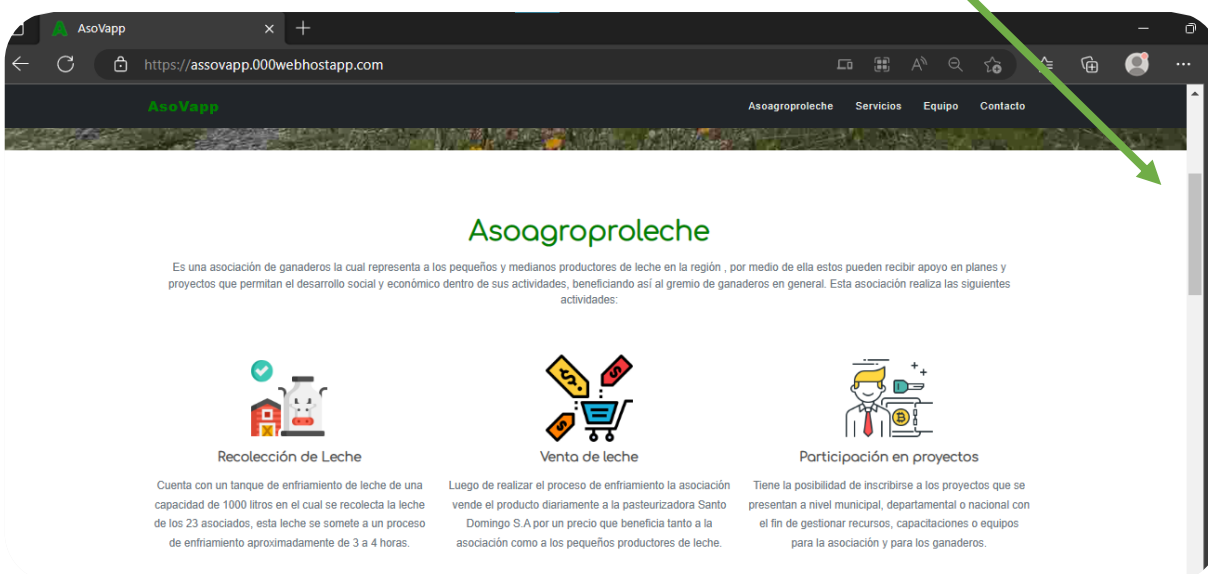


Encontrará un

carrusel el cual muestra una descripción de cada uno de los módulos del sistema



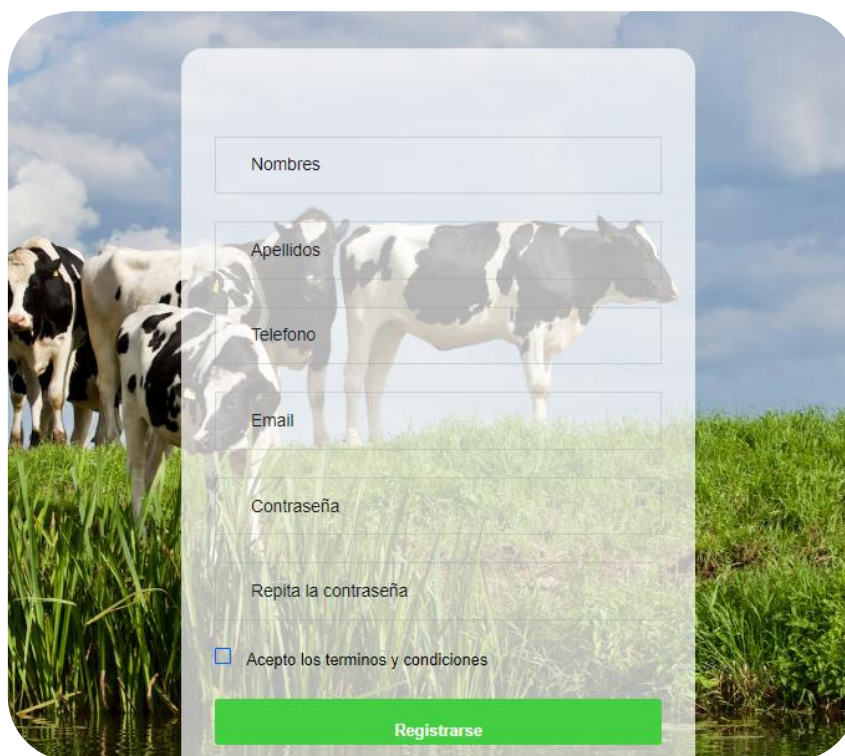
- En el botón de iniciar sesión encontrará el inicio de sesión.
- En el botón de crear cuenta encontrará el formulario para el respectivo registro.
- Al navegar en las siguientes secciones encontrará información de los servicios que ofrece el sistema de información. A continuación, se mostrará las vistas de la página de inicio de Asovapp.



Encontrará fragmentos de texto para contextualizarlo o familiarizarlo con el sistema.

- **Módulo inicio de sesión**

Encontrará el formulario que contiene campos obligatorios como lo son registrar los respectivos nombres y apellidos, teléfono, un correo electrónico y una respectiva contraseña.

A registration form is overlaid on a background image of a cow field. The form consists of several input fields and a checkbox. The fields are labeled: 'Nombres', 'Apellidos', 'Telefono', 'Email', 'Contraseña', and 'Repita la contraseña'. Below these fields is a checkbox labeled 'Acepto los terminos y condiciones'. At the bottom of the form is a green button labeled 'Registrarse'.

Nombres

Apellidos

Telefono

Email

Contraseña

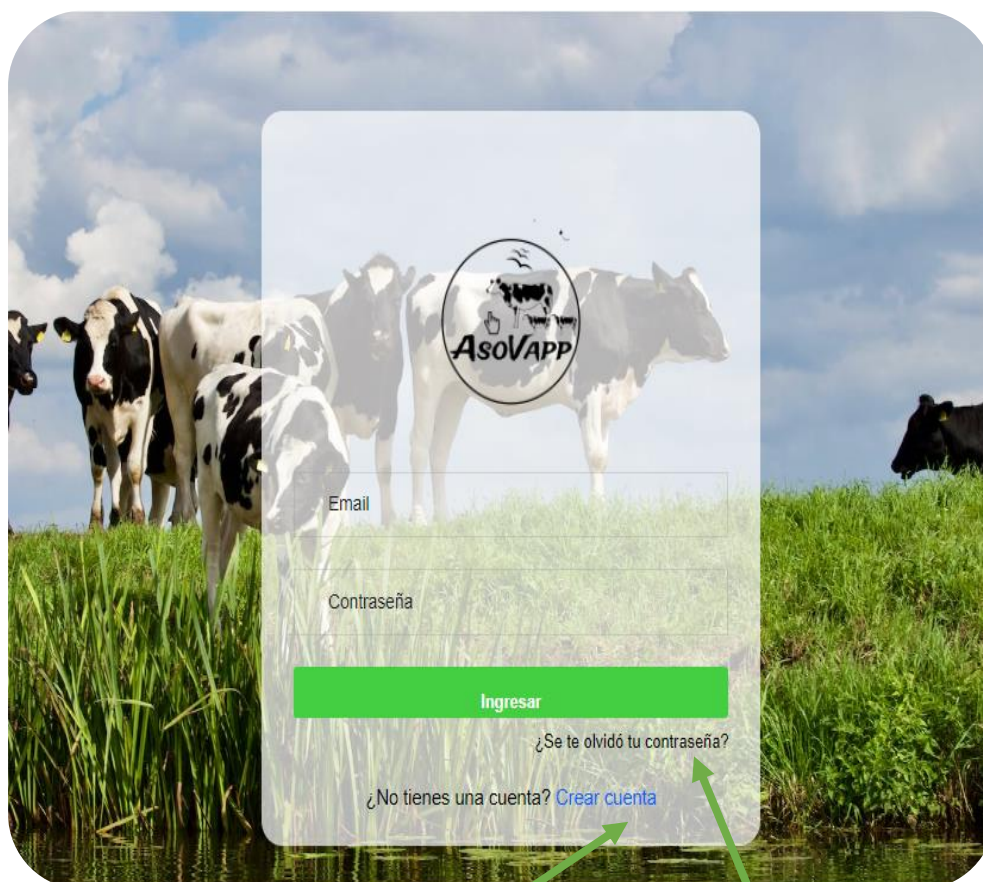
Repita la contraseña

Acepto los terminos y condiciones

**Registrarse**

Deberá aceptar los términos y condiciones para poder registrarse en el software y poder disfrutar de los servicios que este ofrece.

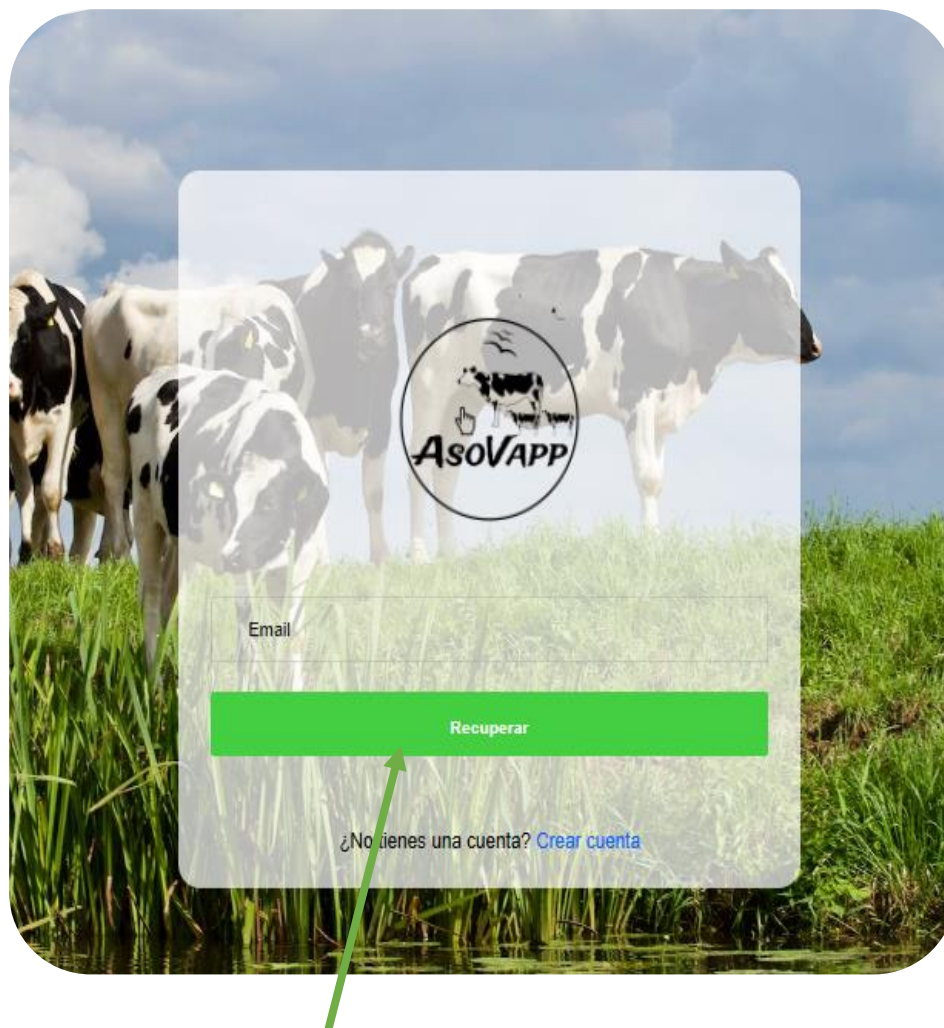
- Luego de registrarse será redirigido al formulario de ingreso los campos de correo electrónico y la respectiva contraseña.



Si no está registrado podrá dar click aquí

En caso de que olvide su contraseña podrá  
recuperarla dando click aquí

Lo llevara a la siguiente pantalla:



Deberá ingresar el correo con el cual se registró donde le llegará un mensaje con la nueva contraseña para su ingreso.

- **Módulo de presidente**

Aquí encontrará la página de inicio donde podrá seleccionar por medio de los dos recuadros la fecha de la quincena que quiera ver, mostrando así las respectivas estadísticas de la producción lechera que se ha obtenido en ese rango de tiempo.



Otra de las opciones es que también puede ver información detallada de cada uno de los socios como:

- El estado en el que se encuentra activo es decir actualmente se está recibiendo producción lechera de ese socio.
- Inactivo es decir que fue parte de la asociación en algún momento, pero actualmente no lleva producción lechera.

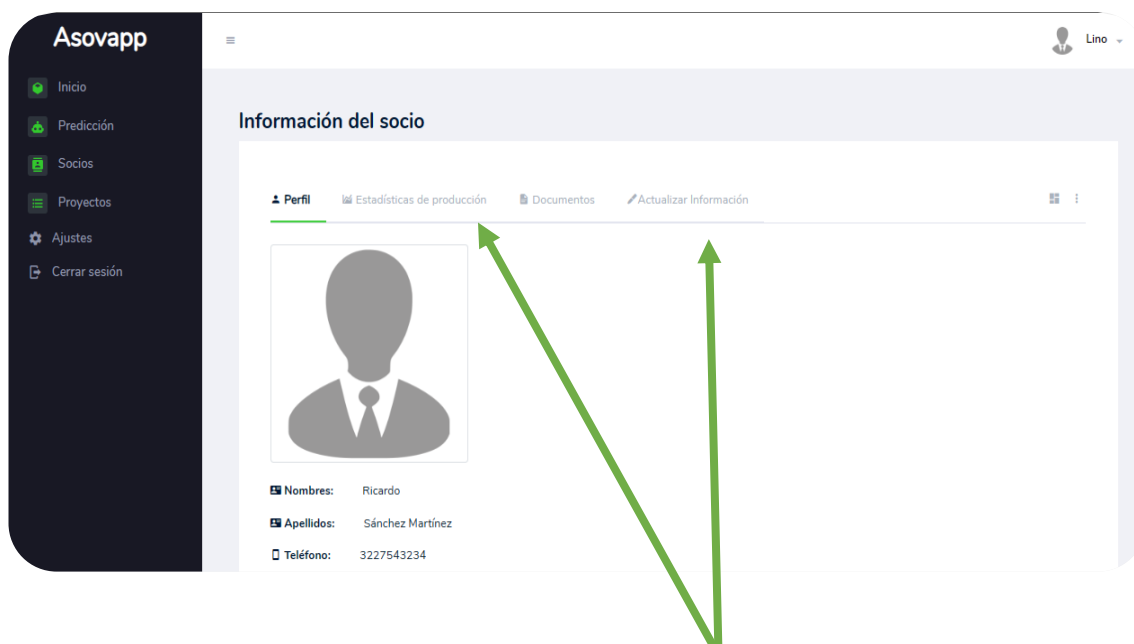
- Ver el perfil como los nombres, apellidos, correo, teléfono.
- Podrá ver las estadísticas de la producción lechera de cada socio
- Revisar que los documentos de registro de finca y certificado de vacunas este al día es decir actualizado a la fecha.

En este podrá buscar por iniciales o por números del socio y le traerá todos los datos.

The screenshot shows the Asovapp mobile application interface. On the left is a dark sidebar with menu items: Inicio, Predicción, Socios (highlighted), Proyectos, Ajustes, and Cerrar sesión. The main content area displays a list of members with columns for Nombre, Apellidos, Teléfono, Estado, Rol, and Acción. A search bar is located at the top right of the list area, and a 'Ver perfil' button is highlighted for the member Ricardo Sánchez Martínez.

Nombre	Apellidos	Teléfono	Estado	Rol	Acción
Lino	Guzmán	3227299457		Presidente	Ver perfil
Ana Maria	Castiblanco	1234567890		Admin	Ver perfil
Ricardo	Sánchez Martínez	3227543234		Socio	Ver perfil
Emilio	Perez Sanchez	32456663789		Socio	Ver perfil
Favio Ignacio	Castiblanco	3214657643		Socio	Ver perfil

Al dar click aquí podrá visualizar el perfil del socio.

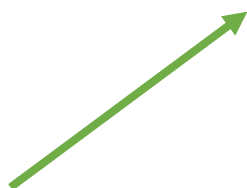


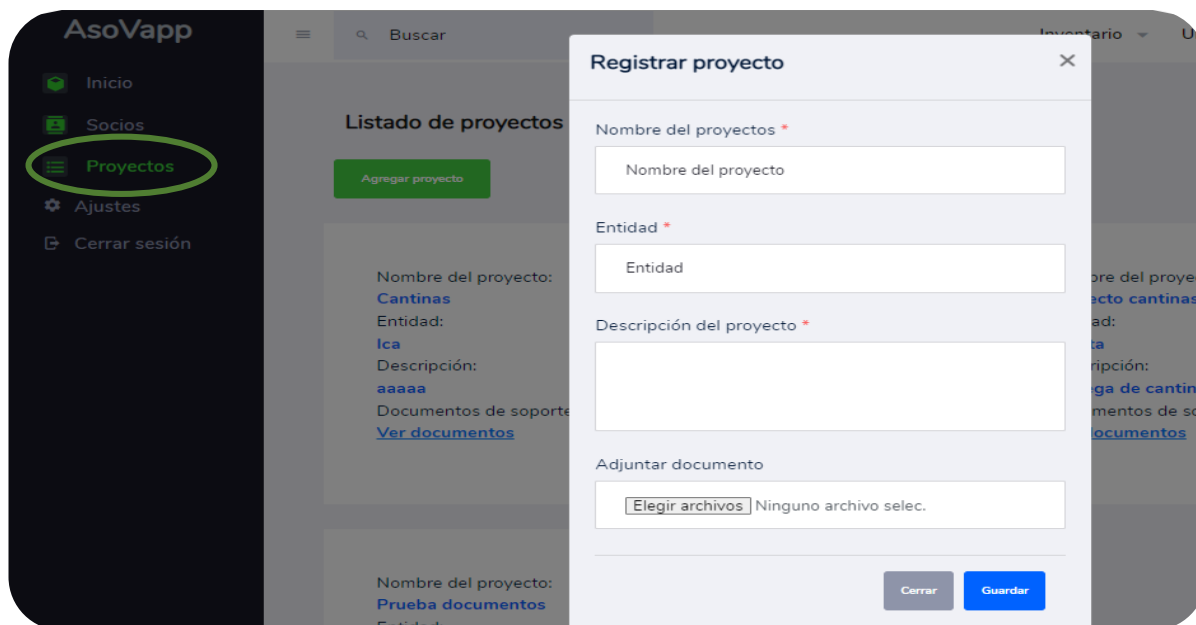
Por medio de estas secciones podrá ver

En la siguiente opción encontrará información más detallada del socio de cada uno de los proyectos que participa como asociación ya sea capacitaciones.

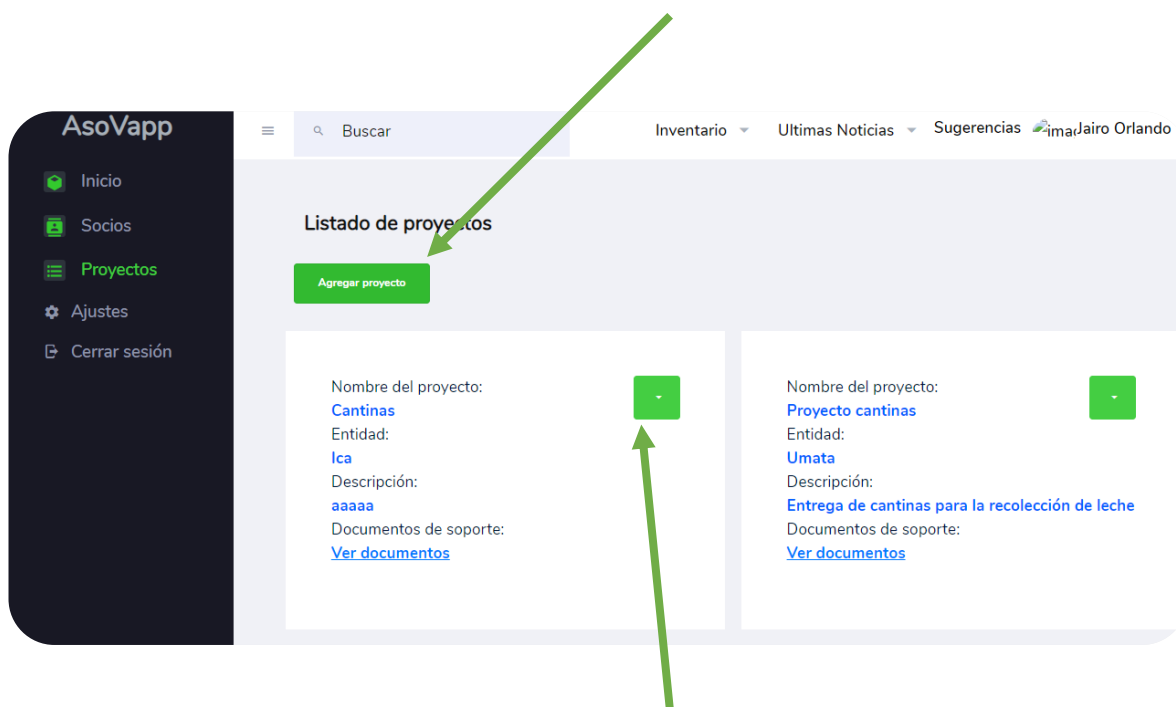
- Aquí podrá ingresar el nombre del proyecto
- La entidad en la que participó.
- La descripción o la nota que desee ingresar.
- Y finalmente podrá adjuntar los respectivos documentos de soporte.
- Así mismo podrá ver, editar, eliminar los datos ingresados.

Al dar click aquí encontrará la sección proyectos.

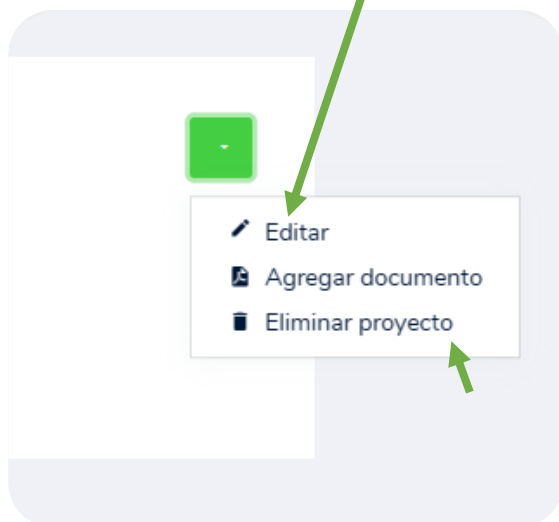




En esta sección encontrará todos los proyectos que ha registrado, y así mismo podrá registrar nuevos proyectos por medio del botón *agregar proyecto*.



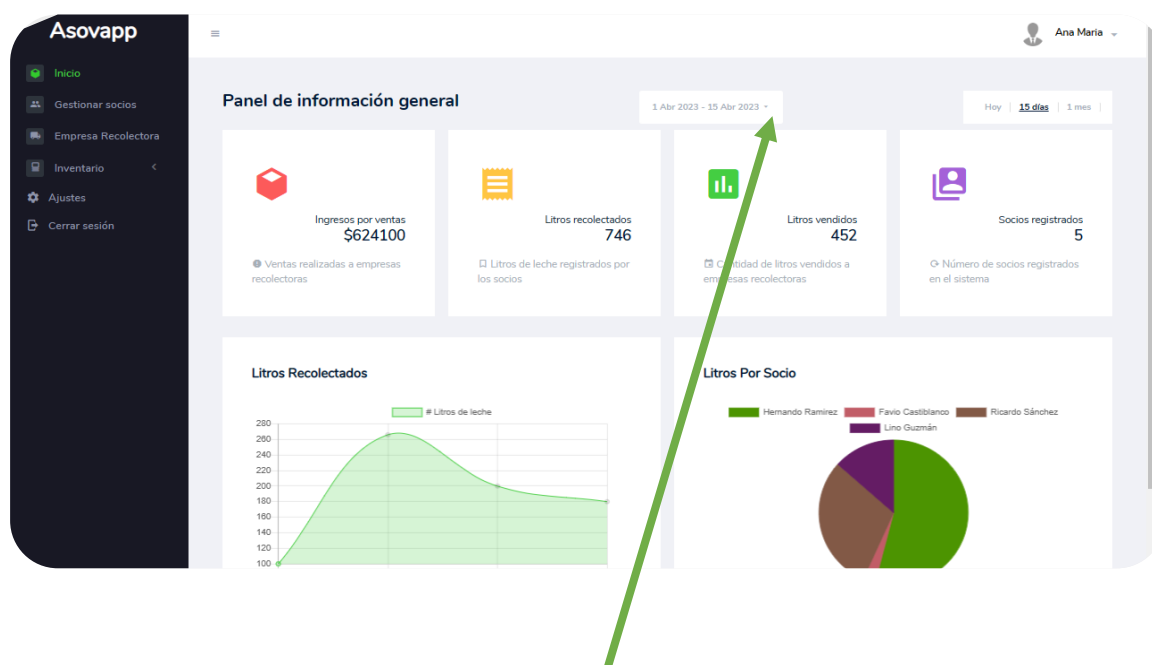
Al dar click aquí le mostrará las siguientes opciones



- Podrá cambiar cualquier dato que hubiese ingresado erróneamente.
- Agregar más documentos de soporte y evidencia.
- Le permite ingresar solo documentos pdf o Word

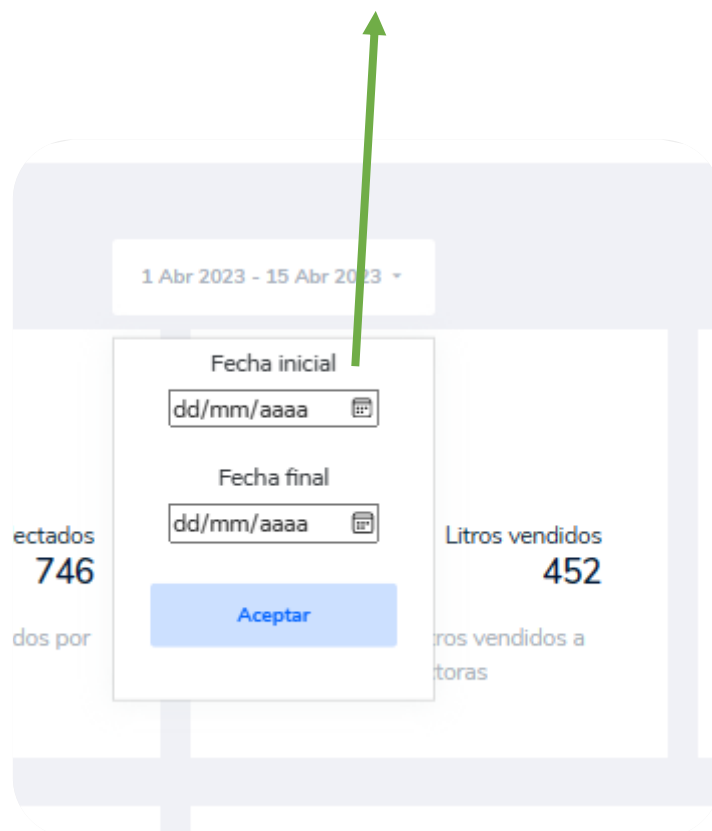
## • Módulo administradora

- Dadas las respectivas credenciales por parte del administrador.
- Al iniciar sesión correctamente encontrará un panel el cual le dará el resumen de la actividad reciente, como los litros de la producción lechera que ha registrado en ese día o días.



Podrá seleccionar el rango de la quincena que desea ver dando click aquí

Se desplegará la pestaña para seleccionar una fecha inicial y final en la que desea realizar revisión general de los datos ingresados.



The image shows a modal window for date selection. At the top, there is a header with the text "1 Abr 2023 - 15 Abr 2023" and a dropdown arrow. Below this, the modal contains two input fields: "Fecha inicial" and "Fecha final", both with the placeholder "dd/mm/aaaa" and a calendar icon. A blue "Aceptar" button is positioned at the bottom of the modal. The background is a blurred screenshot of a dashboard with the following visible text: "ectados 746", "dos por", "Litros vendidos 452", and "ros vendidos a toras". A green arrow points from the "Fecha inicial" field to the text "Podrá seleccionar el rango de la quincena que desea ver dando click aquí" above the image.

- La cantidad de socios registrados.
- Los ingresos que se han obtenido en las ventas de la producción lechera.
- También como se evidencia anteriormente podrá ver graficas en diferentes formatos para mostrar los litros recolectados por todos los socios y los litros de cada uno.

- Otra de las opciones es la de gestión de los socios, es decir, aquí podrá realizar lo siguiente:
  - Podrá agregar, editar, eliminar, actualizar los datos de los socios
  - También en caso de que algún socio se retire podrá cambiar el estado a inactivo lo cual no le permite realizar algún registros o acción dentro del sistema.

Mostrar 10 registros

Buscar:

Nombre	Apellidos	Teléfono	Estado	
Ricardo	Sánchez Martínez	3227543234		<a href="#">Ver perfil</a>
Lino	Guzmán	3227299457		<a href="#">Ver perfil</a>
Jairo Orlando	Pascagaza Barriga	3227299457		<a href="#">Ver perfil</a>
Hernando	Ramirez Carrillo	3216754901		<a href="#">Ver perfil</a>

### Información del socio

[Perfil](#) [Registrar Producción](#) [Registros de producción](#) [Documentos](#)

**Nombres:** Yeimy Constansa

**Apellidos:** Angel Sánchez

**Teléfono:** 3227543234

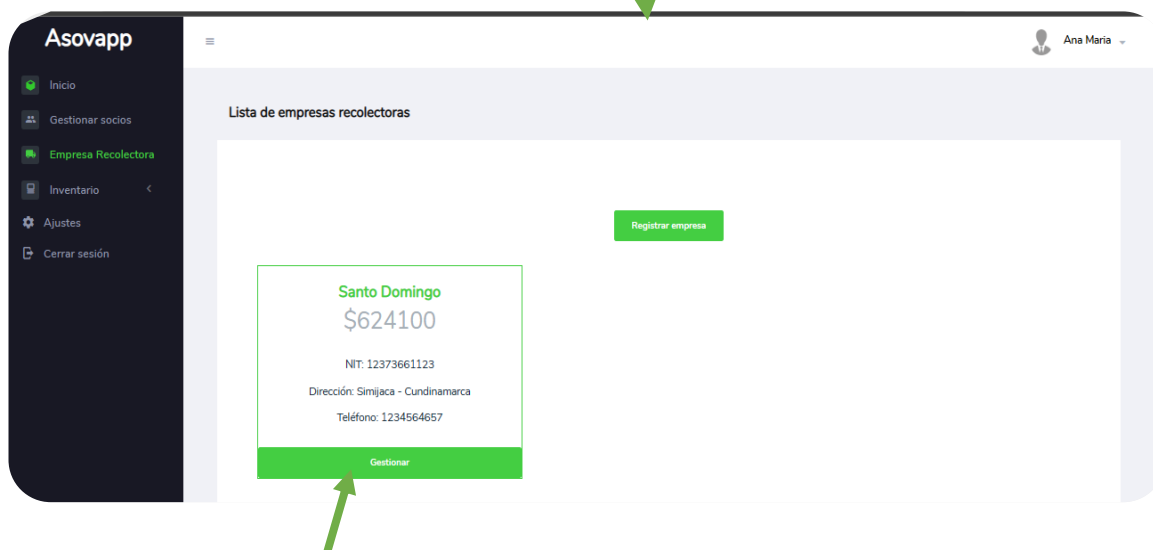
**Correo Electrónico:** [Yeimy@gmail.com](mailto:Yeimy@gmail.com)

**Estado:** Inactivo

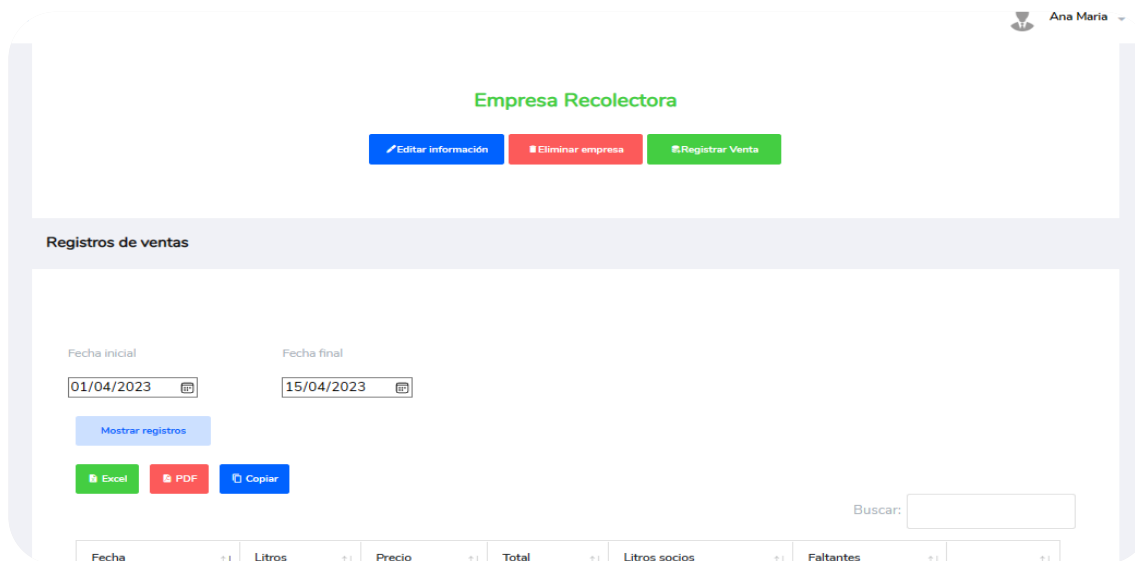
Como administradora podrá navegar por las distintas pestañas para realizar la acción que desee.

En la opción de empresa recolectora podrá registrar la información de la empresa que recolecta la información por medio del botón *agregar empresa*:

- Nit
- Dirección
- Documentos de soporte



Al dar click en el botón de gestionar encontraras la siguiente sección.



Permite también registrar cada uno de los elementos que usa en la asociación ya se cantinas, filtros, elementos para tomar las pruebas de la leche.

- Este es el formulario el cual podrá ingresar cada uno de los datos del artículo que vaya a registrar como:
  - Nombre del artículo
  - Cantidad
  - Descripción
  - La imagen que desee ingresar

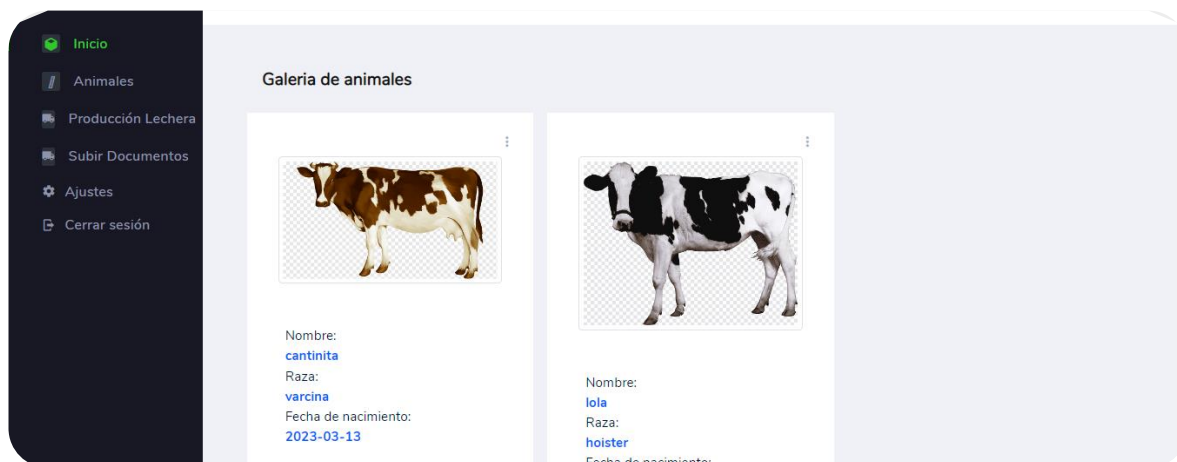


Al dar click en ver  
artículos mostrara la foto con  
los respectivos datos.



- **Módulo de socio**

Al iniciar sesión se encontrará con la galería de fotos y datos de animales que han sido registrados.



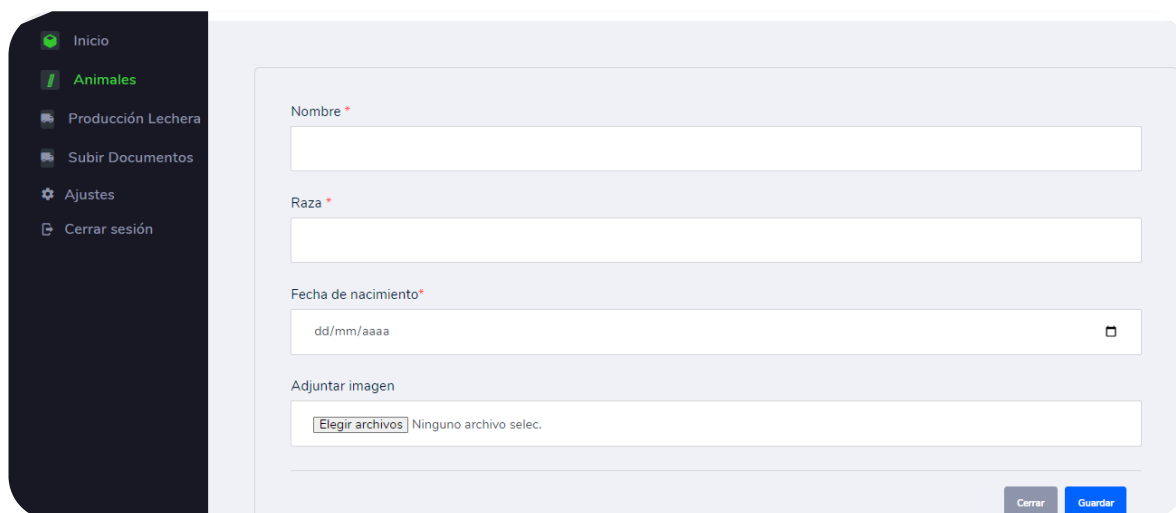
Al dar click en la opción de animales encontrará este formulario podrá ingresar:

- El nombre de su animal
- La raza
- La fecha de nacimiento y la respectiva foto



The screenshot shows a mobile browser interface with a URL bar displaying 'p.000webhostapp.com'. The form is titled 'Asovapp' and contains the following fields: 'Nombre \*' (text input), 'Raza \*' (text input), and 'Fecha de nacimiento\*' (text input). Below these is a section for 'Adjuntar imagen' with a file selection button labeled 'Elegir archivos' and the text 'Sin archivos seleccionados'. At the bottom of the form are two buttons: 'Cerrar' and 'Guardar'.

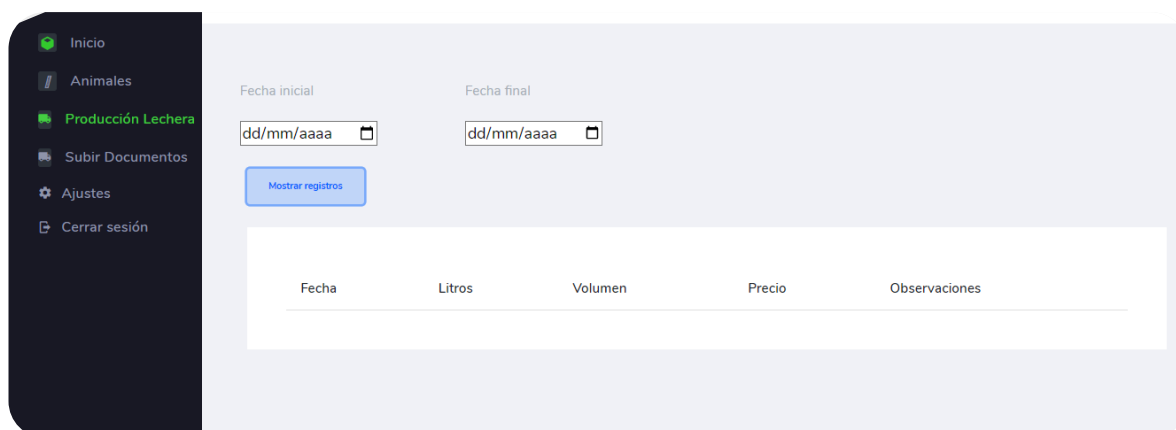
Ya ingresados correctamente los datos, dará click en el botón guardar, dado el caso que hubieras ingresado mal los datos podrán dar click en el botón cancelar.



The screenshot shows a sidebar menu on the left with options: Inicio, Animales (highlighted), Producción Lechera, Subir Documentos, Ajustes, and Cerrar sesión. The main content area contains a form with the following fields: 'Nombre \*' (text input), 'Raza \*' (text input), 'Fecha de nacimiento\*' (date picker with 'dd/mm/aaaa' placeholder), and 'Adjuntar imagen' (file upload area with 'Elegir archivos' button and 'Ninguno archivo selec.' text). At the bottom right of the form are 'Cerrar' and 'Guardar' buttons.

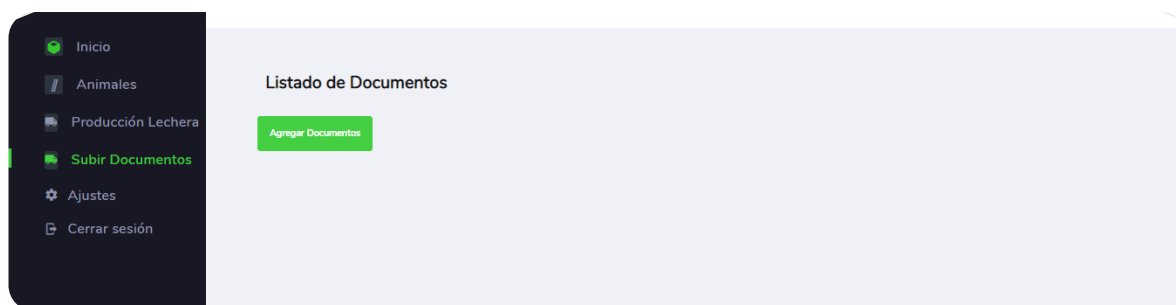
En la opción de producción lechera podrá seleccionar las fechas de la quincena que desea revisar de la producción lechera que ha sido registrada por la administradora de tanque.

- Podrá seleccionar el rango de los quince días los cuales quiere verificar la cantidad de litros de la producción lechera.



The screenshot shows the sidebar menu with 'Producción Lechera' highlighted. The main content area features two date pickers labeled 'Fecha inicial' and 'Fecha final', both with 'dd/mm/aaaa' placeholders. Below them is a blue 'Mostrar registros' button. Underneath is a table with the following headers: 'Fecha', 'Litros', 'Volumen', 'Precio', and 'Observaciones'. The table body is currently empty.

En la opción de subir documentos dando click en el botón agregar documentos.



Al instante aparecerá el formulario donde deberá ingresar:

- Una breve descripción de los documentos que está subiendo
- La fecha y los respectivos documentos como lo es el certificado de vacunas y el registro de finca.



- **Ajustes de todos los módulos**

Todos los módulos cuentan con la opción de ajustes de perfil y cerrar sesión.

En esta sección podrá colocar una foto de perfil en caso de que no desee no será obligatorio.

The screenshot shows the 'Actualizar imagen de perfil' (Update profile picture) section. On the left is a dark sidebar with menu items: Inicio, Animales, Producción Lechera, Subir Documentos, Ajustes, and Cerrar sesión. The main content area has a header 'Actualizar imagen de perfil' and a placeholder image of a person. Below the image are two buttons: 'Cambiar foto' (Change photo) and 'Eliminar foto' (Remove photo).

En la sección de actualizar los datos que desee menos el correo electrónico.

The screenshot shows the 'Actualizar datos personales' (Update personal data) section. It contains four input fields: 'Nombres:' with the value 'Yeimy', 'Apellidos:' with the value 'Angel', 'Correo Electrónico:' with the value 'socio2@socio', and 'Telefono:' with the value '1234567890'. Below these fields is a green 'Actualizar' (Update) button. Below the update section is the 'Cambiar contraseña' (Change password) section, which has a label 'Contraseña Actual:' and an input field with the placeholder 'Ingrese Contraseña'.

Y finalmente podrá cambiar la contraseña, para poder hacerlo deberá confirmar la contraseña actual y así ingresar la contraseña nueva para guardar los nuevos datos deberá dar click en el botón de cambiar contraseña.

## **Manual del programador**

Yeimy Constanza Ángel Sánchez y Jairo Orlando Pascagaza Barriga

Programa de ingeniería de sistemas, Universidad de Cundinamarca

Trabajo de grado

Director

Cristian Eduardo Cano López

Junio 2,2023

## Índice

Introducción .....	33
Objetivo.....	34
Herramientas .....	35
HTML     35	
JavaScript 35	
PHP .....	36
MySQL.....	37
Visual Studio Code .....	37
Funcionalidades del sistema.....	38
Configuración del entorno de desarrollo 38	
Conexión a la base de datos     39	
Registro de usuarios     40	
Inicio de sesión     44	
Recuperar contraseña     47	
Módulo del presidente.....	52
Página de inicio     52	
Módulo de socio.....	63
Página de inicio del socio 64	
Módulo de administrador .....	70

## Índice de figuras

<b>Figura 22</b> Configuración del entorno de desarrollo. ....	38
<b>Figura 23</b> Conexión a la base de datos. ....	40
<b>Figura 24</b> Registro de usuarios vista. ....	41
<b>Figura 25</b> Registro de usuarios validación de campos. ....	42
<b>Figura 26</b> Registro de usuarios controlador. ....	43
<b>Figura 27</b> Registro de usuarios modelo. ....	43
<b>Figura 28</b> Inicio de sesión vista. ....	44
<b>Figura 29</b> Inicio de sesión modelo. ....	46
<b>Figura 30</b> Recuperar contraseña JavaScript primera parte. ....	48
<b>Figura 31</b> Recuperar contraseña JavaScript segunda parte. ....	48
<b>Figura 32</b> Recuperar contraseña controlador. ....	49
<b>Figura 33</b> Recuperar contraseña modelo. ....	49
<b>Figura 34</b> Recuperar contraseña JavaScript tercera parte. ....	50
<b>Figura 35</b> Recuperar contraseña controlador. ....	50
<b>Figura 36</b> Recuperar contraseña modelo. ....	51
<b>Figura 37</b> Vista del menú de opciones. ....	52
<b>Figura 38</b> Validación del rol. ....	53
<b>Figura 39</b> Diseño de la vista del contenido. ....	54
<b>Figura 40</b> Función para graficar los datos. ....	55
<b>Figura 41</b> Funcion para graficar. ....	56
<b>Figura 42</b> Producción lechera controlador. ....	57

<b>Figura 43</b> Función para obtener documentos.....	58
<b>Figura 44</b> controlador de los certificados.....	59
<b>Figura 45</b> Modelo de los proyectos.....	60
<b>Figura 46</b> Controlador de Proyectos .....	61
<b>Figura 47</b> vista de cerrar sesión.....	62
<b>Figura 48</b> Página de Inicio .....	64
<b>Figura 49</b> Controlador de animal .....	65
<b>Figura 50</b> Modelo de Animal.....	66
<b>Figura 51</b> JavaScript para mostrar galería.....	67
<b>Figura 52</b> JavaScript para editar galería.....	68
<b>Figura 53</b> Archivo para editar la foto.....	69
<b>Figura 54</b> Vista inicio administrador .....	70
<b>Figura 55</b> Archivo js inicio administrador parte 1 .....	71
<b>Figura 56</b> Archivo js inicio administrador parte 2 .....	71
<b>Figura 57</b> Controlador inicio administrador.....	72
<b>Figura 58</b> Modelo inicio administrador .....	73
<b>Figura 59</b> Vista registro de producción.....	73
<b>Figura 60</b> Archivo js registro de producción .....	74
<b>Figura 61</b> Controlador registro de producción.....	75
<b>Figura 62</b> Modelo registrar producción .....	76
<b>Figura 63</b> Vista mostrar producción.....	77
<b>Figura 64</b> Mostrar producción JavaScript.....	78
<b>Figura 65</b> Mostrar producción controlador.....	79

<b>Figura 66</b> Mostrar producción modelo .....	80
<b>Figura 67</b> Registrar salidas Vista .....	81
<b>Figura 68</b> Registrar salidas JavaScript.....	82
<b>Figura 69</b> Registrar salidas controlador .....	82
<b>Figura 70</b> Registrar salidas controlador .....	83

## **Introducción**

Este documento proporciona una guía detallada para aquellos que deseen entender y modificar el código fuente de la página web asovapp. Aquí se encontrará información detallada acerca de la construcción del software, se mostrarán las diferentes herramientas utilizadas y se realizará una breve descripción acerca de las estructuras de código más relevantes en el desarrollo.

**Objetivo**

Documentar de manera clara y precisa la estructura, funcionamiento y arquitectura utilizada en el desarrollo de la página web asovapp, con el fin de brindar una herramienta que les sirva a los programadores a entender la lógica utilizada y les permita hacer modificaciones de manera sencilla.

## Herramientas

### HTML

HTML es un lenguaje de marcado utilizado para crear y estructurar contenidos en la web, utiliza etiquetas para identificar diferentes tipos de contenido, como encabezados, párrafos, listas, enlaces y más. Las etiquetas son códigos que se escriben en un documento HTML para indicar al navegador web cómo debe interpretar el contenido. En la creación de la página web, se utilizó html para la adaptación del frontend, es decir, la parte en la que el usuario interactúa directamente con la página web. Debido a que la arquitectura que se utilizó fue MVC (Modelo, vista, controlador), sirvió como herramienta para la creación de las vistas.



*Nota.* Logo oficial del lenguaje de marcado html5. W3.org, 2023, ([https://www.w3.org/html/logo/downloads/HTML5\\_Logo.svg](https://www.w3.org/html/logo/downloads/HTML5_Logo.svg)). CC

### JavaScript

JavaScript es un lenguaje de programación interpretado que se utiliza principalmente en el desarrollo de aplicaciones web, se utiliza para agregar interactividad y dinamismo a las páginas web, permitiendo a los desarrolladores crear efectos visuales, animaciones, validaciones de formularios y otras características interactivas en la página web. JavaScript se ejecuta en el lado del cliente, es decir, en el navegador web del usuario final. Esto significa que el código

JavaScript se descarga desde el servidor web al navegador del usuario y se ejecuta en su máquina local.



*Nota.* Logo no oficial del lenguaje de programación JavaScript. Chris Williams, 2023, (<https://github.com/voodootikigod/logo.js>). CC

## **PHP**

PHP es un lenguaje de programación de código abierto y de propósito general ampliamente utilizado en la creación de aplicaciones web y sitios web dinámicos. PHP se ejecuta en el servidor web y se utiliza principalmente para generar contenido dinámico, como páginas web generadas por bases de datos y formularios de contacto. En la página web asovapp, se utilizó este lenguaje de programación para consultas a la base de datos, debido a la arquitectura utilizada para el desarrollo, este lenguaje se utilizó para definir los modelos y el controlador.



*Nota.* Logo oficial del lenguaje de programación PHP. php.net, 2023, (<https://www.php.net/download-logos.php>). CC

## MySQL

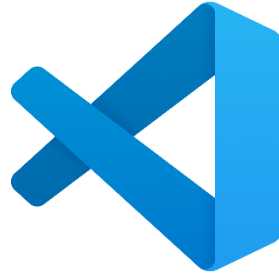
MySQL es un sistema de gestión de bases de datos relacionales (RDBMS) de código abierto que permite almacenar, organizar y acceder a grandes cantidades de información de manera eficiente. Es una herramienta poderosa y versátil que ha demostrado ser confiable y eficiente en una amplia variedad de casos de uso.



*Nota.* Logo oficial del gestor de base de datos MySQL. [mysql.com](https://www.mysql.com/about/legal/logos.html), 2023, (<https://www.mysql.com/about/legal/logos.html>). CC

## Visual Studio Code

Visual Studio Code (VS Code) es un editor de código fuente gratuito y de código abierto desarrollado por Microsoft. Se ha convertido en uno de los editores de código más populares y utilizados en la actualidad debido a su facilidad de uso, eficiencia y gran cantidad de extensiones disponibles. Una de las características más atractivas de VS Code es su gran cantidad de extensiones, que permiten agregar funcionalidades adicionales y personalizar el editor según las necesidades del usuario. Por las características mencionadas anteriormente se hizo la elección de este editor para el desarrollo de la web asovapp.



*Nota.* Logo oficial del editor de código Visual Studio Code. visualstudio, 2023, (<https://code.visualstudio.com/brand>). CC

### **Funcionalidades del sistema**

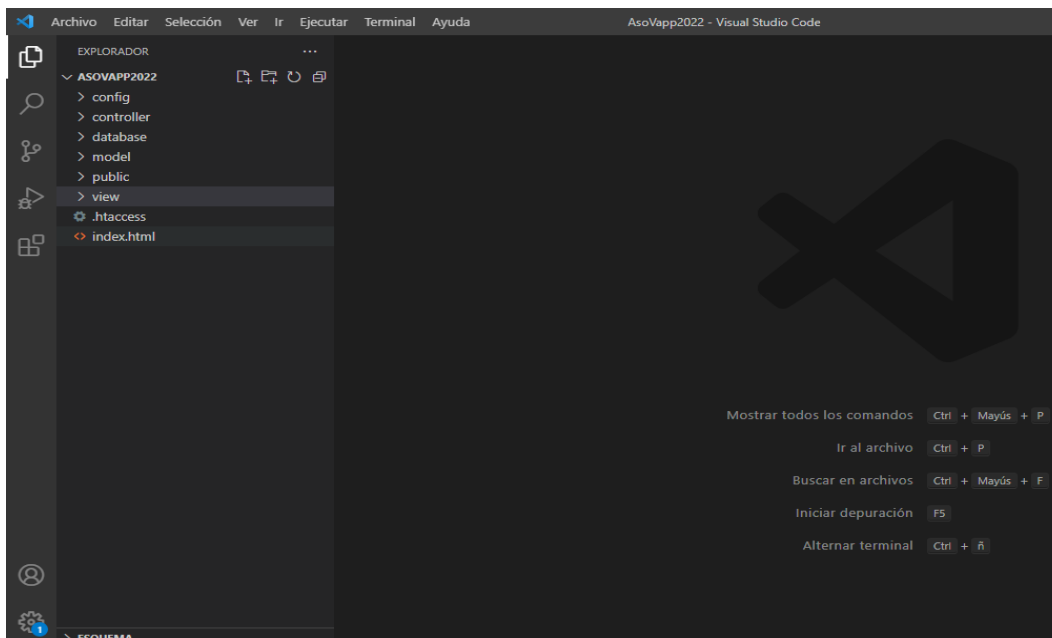
El desarrollo de la aplicación web AsoVapp, se ha venido realizando de manera modular, y por medio de la arquitectura MVC (Modelo, Vista, Controlador). Se han definido los lenguajes de programación con los cuales se va a realizar el proceso de desarrollo y son php, javascript, html y css.

A continuación, se mostrarán imágenes del código desarrollado y se explicará de manera breve la funcionalidad de este.

### **Configuración del entorno de desarrollo**

#### **Figura 22**

*Configuración del entorno de desarrollo.*



*Nota.* El entorno de desarrollo se ha configurado en Visual Studio Code.

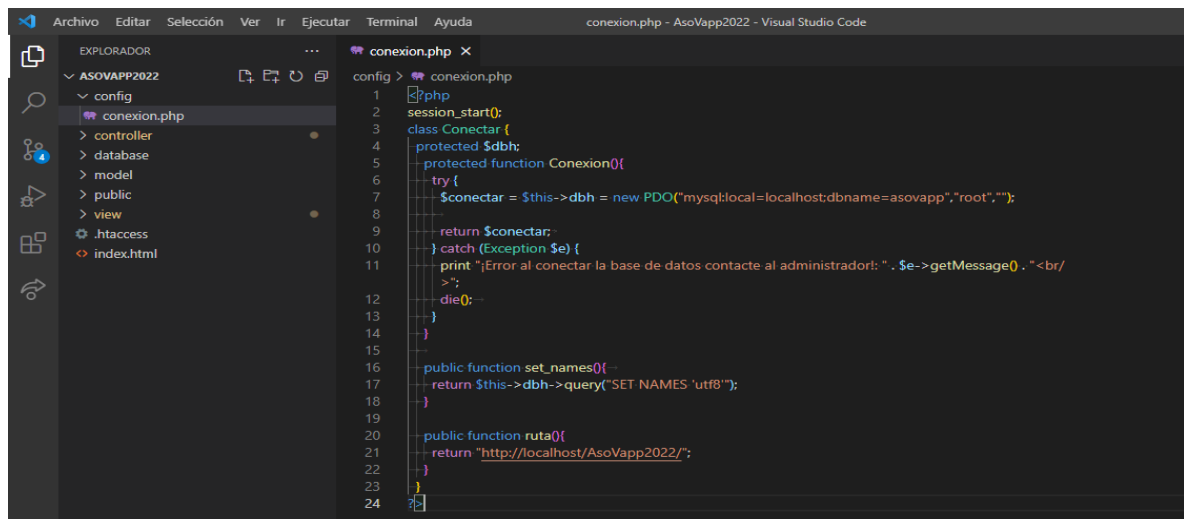
y se ha sincronizado con los integrantes del equipo de trabajo por medio de un repositorio creado en GitHub, como se puede observar se crearon diferentes carpetas que son utilizadas para realizar el desarrollo de manera modular por medio de la arquitectura MVC. Se crearon 3 carpetas principales, la carpeta model que contiene diferentes archivos escritos en php, que sirven para la comunicación entre la vista y el controlador, la carpeta view contiene archivos html con la parte grafica de los diferentes módulos que componen el sistema y finalmente la carpeta controller que se utiliza para almacenar los archivos php que realizan las consultas a bases de datos y la comunicación con el servidor.

### **Conexión a la base de datos**

La conexión a la base de datos se realiza desde un archivo nombrado conexión, en el cual se establece la clase conectar, en esta clase se define una función nueva llamada Conexión en la cual se define una variable que contiene una serie de parámetros necesarios para realizar la conexión como los son el nombre del host, el nombre de la base de datos y la contraseña.

## Figura 23

*Conexión a la base de datos.*



```

1  <?php
2  session_start();
3  class Conectar {
4      protected $dbh;
5      protected function Conexion0{
6          try {
7              $sconectar = $this->dbh = new PDO("mysql:host=localhost;dbname=asovapp","root","");
8          }
9          return $sconectar;
10         } catch (Exception $e) {
11             print "¡Error al conectar la base de datos contacte al administrador: ". $e->getMessage0 . "<br/>";
12             die0;
13         }
14     }
15
16     public function set_names0{
17         return $this->dbh->query("SET NAMES 'utf8'");
18     }
19
20     public function ruta0{
21         return "http://localhost/AsoVapp2022/";
22     }
23 }
24

```

*Nota.* La conexión a la base de datos se hace con php y se crea la clase conectar, en la cual se especifica el nombre de la base de datos y la contraseña.

## Registro de usuarios

La creación o registro de usuarios nuevos es fundamental para el funcionamiento del sistema, por lo tanto, fue necesario implementarlo por medio de la arquitectura mvc.

Inicialmente se crearon los archivos necesarios para trabajar, los cuales son una carpeta nueva dentro de la carpeta principal view que se llamó SignUp y en la cual se crearon dos archivos, uno llamado index.php que contiene la parte grafica de la aplicación y un archivo

llamado `signup.js`, el cual contiene el código JavaScript necesario para que funcione la vista del usuario.

## Figura 24

### *Registro de usuarios vista*

```
<form class="pt-3" id="registro">
  <div class="form-group">
    <input type="text" class="form-control form-control-lg" id="nombre" name="nombre"
      placeholder="Nombres" required>
    </div>
  <div class="form-group">
    <input type="text" class="form-control form-control-lg" id="apellido" name="apellido"
      placeholder="Apellidos" required>
    </div>
  <div class="form-group">
    <input type="number" class="form-control form-control-lg" id="telefono" name="telefono"
      placeholder="Telefono" required>
    </div>
  <div class="form-group">
    <input type="email" class="form-control form-control-lg" id="correo" name="correo"
      placeholder="Email" required>
    </div>
  <div class="form-group">
    <input type="password" class="form-control form-control-lg" id="contrasena" name="contrasena"
      placeholder="Contraseña" required>
    </div>
  <div class="form-group">
    <input type="password" class="form-control form-control-lg" id="contrasena2" name="contrasena2"
      placeholder="Repita la contraseña" required>
    </div>
  <div class="mb-4">
    <div class="form-check">
      <label class="form-check-label text-muted">
        <input type="checkbox" name="terms" class="form-check-input"> Acepto los terminos y condiciones </
        label>
      </div>
    </div>
</form>
```

*Nota.* Se crea un formulario con los campos necesarios, en este caso se solicitan cinco datos personales. Y se exige la validación para aceptar los términos y condiciones del servicio.

**Figura 25***Registro de usuarios validación de campos*

```
function guardarusuarios() {
  let datosformulario = new FormData($("#registro")[0]);
  $.ajax({
    url: "../../controller/usuario.php?op=guardar",
    type: "POST",
    data: datosformulario,
    contentType: false,
    processData: false,
    success: function (datos) {
      if (datos == "pass") {
        Swal.fire(
          'AsoVapp',
          'Contraseña Incorrecta no coinciden los campos',
          'error'
        );
      } else if (datos == "correo") {
        Swal.fire(
          'AsoVapp',
          'Correo electrónico ya registrado',
          'error'
        );
      } else if (datos == "ok") {
        Swal.fire(
          'AsoVapp',
          'Se registro Correctamente',
          'success'
        ).then((result) => {
          if (result.isConfirmed) {
            window.location = "../Login";
          }
        });
        $("#registro")[0].reset();
      }
    }
  });
}
```

*Nota.* Se validan los datos ingresados por el usuario, a través del archivo signup.js, y si los datos son correctos, se envían los parámetros al archivo controlador llamado usuario.php, esto por medio de ajax.

**Figura 26***Registro de usuarios controlador*

```
case "guardar":
    $datos = $usuario->get_correo_usuario($_POST["correo"]);
    if($_POST["contrasena"] == $_POST["contrasena2"]){
        if(is_array($datos)==true and count($datos)==0){
            $usuario->insert_usuario($_POST["nombre"],$_POST["apellido"],$_POST["telefono"],$_POST["correo"],
            $_POST["contrasena"]);
            echo "ok";
        }else{
            echo "correo";
        }
    }else{
        echo "pass";
    }
}
```

*Nota.* Se reciben los parámetros provenientes de la vista y se valida si el correo ya está registrado, si no está registrado, se envían los parámetros al modelo.

**Figura 27***Registro de usuarios modelo*

```

public function insert_usuario($nombre,$apellido,$telefono,$correo,$contrasena){
    $hash=password_hash($contrasena, PASSWORD_DEFAULT);
    $conectar= parent::conexion();
    parent::set_names();
    $sql="INSERT INTO usuario values (NULL,?,?,?,?,?,?,?,?)";
    $sql=$conectar->prepare($sql);
    $sql->bindValue(1,$nombre);
    $sql->bindValue(2,$apellido);
    $sql->bindValue(3,$telefono);
    $sql->bindValue(4,$correo);
    $sql->bindValue(5,$hash);
    $sql->bindValue(6,3);
    $sql->bindValue(7,1);
    $sql->bindValue(8,'avatar.png');
    $sql->execute();
}

```

*Nota.* Se reciben los parámetros provenientes del controlador a través de una función llamada insert\_usuario, luego se realiza el proceso de encriptación de contraseña y finalmente se establece la conexión con la base de datos y se almacenan. Cabe destacar que se maneja un avatar por defecto hasta que el usuario modifique ese campo en el perfil.

### **Inicio de sesión**

El inicio de sesión es un procedimiento en el cual el usuario se identifica por medio de un correo y una contraseña previamente registrado en el sistema, con el fin de ingresar y hacer uso de este, cabe destacar que la aplicación web está compuesta por diferentes roles del sistema, con lo cual las funciones que encontrarán dependerán del rol con el que se haya registrado, dichos roles pueden ser asignados por el administrador del sistema.

Para el desarrollo de este módulo igual que en el módulo de registro fue necesario crear una carpeta dentro de view llamada login, en la cual se crea el archivo con la parte grafica denominado index.php, luego se solicitan los datos de usuario y contraseña para validar y dar acceso al usuario, esto por medio del archivo Usuario.php que se encuentra en la carpeta model.

### **Figura 28**

*Inicio de sesión vista*

```

<form class="pt-3" method="post" id=loginnu1>
  <div class="form-group">
    <input type="email" class="form-control form-control-lg" id="correo" name="correo" placeholder="Email">
  </div>
  <div class="form-group">
    <input type="password" class="form-control form-control-lg" id="password" name="password"
      placeholder="Contraseña">
  </div>
  <div class="mt-3">
    <input type="hidden" name="enviar" class="form-control" value="si">
    <button type="submit" class="btn btn-block btn-success btn-lg font-weight-medium
      auth-form-btn">Ingresar</button>
  </div>
  <div class="my-2 d-flex justify-content-between align-items-center">
    <div class="form-check">
      <label class="form-check-label text-muted">
    </div>
    <a href=" ../RecuperarContrasena" class="auth-link text-black">¿Se te olvidó tu contraseña?</a>
  </div>
  <div class="text-center mt-4 font-weight-light"> ¿No tienes una cuenta? <a href=" ../SignUp"
    class="text-primary">Crear cuenta</a>
  </div>
</form>

```

*Nota.* Este es el código HTML utilizado para mostrar el formulario de inicio de sesión al usuario, en este se encuentran los campos de correo electrónico y contraseña que el usuario debe proporcionar.

Figura 29

*Inicio de sesión modelo*

```

public function login()
{
    $conectar=parent::Conexion();
    parent::set_names();
    if(isset($_POST["enviar"])){
        $password = $_POST["password"];
        $correo = $_POST["correo"];
        if(empty($correo) and empty($password)){
            header("Location:".Conectar::ruta()."view/Login/index.php?m=2");
            exit();
        }
    }
    else {
        $sql= "select contrasena from usuario where correo=?";
        $sql=$conectar->prepare($sql);
        $sql->bindValue(1, $correo);
        $sql->execute();
        $resultado = $sql->fetch();
        $hash=$resultado["contrasena"];
        if (password_verify($password, $hash)) {
            header("Location:".Conectar::ruta()."view/Inicio/");

            $sql= "select * from usuario where correo=? and contrasena=?";
            $sql=$conectar->prepare($sql);
            $sql->bindValue(1, $correo);
            $sql->bindValue(2, $hash);
            $sql->execute();
            $resultado = $sql->fetch();
            $_SESSION["id_usuario"] = $resultado["id_usuario"];
            $_SESSION["nombre"] = $resultado["nombre"];
            $_SESSION["apellido"] = $resultado["apellido"];
            $_SESSION["correo"] = $resultado["correo"];
            $_SESSION["rol"] = $resultado["rol_id"];
            $_SESSION["avatar"] = $resultado["avatar"];
            $rol=(int)$_SESSION["rol"];
            if($rol==1){
                header("Location:".Conectar::ruta()."view/HomePresidente/");
            }else if($rol==2){
                header("Location:".Conectar::ruta()."view/HomeAdministrador/");
            }else if($rol==3){
                header("Location:".Conectar::ruta()."view/HomeSocio/");
            }
            exit();
        }
    }
    header("Location:".Conectar::ruta()."view/Login/index.php?m=1");
    exit();
}
}

```

*Nota.* La imagen anterior contiene el código para iniciar sesión en el sistema, primero que todo en el archivo Usuario.php perteneciente a model, se declara una función llamada login la cual recibe dos parámetros, el correo electrónico y la contraseña, primero se hace una validación del correo, ya que debe estar registrado, luego se verifica que el correo y la contraseña coincida con

lo registrado en la base de datos. Por ultimo redirige a vistas diferentes dependiendo el rol que tenga el usuario dentro del sistema, se obtienen las variables de sesión.

## Recuperar contraseña

Debido a que el aplicativo web cuenta con validación por correo y contraseña, fue importante desarrollar un módulo que permita recuperar la cuenta por medio del correo electrónico, debido a que el usuario puede olvidar la contraseña de ingreso al sistema. Para esto se utilizó una librería de PHP llamada PHPmailer, la cual facilita el envío de correos electrónicos.

A continuación, se mostrará el procedimiento que se llevó a cabo para desarrollar esta funcionalidad.

Primero que todo se realizó el maquetado del formulario en el cual se solicita al usuario el correo electrónico que registró previamente en el sistema y se crea el botón recuperar.

```
<form class="pt-3" method="post" id=loginnum1>
  <div class="form-group">
    <input type="email" class="form-control form-control-lg" id="usu_correo" name="usu_correo"
      placeholder="Email">
    </div>
    <div class="mt-3">
      <button type="button" id="btnrecuperar" class="btn btn-block btn-success btn-lg font-weight-medium
        auth-form-btn">Recuperar</button>
    </div>
    <div class="my-2 d-flex justify-content-between align-items-center">
      <div class="form-check">
        <label class="form-check-label text-muted">
        </div>
      </div>
      <div class="text-center mt-4 font-weight-light"> ¿No tienes una cuenta? <a href=" ../SignUp"
        class="text-primary">Crear cuenta</a>
      </div>
    </form>
```

*Nota.* HTML del formulario de recuperación de contraseña

Luego por medio de JavaScript, se valida que el campo de correo electrónico no esté vacío, y se inicializa una función de espera que muestra un mensaje al usuario que se está procesando la solicitud como se muestra a continuación.

### Figura 30

*Recuperar contraseña JavaScript primera parte.*

```

$(document).on("click", "#btnrecuperar", function() {
    Swal.fire({
        title: 'Espere mientras procesamos su solicitud',
        html: '<div class="spinner-border text-primary" role="status"><span class="sr-only">Loading...</span></div>',
        showConfirmButton: false,
        allowOutsideClick: false
    });

    timeoutId = setTimeout(() => {
        Swal.fire({
            title: 'Tiempo de espera agotado',
            text: 'Ha transcurrido 1 minuto sin recibir una respuesta del servidor',
            icon: 'warning'
        });
    }, 60000);

    var usu_correo = $('#usu_correo').val();

    if(usu_correo == ""){
        Swal.close();
        clearTimeout(timeoutId);
        Swal.fire(
            'Asovapp',
            'Error correo vacio',
            'error'
        );
    }
}

```

*Nota.* Se muestra la primera parte del proceso luego que el usuario de click en el botón recuperar.

Luego se realiza la validación de que el correo electrónico este registrado como se muestra a continuación.

### Figura 31

*Recuperar contraseña JavaScript segunda parte*

```

const respuestaCorreo=$.post("../controller/usuario.php?op=correo", { usu_correo : usu_correo}, function(data){
    if (data=='[]'){
        Swal.close();
        clearTimeout(timeoutId);
        Swal.fire(
            'Asovapp',
            'Correo electrónico no Registrado',
            'error'
        );
    }else{

```

*Nota.* Se envía el correo electrónico que proporcionó el usuario al controlador, y del controlador al modelo con el fin de saber si el correo está registrado,

A continuación, se muestra el código del modelo y controlador respectivamente.

### Figura 32

*Recuperar contraseña controlador*

```
case "correo":
    $datos = $usuario->get_correo($_POST["usu_correo"]);
    echo json_encode( $datos);
break;
```

*Nota.* se recibe el parámetro correo proveniente de la vista recuperar contraseña.

### Figura 33

*Recuperar contraseña modelo*

```
function get_correo($correo){
    $conectar= parent::conexion();
    parent::set_names();
    $sql="SELECT nombre, apellido, correo FROM usuario WHERE correo=?";
    $sql=$conectar->prepare($sql);
    $sql->bindValue(1,$correo);
    $sql->execute();
    return $resultado=$sql->fetchAll();
}
```

*Nota.* Se crea una función que retorna los registros de la base de datos que coinciden con la consulta.

Luego que se valida que el correo está registrado se procede a crear un archivo en carpeta controller llamado email.php que será el encargado de comunicar la vista con el modelo como se muestra a continuación.

## Figura 34

*Recuperar contraseña JavaScript tercera parte.*

```
    }else{
      const respuesta = $.post("../controller/email.php?op=send_recuperar", { usu_correo : usu_correo},
      function(data){
        $('#usu_correo').val('');
        console.log(data);

        if(data=="ok"){
          clearTimeout(timeoutId);
          Swal.close();
          // Muestra una nueva alerta de éxito
          Swal.fire({
            title: '¡Éxito!',
            text: 'Se ha enviado un correo electrónico con la nueva contraseña',
            icon: 'success',
            allowOutsideClick: false
          }).then((result) => {
            if (result.value) {
              window.location.replace('../Login');
            }
          });
        }else{
          Swal.close();
          clearTimeout(timeoutId);
          // Muestra una alerta de error
          Swal.fire({
            title: 'Error',
            text: 'Se ha producido un error contacte con el administrador',
            icon: 'error'
          });
        }
      }
    }
  }
}
```

*Nota.* Se envía el parámetro de correo al controlador que realiza el proceso de envío con el modelo y se envía o no el correo al usuario, luego que el servidor ejecute las instrucciones correspondientes se muestra un aviso al usuario de éxito o fracaso en la solicitud, esto por medio de una alerta perteneciente a la librería sweet alert 2.

## Figura 35

*Recuperar contraseña controlador*

```

<?php
require_once("../config/conexion.php");
require_once("../model/Email.php");
$email = new Email();

switch($_GET["op"]){

    case "send_recuperar":
        $respuesta=$email->recuperar($_POST["usu_correo"]);
        echo $respuesta;
        break;
}
?>

```

*Nota.* Se muestra la función para poder recuperar contraseña. .

### Figura 36

#### *Recuperar contraseña modelo*

```

public function recuperar($usu_correo){
    $mail = new PHPMailer(true);
    $usuario = new Usuario();

    try {
        $datos = $usuario->get_correo($usu_correo);
        $nuevaContraseña=rand(1000,1000000);

        foreach ($datos as $row) {
            $nom = $row["nombre"].' '.$row["apellido"];
        }
        $nuevaPass = $usuario->recuperar_contraseña($usu_correo,$nuevaContraseña);
        // Variables exigidas por PHPmailer
        $mail->isSMTP();
        $mail->Host = 'smtp.gmail.com';
        $mail->SMTPAuth = true;
        $mail->Username = 'pascagazajairo96@gmail.com';
        $mail->Password = 'athaorrrrrrrrr';
        $mail->SMTPSecure = PHPMailer::ENCRYPTION_STARTTLS;
        $mail->Port = 587;

        $mail->setFrom('pascagazajairo96@gmail.com', 'Asovapp');
        $mail->addAddress($usu_correo, $nom);
        $mail->CharSet='UTF8';
        $mail->isHTML(true);
        $mail->Subject = 'Recuperación contraseña Asovapp';
        $cuerpo = file_get_contents('../public/recuperar/recuperar.html');
        $cuerpo = str_replace('lblnomx',$nom,$cuerpo);
        $cuerpo = str_replace('lblpassx',$nuevaContraseña,$cuerpo);
        $mail->Body = $cuerpo;
        $mail->send();
        //Fin del uso de PHPmailer

        echo 'ok';
    } catch (Exception $e) {
        echo 'fallo';
    }
}

```

*Nota.* Se muestra la creación de la función “recuperar” que recibe como parámetro el correo que ya se validó previamente, luego se crea una contraseña temporal de un número de siete cifras, y se envía a la base de datos en donde se encripta. Posteriormente se envía el correo al usuario en el cual se le indica su nueva contraseña, cabe destacar que esto se realiza utilizando la librería PHPmailer y que todas las variables que se muestran en la parte final del código corresponden a exigencias de la librería para el correcto funcionamiento.

### **Módulo del presidente**

#### **Página de inicio**

La vista del inicio del módulo del presidente primero consta del menú de opciones, es decir, se crea una carpeta en la que se guardara cada uno de los sidebar de los respectivos módulos, en el cual está el sidebar del presidente donde consta de cinco opciones como se muestra a continuación. El id de la etiqueta li se encarga de que cuando de muestre el contenido al cual lo va a direccionar tome un color verde indicando que se encuentra en esa ventana la cual solicito por medio del href el atributo que contiene la ruta de la ubicación del archivo que desea mostrar.

#### **Figura 37**

*Vista del menú de opciones*

```

<nav class="sidebar sidebar-offcanvas" id="sidebar">
  <ul class="nav">
    <!-- <li class="nav-item nav-category">Main</li> -->
    <li class="nav-item" id="estadoInicio">
      <a class="nav-link" href=" ../HomePresident">
        <span class="icon-bg"><i class="mdi mdi-cube menu-icon"></i></span>
        <span class="menu-title">Inicio</span>
      </a>
    </li>
    <li class="nav-item" id="socios">
      <a class="nav-link" href=" ../Socios">
        <span class="icon-bg"><i class="mdi mdi-contacts menu-icon"></i></span>
        <span class="menu-title">Socios</span>
      </a>
    </li>
    <li class="nav-item" id="proyectos">
      <a class="nav-link" href=" ../Proyectos">
        <span class="icon-bg"><i class="mdi mdi-format-list-bulleted menu-icon"></i></span>
        <span class="menu-title">Proyectos</span>
      </a>
    </li>
    <li class="nav-item sidebar-user-actions">
      <div class="sidebar-user-menu">
        <a href=" ../SettingsP" class="nav-link"><i class="mdi mdi-settings menu-icon"></i>
        <span class="menu-title">Ajustes</span>
      </a>
      </div>
    </li>
    <li class="nav-item sidebar-user-actions">
      <div class="sidebar-user-menu">
        <a href=" ../Logout/logout.php" class="nav-link"><i class="mdi mdi-logout menu-icon"></i>
        <span class="menu-title">Cerrar sesión</span></a>
      </div>
    </li>
  </ul>
</nav>

```

*Nota.* Se muestra el código que redirección cada uno de los contenidos que el presidente navegue.

### Figura 38

#### Validación del rol

```

<?php
  require_once("../../config/conexion.php");
  if(isset($_SESSION["nombre"])&(int)$_SESSION["rol"]==1){
  ?>
</DOCTYPE html>

```

*Nota.* Indica el acceso que permitirá navegar según el rol en este caso el uno que es solamente para mostrar contenido al presidente.

## Figura 39

### Diseño de la vista del contenido

```

<div class="d-xl-flex justify-content-between align-items-start">
  <h2 class="text-dark font-weight-bold mb-2">Bienvenido señor presidente</h2>
  <div class="d-sm-flex justify-content-xl-between align-items-center mb-2">
  </div><br>
<div class="col-lg-12 grid-margin stretch-card">
  <div class="card">
    <div class="card-body">
      <h2 class="font-weight mb-2">Estadísticas de la asociación</h2>
      <div class="form-group row">
        <div class="form-group row">
          <label for="inicio" class="text-dark col-sm-5 col-form-label">Fecha inicial</label>
          <div class="col-sm-12">
            <input type="date" id="inicio" value="<?php echo date("Y-m-d");?>">
          </div>
        </div>
        <div class="form-group row">
          <label for="inicio" class="text-dark col-sm-5 col-form-label">Fecha final</label>
          <div class="col-sm-12">
            <input type="date" id="fin" value="<?php echo date("Y-m-d");?>">
          </div>
        </div>
        <div class="col-sm-12">
          <button type="button" onclick="datosEstadisticas();" id="graficar" class="btn btn-inverse-primary btn-fw">Mostrar gráfico</button>
        </div>
      </div>
      <div class="row">
        <div class="col-sm-6">
          <div class="text-center">
            <h4>litros totales</h4>
          </div>
        </div>
      </div>
    </div>
  </div>
</div>
<div class="row">
  <div class="col-sm-6">
    <div class="text-center">
      <h4>litros totales</h4>
    </div>
  </div>
</div>

```

*Nota.* Se muestra las funciones creadas en los archivos de js que se llaman para mostrar los datos de la quincena seleccionada por el presidente en estadísticas.

El presidente deberá elegir el rango de las fechas de la quincena entonces por medio del atributo value se le asigna la fecha de inicio y fin de los 15 días que desea graficar.

**Figura 40**

*Función para graficar los datos*

```
function datosEstadisticas() {
  var FechaInicial = $('#inicio').val();
  var FechaFinal = $('#fin').val();

  if (FechaInicial == "" || FechaFinal == "") {

    Swal.fire(
      'AsoVapp',
      'Error debe completar todos los campos',
      'error'
    );

  } else {
    if (FechaInicial > FechaFinal) {

      Swal.fire(
        'AsoVapp',
        'Error rango de fecha incorrecto',
        'error'
      );

    } else {
      $.post("../controller/produccion.php?op=produccionTotal", {
        FechaInicial: FechaInicial,
        FechaFinal: FechaFinal
      }, function (data) {
        let fechas = [];
        let litros = [];
        let precios = [];
        let preciotal = [];
        data = JSON.parse(data);

        for (let step = 0;
            (step < data.aaData.length); step++) {
```

*Nota.* Se muestra las variables que construirán la graficas en el rango de fechas que seleccione.

Para poder realizar las gráficas se hace uso de la librería **Chart.js** que proporciona gran variedad de gráficos de código abierto.

### Figura 41

*Funcion para graficar*

```
datosEstadisticas();
function graficar(fecha, litros, precio) {
  let pagos = sumarArray(precio);
  const nuevo = litros.map((i) => Number(i));
  let totalLitros = sumarArray(nuevo);

  let litrostot = document.getElementById('litrost');
  litrostot.innerHTML = totalLitros;
  let dinero = document.getElementById('dinero');
  dinero.innerHTML = '$' + pagos;

  const ctx = document.getElementById('lineCharts');
  const myChart = new Chart(ctx, {
    type: 'line',
    data: {
      labels: fecha,
      datasets: [{
        label: '# Litros de leche',
        data: litros,
        backgroundColor: [
          'rgb(51, 201, 45 , 0.2)'
        ],
        borderColor: [
          'rgba(51, 201, 45)'
        ],
        borderWidth: 1
      }]
    },
    options: {
      scales: {
        y: {
          beginAtZero: true
        }
      }
    }
  });
}
```

*Nota.* Se muestra el código de la función que va a realice la gráfica según los datos registrados en esos quince días.

La función consta de tres parámetros como fecha, litros y precio los cuales almacenan cada uno de los datos para generar el total del precio de la producción lechera de todos los socios en el rango de fechas seleccionadas.

La función **sumarArray** está pasando el parámetro precio. Luego se convierte cada elemento del **array** litros en un número utilizando la función **map** y se almacena en una nueva variable llamada nuevo. A continuación, se suma los valores de la nueva variable nuevo utilizando la función **sumarArray** y se almacena el resultado en la variable **totalLitros**.

## Figura 42

### *Producción lechera controlador*

```

controller > production.php > ...
1  k?php
2  require_once("../config/conexion.php");
3  require_once("../model/Produccion.php");
4  $produccion = new Produccion();
5
6  switch($_GET["op"]){
7
8      case "dataProduccion":
9          $datos = $produccion->data_Produccion($_POST["IdUsuario"],$_POST["FechaInicial"],$_POST["FechaFinal"]);
10         $data= Array();
11         foreach($datos as $row){
12             $sub_array = array();
13             $sub_array["Cantidad"] = $row["cantidad"];
14             $sub_array["Fecha"] = $row["fecha"];
15             $sub_array["Precio"] = $row["precio"];
16             $data[] = $sub_array;
17         }
18         $results = array(
19             "sEcho"=>1,
20             "iTotalRecords"=>count($data),
21             "iTotalDisplayRecords"=>count($data),
22             "aaData"=>$data);
23         echo json_encode($results);
24
25         break;
26         case "produccionTotal":
27             $datos = $produccion->produccion_Total($_POST["FechaInicial"],$_POST["FechaFinal"]);
28             $data= Array();
29             foreach($datos as $row){

```

*Nota.* Se muestran cada uno de los casos que cumplen con una función en específica al ejecutar la sección del código en específico.

Como se ve en la figura anterior se crea una variable llamada **\$datos** que se le asigna el valor que devuelve la función **data\_Produccion()**.

La función **data\_Produccion()** consta de tres parámetros, que se envían mediante el método **\$\_post** donde estos valores se usan para realizar la respectiva consulta en las secciones que permiten mostrar datos ya sea en las gráficas o datos de los usuarios en general.

### Figura 43

*Función para obtener documentos*

```
model > Certificado.php > ...
1  <?php
2
3  class Certificado extends Conectar{
4      public function obtener_Certificados($IdUsuario){
5          $conectar=parent::conexion();
6          parent::set_names();
7          $sql="SELECT * FROM certificado_vacuna WHERE idUsuario=? ORDER BY id_certificado DESC";
8          $sql=$conectar->prepare($sql);
9          $sql->bindvalue(1,$IdUsuario);
10         $sql->execute();
11         return $resultado=$sql->fetchall(PDO::FETCH_ASSOC);
12     }
13 }
14 ?>
15
```

*Nota.* Se muestra el SQL select que mostrara los certificados de vacunas que han registrados los socios.

Figura 44

*controlador de los certificados*

```

controller > certificado.php > ...
1  <?php
2  require_once("../config/conexion.php");
3  require_once("../model/Certificado.php");
4  $certificado = new Certificado();
5
6  switch($_GET["op"]){
7
8      case "obtenerCertificados":
9          $datos=$certificado->obtenerCertificados($_GET["IdUsuario"]);
10         $data= Array();
11         foreach($datos as $row){
12             $sub_array = array();
13             $sub_array["Documento"] = $row["documento"];
14             $sub_array["Fecha"] = $row["fecha"];
15             $sub_array["Descripcion"] = $row["descripcion"];
16             $data[] = $sub_array;
17         }
18
19         $results = array(
20             "sEcho"=>1,

```

*Nota.* Se muestran los arrays en los que se almacenaran cada una de los parámetros.

La función **obtenerCertificados** se pasa el valor de **\$\_post** como argumento, donde finalmente se almacena en la variable **\$datos**.

Donde luego se crea el array **\$data** para almacenar cada uno de los parámetros como lo son los documentos ,descripciones y fechas en las que se realicen registros por parte del presidente.

Figura 45

*Modelo de los proyectos*

```

model > Proyecto.php > ...
1  <?php
2
3  class Proyecto extends Conectar{
4      public function insert_proyecto($Nombre,$Entidad,$Descripcion,$IdUsuario){
5          $conectar= parent::conexion();
6          parent::set_names();
7          $sql="INSERT INTO proyecto VALUES (NULL,?,?,?,?)";
8          $sql=$conectar->prepare($sql);
9          $sql->bindValue(1,$Nombre);
10         $sql->bindValue(2,$Entidad);
11         $sql->bindValue(3,$Descripcion);
12         $sql->bindValue(4,$IdUsuario);
13         $sql->execute();
14         $sql1="SELECT MAX(id_proyecto) FROM proyecto;";
15         $sql1=$conectar->prepare($sql1);
16         $sql1->execute();
17         return $resultado=$sql1->fetchall(pdo::FETCH_ASSOC);
18     }
19     public function actualizar($Nombre,$Entidad,$Descripcion,$IdProyecto){
20         $conectar= parent::conexion();
21         parent::set_names();
22         $sql="UPDATE proyecto SET nombre=?, entidad=?, descripcion=? WHERE id_proyecto=?";
23         $sql=$conectar->prepare($sql);
24         $sql->bindValue(1,$Nombre);
25         $sql->bindValue(2,$Entidad);
26         $sql->bindValue(3,$Descripcion);

```

*Nota.* Se muestra la clase proyecto y el método **insert\_proyecto** que consta de cuatro parámetros que se conectan en la base de datos.

Se puede visualizar el código de la estructura SQL en donde se insertarán cada uno de los datos de los proyectos de la asociación.

Figura 46

*Controlador de Proyectos*

```

controller > proyecto.php > ...
1  <?php
2  require_once("../config/conexion.php");
3  require_once("../model/Proyecto.php");
4  $proyecto = new Proyecto();
5
6  switch($_GET["op"]){
7
8      case "anadirproyecto":
9          $datos=$proyecto->insert_proyecto($_POST["nombreProyecto"],$_POST["entidadEncargada"],$_POST["descripcionProyecto"]);
10         if(is_array($datos)==true and count($datos)>0){
11             foreach($datos as $row)
12             {
13                 $output["IdProyecto"] = $row["MAX(id_proyecto)"];
14             }
15             echo json_encode($output);
16         }
17         break;
18     case "editarproyecto":
19         $datos=$proyecto->edit_proyecto($_POST["nombreProyecto"],$_POST["entidadEncargada"],$_POST["descripcionProyecto"]);
20         echo "ok";
21         break;
22     case "insertdetalle":
23         $datos=$proyecto->insert_partesdetalle($_POST["partd_file"],$_POST["fecha"],$_POST["IdProyecto"]);
24         break;
25     case "obtenerProyectos":
26         $datos=$proyecto->obtener_Proyectos($_POST["IdUsuario"]);
27         $data= Array();
28         foreach($datos as $row){
29             $sub_array = array();

```

*Nota.* El método post envía los datos de cada atributo según la primera sección de código que añade el proyecto correspondiente.

En el caso de **editarproyecto** permite poder realizar alguna modificación de uno de los campos, es decir, los atributos. Y en el de **obtenerProyectos** con el **IdUsuario** que permite traer toda la información de los documentos y descripciones que se hayan ingresado.

**Figura 47**

*vista de cerrar sesión*

```
view > LogOut > 🐞 logout.php
1  <?php
2      require_once("../../config/conexion.php");
3      session_destroy();
4      header("Location:".Conectar::ruta()."");
5      exit();
6
```

*Nota.*Se muestra la función para cerrar sesión del módulo en el que este navegando.

Se hace uso de la función `sesión_destroy`, es posible que desee redirigir al usuario a otra página para evitar que el usuario siga accediendo a la sesión destruida. Para hacer esto, puede utilizar la función `header` de PHP para enviar una cabecera HTTP de redirección como se muestra en la figura anterior.

### **Módulo de socio**

Al validar datos en la base de datos el socio es decir el ganadero podrá iniciar sesión y encontrará la página de inicio la galería con los respectivos datos de los animales de la finca que ha registrado en el sistema.

Se hace el llamado de archivos por medio de la función **require\_once** que se encarga de traer el archivo que se requiere como se muestra en la siguiente figura.

## Página de inicio del socio

### Figura 48

#### *Página de Inicio*

```
<?php
require_once("../config/conexion.php");
if (isset($_SESSION["nombre"]) & (int)$_SESSION["rol"] == 3) {
?>
<!DOCTYPE html>
<html lang="en">

<head>
<?php require_once("../MainHead/mainHead.php"); ?>
<link rel="stylesheet" href="../../public/assets/vendors/owl-carousel-2/owl.carousel.min.css">
<link rel="stylesheet" href="../../public/assets/vendors/owl-carousel-2/owl.theme.default.min.css">
</head>

<body>
<div class="container-scroller">
<?php require_once("../MainHeader/headersocio.php"); ?>

<div class="container-fluid page-body-wrapper">
<?php require_once("../MainSidebar/sidebarsocio.php"); ?>
<div class="main-panel">
<div class="content-wrapper">

<div class="page-header">
<h3 class="page-title"> Galeria de animales </h3>
</div>
<div class="row" id="animales">

</div>
</div>
<?php require_once("../MainFooter/mainFooter.php"); ?>
</div>
</div>
<?php require_once("modaleditar.php"); ?>
<?php require_once("modalfoto.php"); ?>
```

*Nota.* Se muestra el id animales que se encarga de mostrar la galería de manera organizada en columnas de cada registro que se haga por medio de un formulario.

Figura 49

*Controlador de animal*

```

controller > animal.php > ...
4  $animal = new Animal();
5
6  switch ($_GET["op"]) {
7
8      case "anadirAnimal":
9          $datos = $animal->registrarAnimal($_POST["nombreAnimal"], $_POST["raza"], $_POST["fecha"],$_PO
10
11          if (
12              $datos == "ok"
13
14          ) {
15              echo "ok";
16          } else {
17              echo $datos;
18          }
19          break;
20
21      case "obteneranimales":
22          $datos=$animal->obtener_animales($_POST["IdUsuario"]);
23          $data= Array();
24          foreach($datos as $row){
25              $sub_array = array();
26              $sub_array["nombre"] = $row["nombre"];
27              $sub_array["raza"] = $row["raza"];
28              $sub_array["fecha_nacimiento"] = $row["fecha_nacimiento"];
29              $sub_array["foto"] = $row["foto"];
30              $sub_array["Id"] = $row["id_animal"];
31              $data[] = $sub_array;
32          }

```

*Nota.* Se muestra el método GET se utiliza para enviar los datos por medio del formulario de registrar animal.

El primer caso se encarga de hacer el registro del animal, en la siguiente sección se podrá realizar la consulta por medio de una galería.

Figura 50

*Modelo de Animal*

```

model > Animal.php > ...
1  <?php
2
3  class Animal extends Conectar
4  {
5      public function registrarAnimal($nombre, $raza, $fecha_nacimiento, $IdUsuario)
6      {
7          $conectar = parent::conexion();
8          parent::set_names();
9          require_once("Animal.php");
10         $partx = new Animal();
11         $partd_file = '';
12         if ($_FILES["partd_file"]["name"] != '') {
13             $partd_file = $partx->upload_file();
14         } else {
15             $partd_file = $_POST["hidden_file_imagen"];
16         }
17         $sql = "INSERT INTO animal VALUES (NULL,?,?,?,?,?)";
18         $sql = $conectar->prepare($sql);
19         $sql->bindValue(1, $nombre);
20         $sql->bindValue(2, $raza);
21         $sql->bindValue(3, $fecha_nacimiento);
22         $sql->bindValue(4, $partd_file);
23         $sql->bindValue(5, $IdUsuario);
24

```

*Nota.* Se muestra la clase Animal y el método conectar para ingresar cada uno de los datos e imágenes con la sentencia sql.

La función **bindValue** se usó para relacionar la variable con una sentencia preparada de SQL.

Figura 51

## JavaScript para mostrar galería

```

22     for (let step = 0;
23         (step < data.aaData.length); step++) {
24         let dataAnimales = data.aaData[step];
25         let foto = dataAnimales.foto;
26         animales.innerHTML += `<div class="col-md-4 grid-margin stretch-card">
27     <div class="card">
28     <div class="d-md-block" style="padding-top: 20px;padding-right: 20px;">
29     <div class="dropdown float-right">
30     <a href="#" class="text-light p-1" dropdown-toggle type="button" id="dropdownMenuIconButton2"
31     data-toggle="dropdown" aria-haspopup="true" aria-expanded="false"><i class="mdi mdi-dots-vertical"></i></a>
32     <div class="dropdown-menu" aria-labelledby="dropdownMenuIconButton2">
33     <a class="dropdown-item text-dark"
34     href="#" onclick="editarAnimal('${dataAnimales.Id}','${dataAnimales.nombre}','${dataAnimales.raza}','${dataAnimales.fecha_nacimiento}');
35     <a class="dropdown-item text-dark" href="#" onclick="editarFoto('${dataAnimales.Id}','${foto}');"><i
36     class="mdi mdi-image mr-2"></i>Editar imagen</a>
37     <a class="dropdown-item text-dark" href="#" onclick="eliminarAnimal('${dataAnimales.Id}','${foto}');"><i
38     class="mdi mdi-delete mr-2"></i>Eliminar artículo</a>
39     </div>
40     </div>
41     </div>
42     <div class="card-body" style="padding-top: 0px;">
43     <div class="template-demo">
44     <div id="lightgallery" class="row lightGallery">
45     
46     </div>
47     <address class="text-primary">
48     <br><br>
49     <p class="text-dark">Nombre:</p>
50     <p class="font-weight-bold">${dataAnimales.nombre}</p>
51     <p class="text-dark">Raza:</p>
52     <p class="font-weight-bold">${dataAnimales.raza}</p>
53     <p class="text-dark">Fecha de nacimiento:</p>
54     <p class="font-weight-bold">${dataAnimales.fecha_nacimiento}</p>
55     </address>
56     </div>

```

*Nota.* Se muestra la sección de código que muestra los datos del animal en una galería dinámica.

Figura 52

*JavaScript para editar galería*

```

view > HomeSocio > JS editar.js > editarAnimal
1  function editarAnimal(idAnimal, nombree, razaa, fecha_nacimientooo) {
2      document.getElementById("Animal").value = idAnimal;
3      document.getElementById("editar_nombre").value = nombree;
4      document.getElementById("editar_raza").value = razaa;
5      document.getElementById("editar_fecha").value = fecha_nacimientooo;
6      $("#modal_editar_registro").modal("show");
7
8  }
9  $(document).on("click", "#guardar_animal_editado", function () {
10     Swal.fire({
11         title: 'Espere mientras procesamos su solicitud',
12         html: '<div class="spinner-border text-primary" role="status"><span class="sr-only">Loading...</span></div>',
13         showConfirmButton: false,
14         allowOutsideClick: false
15     });
16     let nombre = document.getElementById("editar_nombre").value;
17     let raza = document.getElementById("editar_raza").value;
18     let fecha_nacimiento = document.getElementById("editar_fecha").value;
19     let idAnimal = document.getElementById("Animal").value;
20     if (nombre != "" && raza != "" && fecha_nacimiento != "" && idAnimal != "") {
21         let dataAnimal = new FormData;
22         dataAnimal.append('idAnimal', idAnimal);
23         dataAnimal.append('nombre', nombre);
24         dataAnimal.append('raza', raza);
25         dataAnimal.append('fecha_nacimiento', fecha_nacimiento);
26         $.ajax({
27             url: "../../controller/animal.php?op=editar_animal",
28             type: "POST",
29             data: dataAnimal,
30             contentType: false,
31             processData: false,
32             success: function (data) {
33                 console.log(data)
34                 if (data == "ok") {
35                     mostrarAnimales().then(function (resolver) {
36                         Swal.close();

```

*Nota.* Se muestra la función que permite realizar alguna modificación en los datos registrados por el ganadero.

## Figura 53

Archivo para editar la foto

```

view > HomeSocio > modalfoto.php > ...
1
2 <div class="modal fade" id="modalfoto" tabindex="-1" role="dialog" aria-labelledby="exampleModalCenterTitle" aria-hidden="true">
3 <div class="modal-dialog modal-dialog-centered" role="document">
4 <div class="modal-content">
5 <div class="modal-header">
6 <h5 class="modal-title" id="exampleModalLongTitle">Cargar imagen</h5>
7 <input type="number" id="animalesid" style="visibility:hidden;" readonly>
8 <input type="text" id="foto" style="visibility:hidden;" readonly>
9
10 <button type="button" class="close" data-dismiss="modal" aria-label="Close">
11 <span aria-hidden="true">&times;</span>
12 </button>
13 </div>
14 <div class="modal-body">
15 <div class="d-flex justify-content-center">
16 <div class="col-8">
17 <span id="icon-cerrar"></span>
18 <img class="d-none" id="image">
19 </div>
20 </div>
21 <input type="file" class="" id="partd_file1" name="partd_file1" accept="image/*" required onchange="preview(event)">
22 <div class="modal-footer">
23 <button type="button" class="btn btn-secondary" data-dismiss="modal">Cerrar</button>
24 <button type="button" class="btn btn-primary" id="guardarimg">Guardar</button>
25 <div id="progressBar1" class="progress-bar" role="progressbar" aria-valuenow="0" aria-valuemin="0" aria-valuemax="100" style="width: 0%>0%</div>
26 </div>
27 <div class="row">
28 <div class="col-md-offset-1 col-md-3">
29 <div class="col-md-10">
30 <div class="progress">
31 <div id="progressBar1" class="progress-bar" role="progressbar" aria-valuenow="0" aria-valuemin="0" aria-valuemax="100" style="width: 0%>0%</div>
32 </div>
33 </div>
34 </div>

```

*Nota.* Se muestra la clase progress- bar que permite ver una barra de progreso en algún proceso que se este realizando.

## Módulo de administrador

En el sistema se definió un rol de administrador, que cuenta con diferentes funcionalidades, que corresponden a las definidas en el levantamiento de requerimientos. A continuación, se enumerará cada una de las funcionalidades y se realizará una breve descripción de cada una de estas.

### Inicio

Una vez realizado el inicio de sesión por parte del usuario, se redirigirá a la página de inicio del administrador en la cual se encontrará un panel de control con información acerca de la producción lechera de la asociación en la que se muestran estadísticas y graficas de torta de los socios de los cuales se ha registrado la producción, además de información correspondiente al sistema como la cantidad de usuarios registrados.

A continuación, se mostrará el proceso llevado a cabo para la realización de estas funcionalidades.

### Figura 54

*Vista inicio administrador*

```

<?php |
require_once("../config/conexion.php");
if(isset($_SESSION["nombre"])&&(int)$_SESSION["rol"]==2){
?>
<!DOCTYPE html>
<html lang="en">

<head>
<?php require_once("../MainHead/mainHead.php");?>
<link rel="stylesheet" href="../public/assets/js/plugins/datatables/dataTables.bootstrap4.min.css">
<link rel="stylesheet" href="../public/libs/datatables/responsive.dataTables.min.css">
</head>

<body>
<div class="container-scroller">
<?php require_once("../MainHeader/mainHeader.php");?>

<div class="container-fluid page-body-wrapper">
<?php require_once("../MainSidebar/sidebarAdmin.php");?>
<div class="main-panel">
<div class="content-wrapper">

<div class="d-xl-flex justify-content-between align-items-start">
<h2 class="text-dark font-weight-bold mb-2"> Panel de información general </h2>
<div class="dropdown float-right">

```

*Nota.* Se muestra el archivo html que se mostrara al usuario, el cual contiene los contenedores correspondientes para mostrar la información, y en el cual se recuperan algunas variables de sesión.

## Figura 55

### Archivo js inicio administrador parte 1

```
function init(){
  document.getElementById("hoy").className = "btn btn-link text-dark py-0 border-right";
  document.getElementById("quincena").className = "btn btn-link text-light py-0 border-right";
  document.getElementById("mes").className = "btn btn-link text-light py-0 border-right";
  document.getElementById("dropdownMenuIconButton2").className = "btn bg-white dropdown-toggle p-3 d-flex align-items-center text-light";
  let boton = document.getElementById('dropdownMenuIconButton2');
  boton.innerText= dia+" "+obtenerMes(mes)+" "+ anio +" - "+dia+" "+obtenerMes(mes)+" "+ anio;
  mostrarDatos(hoy,hoy);
  datosEstadisticas(hoy,hoy);
}
init();
```

*Nota.* Se muestra la función “init” que se ejecuta inicialmente para mostrar la información correspondiente. Se puede observar que esta función llama a otras dos funciones, una llamada “mostrarDatos”, que se encarga de realizar la petición al servidor de una consulta por medio de dos fechas con el fin de obtener los datos correspondientes, y la otra llamada “datosEstadisticas”, que se encarga de mostrar la información en una gráfica correspondiente a la librería chart js.

## Figura 56

Archivo js inicio administrador parte 2

```
function mostrarDatos(fecha_inicial, fecha_final){
    Swal.fire({
        title: 'Cargando recursos',
        html: '<div class="spinner-border text-primary" role="status"><span class="sr-only">Cargando ...</span></div>',
        showConfirmButton: false,
        allowOutsideClick: false
    });

    let litros_totales=0;
    let litros_totales_vendidos=0;
    let ingresosTotales=0;

    $.post("../controller/produccion.php?op=produccionTotal", {
        FechaInicial: fecha_inicial,
        FechaFinal: fecha_final
    }, function (data) {
        data = JSON.parse(data);
        for (let step = 0;(step < data.aaData.length); step++) {
            let datose = data.aaData[step];
            litros_totales=litros_totales+parseInt(datose.Cantidad);
        }
    });
}
```

*Nota.* Se muestra el código de la función “mostrarDatos” la cual recibe dos parámetros, que corresponden a un rango de fechas, las cuales se utilizan para hacer la consulta en la base de datos para que retorne los registros realizados en el rango mencionado anteriormente.

## Figura 57

Controlador inicio administrador.

```
case "produccionTotal":
    $datos = $produccion->produccion_Total($_POST["FechaInicial"],$_POST["FechaFinal"]);
    $data= Array();
    foreach($datos as $row){
        $sub_array = array();
        $sub_array["Cantidad"] = $row["cantidad"];
        $sub_array["Fecha"] = $row["fecha"];
        $sub_array["Precio"] = $row["precio"];
        $data[] = $sub_array;
    }
    $results = array(
        "sEcho"=>1,
        "iTotalRecords"=>count($data),
        "iTotalDisplayRecords"=>count($data),
        "aaData"=>$data);
    echo json_encode($results);
break;
```

*Nota.* Se muestra el controlador que recibe los parámetros desde la vista y se encarga de llamar la función “producción\_total”, que se encuentra en el modelo. Cuando se realiza la petición desde el modelo, este se encarga de pasar los parámetros a la vista en forma de un array de datos.

**Figura 58**

*Modelo inicio administrador*

```
public function produccion_Total($FechaInicial,$FechaFinal){
    $conectar=parent::conexion();
    parent::set_names();
    $sql="SELECT cantidad, fecha, precio FROM produccion WHERE fecha BETWEEN ? AND ? ORDER
    BY fecha ASC";
    $sql=$conectar->prepare($sql);
    $sql->bindValue(1,$FechaInicial);
    $sql->bindValue(2,$FechaFinal);
    $sql->execute();
    return $resultado=$sql->fetchall(PDO::FETCH_ASSOC);
}
```

*Nota.* Se muestra la función “producción\_Total”, la cual retorna una consulta SQL, retorna las columnas cantidad, fecha y precio, que se encuentren en el rango de fecha especificado desde la vista.

**Registrar producción**

Una de las funcionalidades principales del administrador es registrar la producción de cada uno de los socios, ya que a partir de estos registros se obtienen los pagos que se deben realizar a cada uno de estos. A continuación, se muestra el proceso realizado para desarrollar esta funcionalidad

Figura 59

Vista registro de producción

```

<form id="registrarProduccion" class="forms-sample">
  <div class="form-group">
    <label class="text-dark" for="exampleInputUsername1">Producción (Litros)<span
      class="text-danger">*</span></label>
    <input type="number" class="form-control" id="litros" name="validarlitro">
  </div>
  <div class="form-group">
    <label class="text-dark" for="exampleInputEmail1">Fecha<span
      class="text-danger">*</span></label>
    <input type="date" name="validarfecha" class="form-control" id="fecha"
      value="<?php echo date("Y-m-d");?>">
  </div>
  <div class="form-group">
    <label class="text-dark" for="exampleInputPassword1">Volumen</label>
    <input type="number" class="form-control" id="volumen">
  </div>
  <div class="form-group">
    <label class="text-dark" for="exampleInputConfirmPassword1">Precio por litro<span
      class="text-danger">*</span></label>
    <input type="number" class="form-control" id="preciolitro">
  </div>
  <div class="form-group">
    <label class="text-dark" for="exampleTextarea1">Observaciones</label>
    <textarea class="form-control" id="observaciones" rows="4"></textarea>
  </div>
  <div class="form-group text-center">
    <button type="button" id="registrarleche"
      class="btn btn-success btn-fw">Registrar</button>
  </div>
</form>

```

*Nota.* Se muestra el formulario utilizado para el registro de la información correspondiente a la producción lechera que cada socio aporta a la asociación. Se solicitan datos como los litros de leche, la fecha (que está definida por defecto al día actual), el volumen, el precio, y un campo de observaciones.

## Figura 60

### Archivo js registro de producción

```
$(document).on("click", "#registrarleche", function () {  
  
  Swal.fire({  
    title: 'Espere mientras procesamos su solicitud',  
    html: '<div class="spinner-border text-primary" role="status"><span class="sr-only">Loading...</span></div>',  
    showConfirmButton: false,  
    allowOutsideClick: false  
  });  
  
  var $this = $(this);  
  
  if ($('#litros').val() == "" || $('#fecha').val() == "" || $('#preciolitro').val() == "") {  
    Swal.close();  
    Swal.fire(  
      'AsoVapp',  
      'Por favor complete los campos requeridos',  
      'error'  
    );  
  } else {  
    let litros = $('#litros').val();  
    let fecha = $('#fecha').val();  
    let precio = $('#preciolitro').val();  
    let volumen = $('#volumen').val();  
    let observaciones = $('#observaciones').val();  
    $.post("../controller/produccion.php?op=validarfecha", {  
      IdUsuario: IdUsuario,  
      Fecha: fecha,  
    }, function (data) {  
  
      if (data == "fecha") {  
        registrar();  
      }  
    });  
  }  
});
```

*Nota.* Se muestra la función que se utiliza para registrar la producción lechera, inicialmente se validan los campos obligatorios, luego se valida que no haya registros del mismo socio en la misma fecha, y finalmente si se cumplen las validaciones, se envían los datos al controlador, en caso que no se cumpla la validación se muestra un mensaje al usuario indicando el error.

**Figura 61***Controlador registro de producción*

```

case "registrar":
    $datos=$produccion->insert_registro($_POST["IdUsuario"],$_POST["Litros"],$_POST
["Fecha"],$_POST["Precio"],$_POST["Volumen"],$_POST["Observaciones"]);
    if($datos=="ok"){
        echo "ok";
    }else{
        echo $datos;
    }
}
break;

```

*Nota.* Se muestra la función utilizada para enviar la información al modelo, se pasan una serie de parámetros provenientes de la vista, y dependiendo de la respuesta del servidor se devuelve un mensaje a la vista.

**Figura 62***Modelo registrar producción*

```

public function insert_registro($IdUsuario,$Litros,$Fecha,$Precio,$Volumen,$Observaciones){
    $conectar=parent::conexion();
    parent::set_names();
    $sql1="INSERT INTO produccion values (NULL,?,?,?,?,?,?)";
    $sql1=$conectar->prepare($sql1);
    $sql1->bindValue(1,$Litros);
    $sql1->bindValue(2,$Fecha);
    $sql1->bindValue(3,$Volumen);
    $sql1->bindValue(4,$Observaciones);
    $sql1->bindValue(5,$Precio);
    $sql1->bindValue(6,$IdUsuario);
    $sql1->bindValue(7,1);
    if ($sql1->execute() {
        if ($sql1->rowCount() > 0) {
            return "ok";
        } else {
            return "No se han realizado cambios en la base de datos";
        }
    } else {
        return "Error al ejecutar la consulta SQL";
    }
}
}

```

*Nota.* Función utilizada para realizar el registro en la base de datos, primero se establece la conexión a la base de datos, luego se estructura la sentencia SQL, se envían los parámetros y finalmente se retorna si el registro fue realizado correctamente o no.

### **Mostrar producción**

Es una de las funcionalidades principales del sistema, ya que muestra la producción registrada por cada uno de los socios, para realizar esta funcionalidad se hizo uso de una librería llamada datatable, que permite mostrar los registros de manera ordenada, permite realizar búsquedas y exportar la información en formatos pdf y xml.

A continuación, se muestra el proceso que se llevó a cabo para realizar esta funcionalidad.

**Figura 63***Vista mostrar producción*

```

<div class="col-sm-12">
  <button type="button" onclick="mostrarProduccion()" id="graficar"
    class="btn btn-inverse-primary btn-fw">Mostrar registros</button>
</div>
</div>
<table id="listaProduccion" class="table table-striped display nowrap">
  <tbody>
    <thead>
      <tr>
        <th>Fecha</th>
        <th>Litros</th>
        <th>Volumen</th>
        <th>Precio</th>
        <th>Observaciones</th>
        <th></th>
      </tr>
    </thead>
  </tbody>
</table><br>
<div class="row">
  <div class="col-sm-6">
    <div class="text-center">
      <h4>Litros totales</h4>
      <h1 class="text-dark font-weight-bold" id="litrost"></h1>
    </div>
  </div>
  <div class="col-sm-6">
    <div class="text-center">
      <h4>Pago por la producción</h4>
      <h1 class="text-dark font-weight-bold" id="dinero"></h1>
    </div>
  </div>
</div>

```

*Nota.* Se muestra el html con la estructura que se utilizó para mostrar la información, se definió una tabla y se le asignó un identificador, esto con el fin de ejecutar la funcionalidad correspondiente desde el script, luego se definieron las columnas que debe tener la tabla en este caso de definieron seis columnas, la sexta columna se definió para mostrar un botón que permite realizar la edición de ese registro.

**Figura 64***Mostrar producción JavaScript*

```
let table = $('#listaProduccion').DataTable({
  destroy: true,
  dom: 'Bfrtip',
  buttons: { ...
  },
  "processing": true,
  "ajax": {
    url: "../../controller/produccion.php?op=listarProduccion",
    type: "post",
    data: { ...
  },
  error: function (e) {
    console.log(e.responseText);
  },
  },
  "responsive": true,
  "bInfo": true,
  "iDisplayLength": 5,
  "order": [[0, "desc"]],
  "language": {
    "sProcessing": "Procesando...",
    "sLengthMenu": "Mostrar _MENU_ registros",
    "sZeroRecords": "No se encontraron resultados",
    "sEmptyTable": "Ningún dato disponible en esta tabla",
    "sInfo": "Mostrando registros del _START_ al _END_ de un total de _TOTAL_ registros",
    "sInfoEmpty": "Mostrando registros del 0 al 0 de un total de 0 registros",
    "sInfoFiltered": "(filtrado de un total de _MAX_ registros)",
    "sInfoPostFix": "",
    "sSearch": "Buscar:",
  }
});
```

*Nota.* Se muestra la declaración de la tabla en la cual se configuran una serie de parámetros, primero se identifica la tabla que se encuentra en la vista, luego se define el tipo de tabla, los botones y funcionalidades que se requieren, finalmente se envía la solicitud al controlador para que se termine de ejecutar lo que hace falta.

Figura 65

*Mostrar producción controlador.*

```

case "listarProduccion":
    $datos=$produccion->list_produccion($_POST["IdUsuario"],$_POST["FechaInicial"],$_POST["FechaFinal"]);
    $data= Array();
    foreach($datos as $row){
        $sub_array = array();
        $sub_array[] = $row["fecha"];
        $sub_array[] = $row["cantidad"];
        $sub_array[] = $row["volumen"];
        $sub_array[] = $row["precio"];
        $sub_array[] = $row["observaciones"];
        $sub_array[] = '<div class="dropdown float-right">
        <button class="btn btn-inverse-dark dropdown-toggle" type="button" id="dropdownMenuIconButton2"
        data-toggle="dropdown" aria-haspopup="true" aria-expanded="false">
        Opciones</button>
        <div class="dropdown-menu" aria-labelledby="dropdownMenuIconButton2">
            <a class="dropdown-item text-dark" onClick="editarRegistro('.$row["id_produccion"].');' href="#"><i
            class="mdi mdi-border-color mr-2"></i>Editar</a>
            <a class="dropdown-item text-dark" onClick="eliminarProyecto()"href="#"><i class="mdi mdi-delete
            mr-2"></i>Eliminar</a>
        </div>
        </div>';
        $sub_array[] = $row["usuario_id"];
        $data[] = $sub_array;
    }

    $results = array(
        "sEcho"=>1,
        "iTotalRecords"=>count($data),
        "iTotalDisplayRecords"=>count($data),
        "aaData"=>$data);
    echo json_encode($results);
break;

```

*Nota.* Se muestra la función realizada en el modelo, la cual consiste primero en realizar una petición al servidor en la cual se deben obtener los registros de determinado rango de fechas, luego se realiza el envío de los parámetros correspondientes al datatable por medio del bucle for, finalmente la librería se encarga de organizar los registros en la tabla declarada anteriormente.

**Figura 66***Mostrar producción modelo*

```

public function list_produccion($IdUsuario,$FechaInicial,$FechaFinal){
    $conectar=parent::conexion();
    parent::set_names();
    $sql="SELECT * FROM produccion WHERE usuario_id=? AND fecha BETWEEN ? AND ? ORDER BY fecha ASC;";
    $sql=$conectar->prepare($sql);
    $sql->bindvalue(1, $IdUsuario);
    $sql->bindvalue(2, $FechaInicial);
    $sql->bindvalue(3, $FechaFinal);
    $sql->execute();
    return $resultado=$sql->fetchall(PDO::FETCH_ASSOC);
}

```

*Nota.* se muestra la consulta SQL, que se utilizó para obtener los registros de la base de datos, se envían como parámetros de búsqueda una fecha inicial, una fecha final y un identificador de usuario.

## Registrar Salidas

La funcionalidad de registrar salidas corresponde a un registro que se realiza cuando la leche es recolectada por la empresa que realiza la compra, en este registro se debe tener en cuenta los litros de leche, la fecha y el precio de la compra.

**Figura 67***Registrar salidas Vista*

```

<div class="modal-header">
<h4 class="text-dark">Registrar venta de leche</h4>
<button type="button" class="close" data-dismiss="modal" aria-label="Close">
<span aria-hidden="true">&times;</span>
</button>
</div>
<div class="card-body">
<form class="forms-sample">
<div class="form-group">
<label class="text-dark">Fecha de venta <span class="text-danger">*</span></label>
<input type="date" name="validarfecha" class="form-control" id="fechahoy" value="<?php echo date("Y-m-d");?>">
</div>
<div class="form-group">
<label class="text-dark">Cantidad (litros) <span class="text-danger">*</span></label>
<input type="number" class="form-control" id="cantidadlitros">
</div>
<div class="form-group">
<label class="text-dark">Precio por litro <span class="text-danger">*</span></label>
<input type="number" class="form-control" id="preciolitro">
</div>
</form>
<div class="modal-footer">
<button type="button" class="btn btn-secondary" data-dismiss="modal">Cerrar</button>
<button type="button" id="guardarventa" class="btn btn-primary">Guardar</button>
</div>
</div>

```

*Nota.* Se muestra el formulario utilizado para recibir la información proveniente por parte del usuario, se definieron los campos correspondientes para el registro.

**Figura 68***Registrar salidas JavaScript*

```

$(document).on("click", "#guardarventa", function () {
    if ($('#fecha').val() == "" || $('#cantidadlitros').val() == "" || $('#preciolitro').val() == "") {
        Swal.fire(
            'AsoVapp',
            'Por favor complete los campos requeridos',
            'error'
        );
    } else {
        let litros = $('#cantidadlitros').val();
        let fecha = $('#fechahoy').val();
        let precio = $('#preciolitro').val();
        let total = precio * litros;
        Swal.fire({
        });
        $.post("../controller/empresa.php?op=validarfecha", {
            IdEmpresa: IdEmpresa,
            Fecha: fecha,
        }, function (data) {
            if (data == "fecha") {
                $.post("../controller/empresa.php?op=registrarventa", {
                    IdEmpresa: IdEmpresa,
                    Litros: litros,
                    Fecha: fecha,
                    Precio: precio,
                    Total: total,
                }, function (data) {
                    console.log(data)
                    $('#guardarventa').prop("disabled", true);
                    // add spinner to button
                    $('#guardarventa').html(
                        '<span class="spinner-border spinner-border-sm" role="status" aria-hidden="true"></span> Cargando..
                    );
                });
            }
        });
    }
});

```

*Nota.* Se muestra el código en donde se realizan las validaciones de los campos y se envían los parámetros correspondientes al controlador.

**Figura 69***Registrar salidas controlador*

```

case "registrarventa":
    $datos = $empresa->insert_registro($_POST["IdEmpresa"],$_POST["Litros"],$_POST["Fecha"],$_POST["Precio"],$_POST["Total"]);
    echo $datos;
break;

```

*Nota.* Se muestra la comunicación que realiza el controlador con el modelo en la cual se envían los parámetros correspondientes para realizar el registro.

**Figura 70**

*Registrar salidas controlador*

```
public function insert_registro($IdEmpresa,$Litros,$Fecha,$Precio,$Total){
    $conectar=parent::conexion();
    parent::set_names();
    $sql1="INSERT INTO registro_entrega values (NULL,?,?,?,?)";
    $sql1=$conectar->prepare($sql1);
    $sql1->bindValue(1,$Fecha);
    $sql1->bindValue(2,$Litros);
    $sql1->bindValue(3,$Precio);
    $sql1->bindValue(4,$Total);
    $sql1->bindValue(5,$IdEmpresa);
    if ($sql1->execute() {
        if ($sql1->rowCount() > 0) {
            return "ok";
        } else {
            return "No se han realizado cambios en la base de datos";
        }
    } else {
        return "Error al ejecutar la consulta SQL";
    }
}
```

*Nota.* se muestra la función encargada de realizar el registro de la venta realizada, se puede apreciar el SQL utilizado para insertar el registro, finalmente se puede ver que se retorna una alerta dependiendo si el registro fue correcto o no.

## Anexos

Figura 71

Árbol de problemas



*Nota.* Se muestran las causas y efectos del problema central en la asociación Asoagroproleche.

**Resultados metodología de investigación**

Yeimy Constanza Ángel Sánchez y Jairo Orlando Pascagaza Barriga

Programa de ingeniería de sistemas, Universidad de Cundinamarca

Trabajo de grado

Director

Cristian Eduardo Cano López

Junio 2, 2023

## Indice

<i>14. Análisis de las entrevistas realizadas</i> .....	88
7.1. Entrevista a presidente.....	88
7.2. Entrevista Administradora del tanque.....	92
7.3. Entrevistas a Socios.....	97

### Índice de figuras

<b>Figura 72</b> Respuesta a primera pregunta.....	88
<b>Figura 73</b> Respuesta a segunda pregunta .....	89
<b>Figura 74</b> Respuesta a tercera pregunta .....	89
<b>Figura 75</b> Respuesta a cuarta Pregunta .....	90
<b>Figura 76</b> Respuesta quinta pregunta .....	90
<b>Figura 77</b> Respuesta a sexta pregunta .....	91
<b>Figura 78</b> Respuesta a séptima pregunta.....	91
<b>Figura 79</b> Respuesta a la primera pregunta.....	93
<b>Figura 80</b> Respuesta a la segunda pregunta .....	94
<b>Figura 81</b> Respuesta a la tercera pregunta .....	94
<b>Figura 82</b> Respuesta a cuarta pregunta .....	95
<b>Figura 83</b> Evidencia de respuesta.....	96
<b>Figura 84</b> Socios de Asoagroproleche .....	97
<b>Figura 85</b> Evidencia de respuesta.....	98
<b>Figura 86</b> Evidencia de respuesta.....	99
<b>Figura 87</b> Evidencia de respuesta.....	100
<b>Figura 88</b> Evidencia de respuesta.....	101
<b>Figura 89</b> Evidencia de respuesta.....	102
<b>Figura 90</b> Evidencia de respuesta.....	103

## Análisis de las entrevistas realizadas

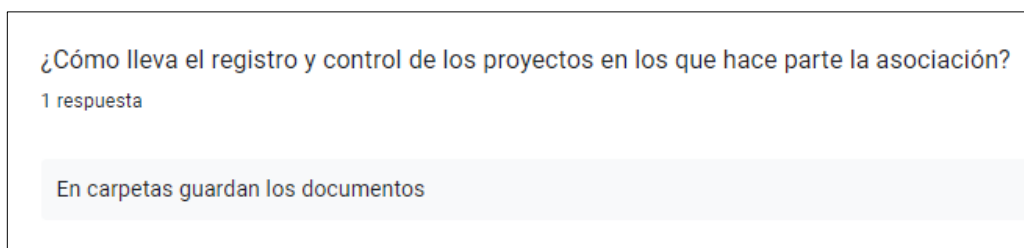
Recopilados los resultados de las entrevistas estadísticas, se realizó un plan de análisis de datos, en el cual, se clasifica, controla y analiza la información, para de esta manera verificar que el aplicativo sea aceptable por los integrantes de la asociación, luego se procede a tabular los datos obtenidos en las diferentes entrevistas para obtener los porcentajes por cada una de las preguntas formuladas. Se validó la calidad de los datos para minimizar el sesgo y optimizar el análisis de estos, finalmente se llevó a cabo el proceso de sacar conclusiones para continuar con la formulación de requerimientos y desarrollo del proyecto. A continuación, se presentan las conclusiones y resultados que se obtuvieron:

### 7.1. Entrevista a presidente

Esta entrevista se realizó al presidente de la asociación Asoagroproleche con el fin de recolectar la información necesaria para la formulación de los requerimientos del módulo perteneciente a este departamento de dicha entidad.

#### Figura 72

*Respuesta a primera pregunta*



¿Cómo lleva el registro y control de los proyectos en los que hace parte la asociación?

1 respuesta

En carpetas guardan los documentos

*Nota.* Formulario realizado por el equipo de trabajo en el cual se evidencia que se llevan los procesos de manera manual. .

**Figura 73**

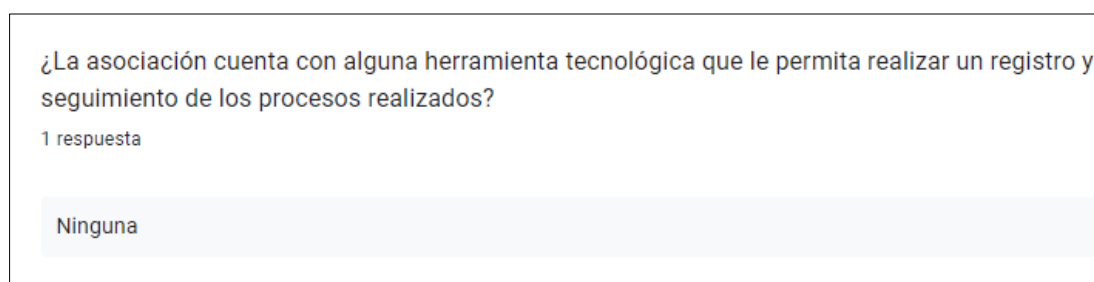
*Respuesta a segunda pregunta*



*Nota.* Gráfico realizado por el equipo de trabajo en el cual se muestra que está dispuesto en un 100% a utilizar una aplicación en la asociación. .

**Figura 74**

*Respuesta a tercera pregunta*



*Nota.* Formulario realizado por el equipo de trabajo en el cual se evidencia que la asociación no cuenta con ninguna Herramienta tecnológica para el control de sus actividades.

**Figura 75**

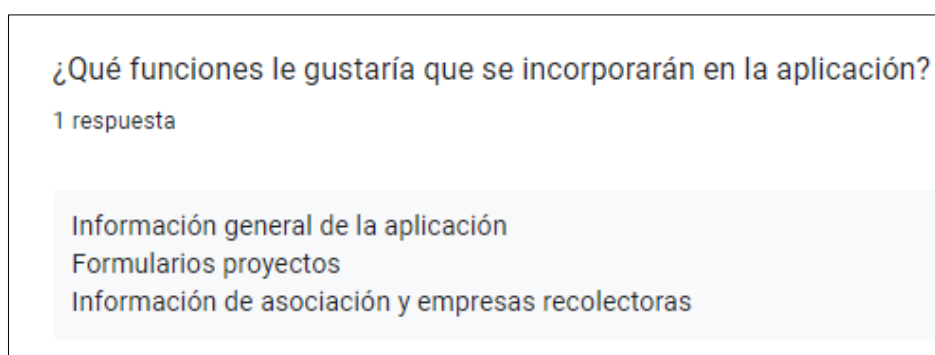
*Respuesta a cuarta Pregunta*



*Nota.* Gráfico realizado por el equipo de trabajo que evidencia que si le aportara la tecnología a la entidad.

**Figura 76**

*Respuesta quinta pregunta*



*Nota.* Formulario realizado por el equipo de trabajo en el cual se evidencian cuáles deberían ser las funciones del sistema.

**Figura 77**

*Respuesta a sexta pregunta*

¿En la asociación se cuenta con algún tipo de tecnología para realizar alguna de las actividades, Entiéndase por tecnología a maquinaria, equipos o Aplicaciones?

1 respuesta

Si.  
Tanque recolector y un computador

*Nota.* Formulario realizado por el equipo de trabajo en el cual se muestra las tecnologías con las que cuenta la entidad.

**Figura 78**

*Respuesta a séptima pregunta*

¿ Como presidente de la asociación Asoagroproleche cree usted que la implementación de tecnologías nuevas e innovadoras posicionarían a la asociación como un ejemplo a seguir en el municipio?

1 respuesta

Si

*Nota.* Formulario realizado por el equipo de trabajo en el cual se muestra que el presidente de esta entidad si estuviese dispuesto a implementar tecnologías para el

mejoramiento y avance de esta.

La recopilación de datos realizada por medio de las siete preguntas con las que cuenta esta entrevista permitió adquirir los datos requerido para el diseño y formulación de los requerimientos paraposteriormente elaborar el módulo inicial del sistema AsoVapp denominado módulo de presidente.

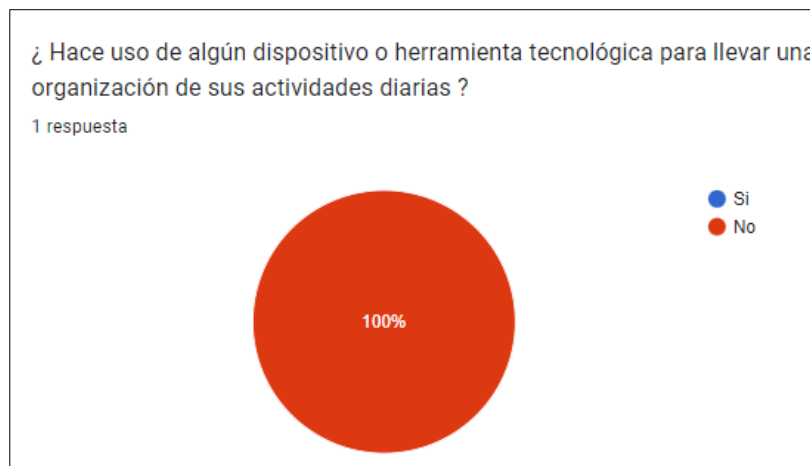
La segunda entrevista fue realizada a la encargada de administrar el tanque recolector de leche en el acopio de la asociación, ya que esta es la sección principal de la empresa en la que se realiza el proceso de recolectar el producto y venderlo a una empresa productora y comercializadora de derivados lácteos, se diseñaron cinco preguntas las cuales permiten obtener los datos requeridos por los desarrolladores para realizar el módulo denominado Administración del tanque.

## **7.2. Entrevista Administradora del tanque.**

La entrevista a la administradora de tanque es para identificar las acciones a seguir para el desarrollo del sistema. Donde se obtuvieron las siguientes respuestas.

**Figura 79**

*Respuesta a la primera pregunta*



*Nota.* Se muestra el grafico de que al momento no hacen uso de una herramienta tecnológica.

**Figura 80**

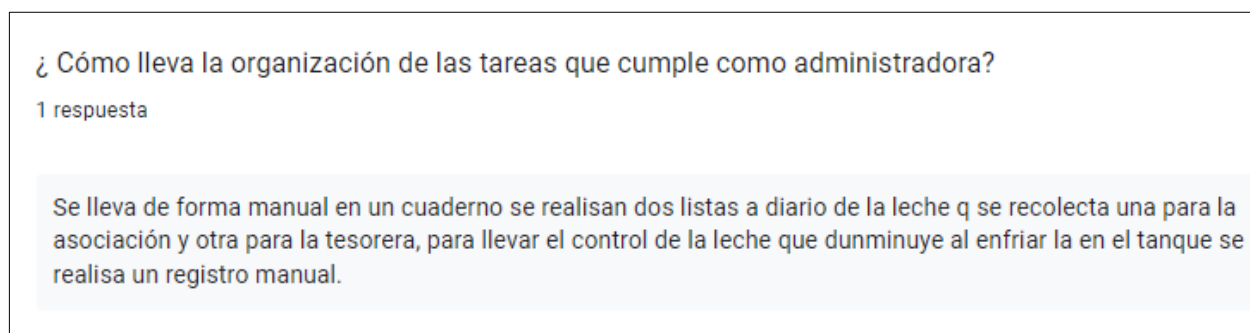
*Respuesta a la segunda pregunta*



*Nota.* Gráfico diseñado por el equipo de trabajo en el cual la respuesta a la segunda pregunta se evidencia que en un 100% no se conoce de aplicaciones o software para este sector.

**Figura 81**

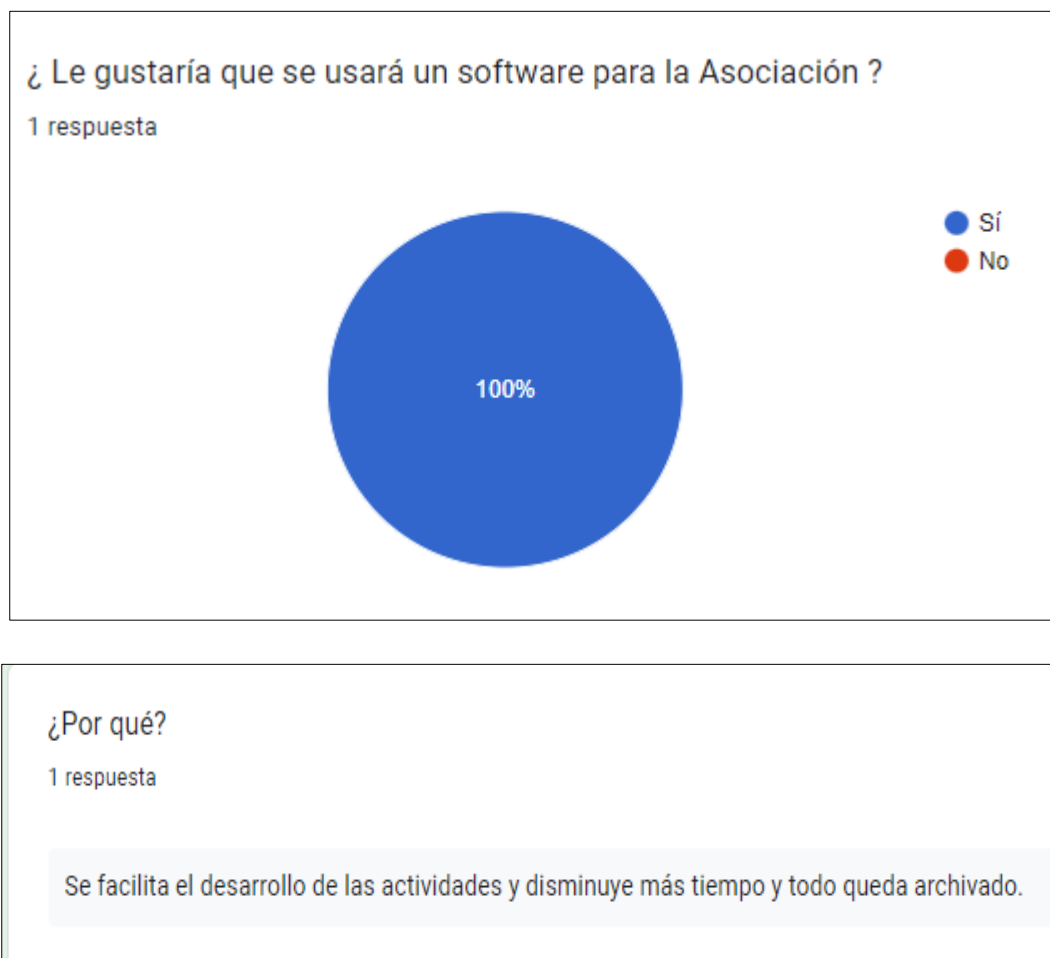
*Respuesta a la tercera pregunta*



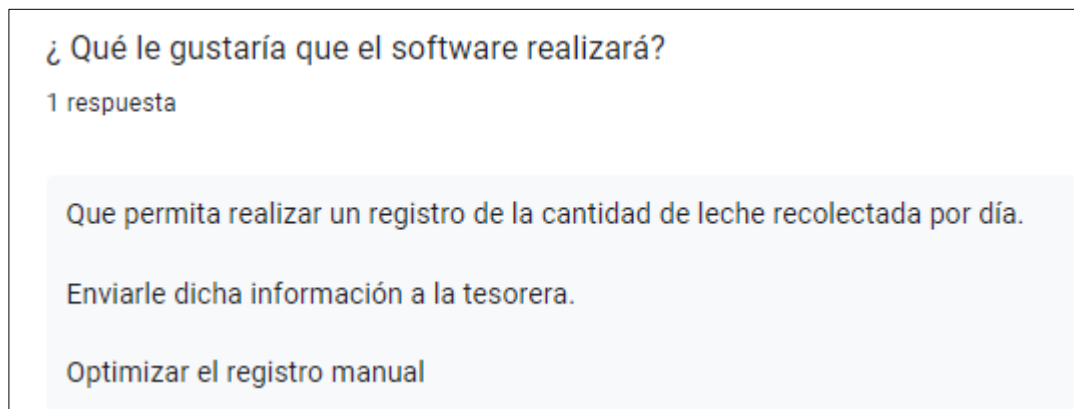
*Nota.* Formulario diseñado por el equipo de trabajo en el cual se visualiza la evidencia que los procesos que realiza la administradora son de forma manual.

**Figura 82**

*Respuesta a cuarta pregunta*



*Nota.* Gráfico diseñado por el equipo de trabajo en el cual se evidencia que en un 100% si implementa un sistema.

**Figura 83***Evidencia de respuesta*

¿ Qué le gustaría que el software realizará?

1 respuesta

Que permita realizar un registro de la cantidad de leche recolectada por día.  
Enviarle dicha información a la tesorera.  
Optimizar el registro manual

*Nota.* Diseñado por el equipo de trabajo en el cual se muestran las perspectivas de lo que llegara a hacer el software en la asociación.

Se evidencia que la recopilación de información fue oportuna respondiendo que los procesos son llevados de manera manual y no hay la presencia de un sistema tecnológico en la asociación “Asoagroproleche”, respaldando así que si se usará un software para facilitar cada una de las tareas y actividades que se cumplen como administradora de tanque permitiendo así conocer cada uno de los requerimientos para llevar el desarrollo del módulo de administradora del sistema AsoVapp.

Por último, la tercera entrevista se realizó a los productores de leche asociados a la entidad es decir los ganaderos los cuales tienen un proceso interno de administración y manejo de cada una de sus fincas y sus animales, Para esta entrevista se realizaron siete preguntas en las cuales se recolecto la información esencial para diseñar y desarrollar el módulo del sistema denominado Ganaderos.

### 7.3. Entrevistas a Socios

Al realizar las entrevistas fue fundamental para conocer los requerimientos para la correcta toma de decisiones en el desarrollo de AsoVapp, como es el manejo de los datos, también se pudo evidenciar las prácticas que estos realizan en cada uno de los procesos que son llevados a cabo.

Esta entrevista fue realizada a los usuarios de la asociación “Asoagroproleche” con una muestra de 17 ganaderos los cuales compartieron las siguientes respuestas.

#### Figura 84

*Socios de Asoagroproleche*

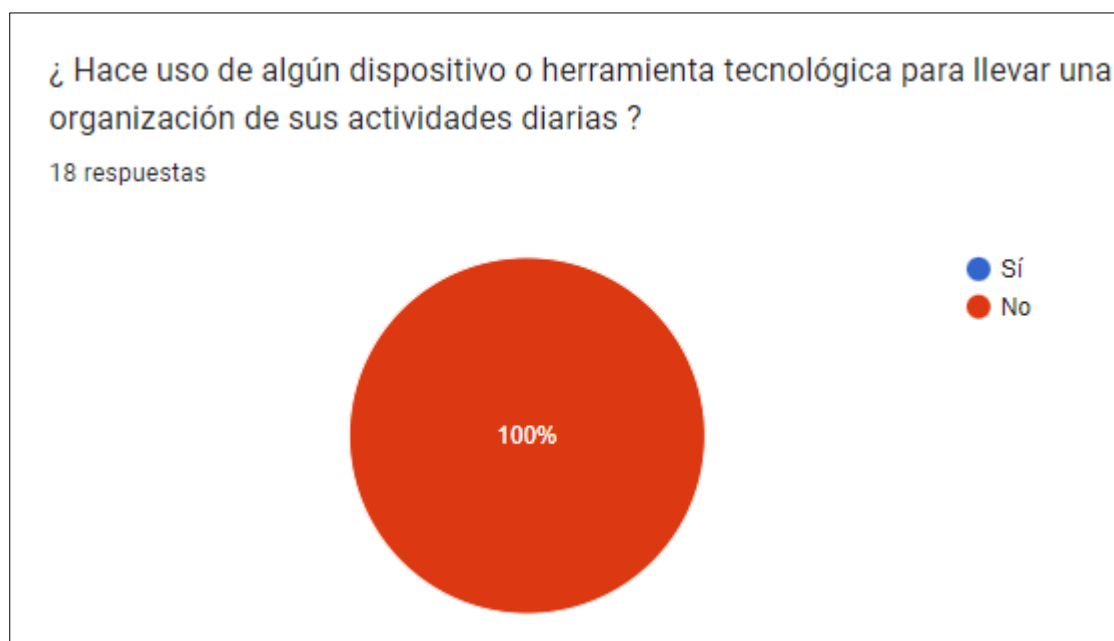


Gloria Marlen Guzmán Garnica
Humberto Martinez
Alirio Rodriguez
Eduardo Guzmán
Isidro Romero
Luz Dary Rodriguez
Elsa Contreras
Trinidad Pinilla
Alvaro fresneda

*Nota.* Diseño realizado por el equipo de trabajo en el cual se muestran los nombres de los ganaderos asociados.

**Figura 85**

*Evidencia de respuesta*



*Nota.* Diseño realizado por el equipo de trabajo en el cual se muestra que ninguno de los asociados usa herramientas tecnológicas.

Se evidencia que el 100% de los entrevistados no utilizan herramientas o dispositivos tecnológicos para realizar algún proceso que llevan a cabo en sus labores. }

**Figura 86***Evidencia de respuesta*

*Nota.* Diseño realizado por el equipo de trabajo en el cual se muestra el gráfico del total de respuesta.

En esta se evidencia que en un 100% de los ganaderos no conocen de la aplicación de tecnologías en el sector lechero y ganadero lo que hace que en esta zona de Carmen de Carupa requieran del trabajo profesional favoreciendo a tener una mejora continua, con la implementación de tecnologías se contribuiría a un mejor desarrollo y progreso de los ganaderos lecheros.

**Figura 87***Evidencia de respuesta*

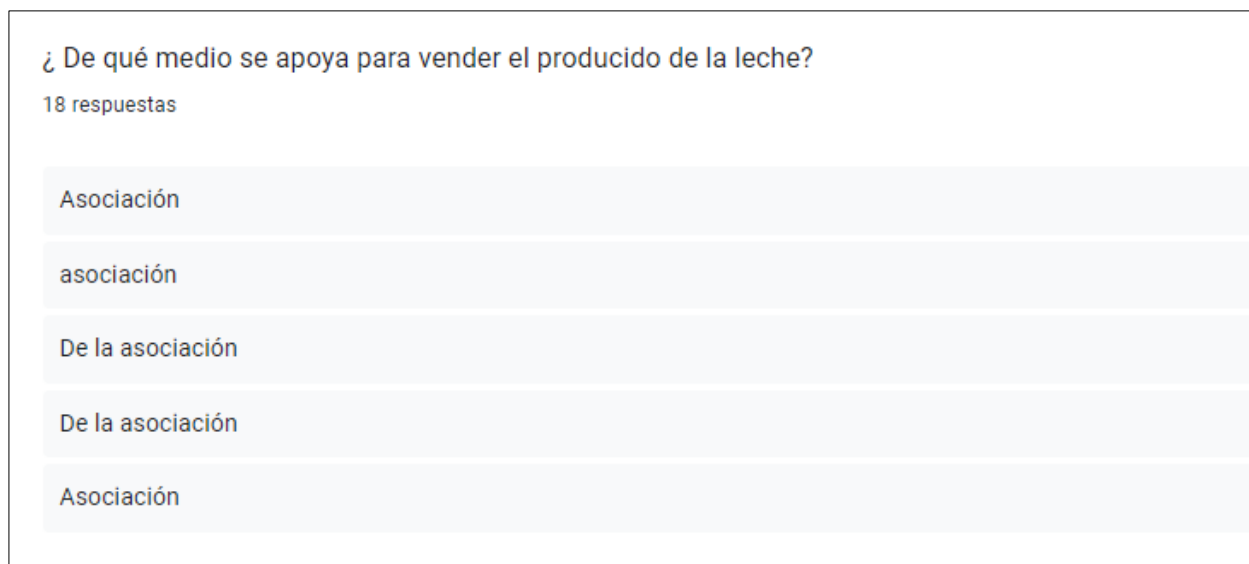
¿ Cómo lleva el registro y control de su ganado como las crías ,vacunas ?

18 respuestas

- En un cuaderno
- No llevo registro
- No llevo
- No llevo ningún registro x que no cuento con las herramientas necesarias
- Algunos en cuaderno
- no tengo ese registro
- En una hoja de papel
- no llevo control
- En un cuaderno

*Nota.* Diseño realizado por el equipo de trabajo en el cual se muestra que la mayoría de los ganaderos llevan los registros de manera manual.

En esta se puede identificar que muy pocos realizan este proceso de manera manual y algunos de los usuarios no llevan registro y control de cada uno de sus animales lo que hace que no estén contribuyendo al bienestar de su ganado a lo que se pueden ver afectados con la producción de su producto.

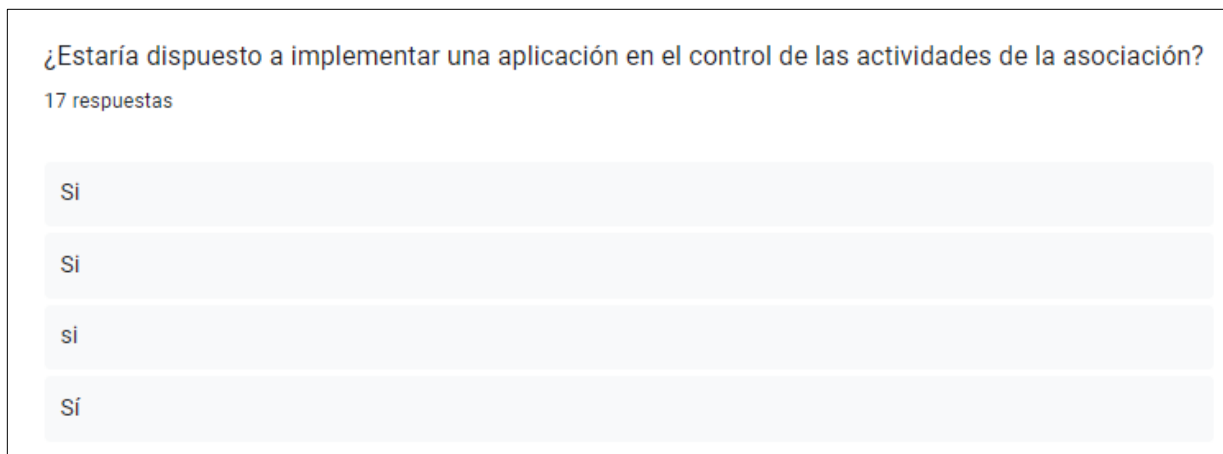
**Figura 88***Evidencia de respuesta*

*Nota.* Diseño realizado por el equipo de trabajo en el cual se muestra que el medio de apoyo para la venta de leche de los entrevistados es la asociación.

En esta se puede identificar que muy pocos realizan este proceso de manera manual y algunos de los usuarios no llevan registro y control de cada uno de sus animales lo que hace que no estén contribuyendo al bienestar de su ganado a lo que se pueden ver afectados con la producción de su producto.

**Figura 89**

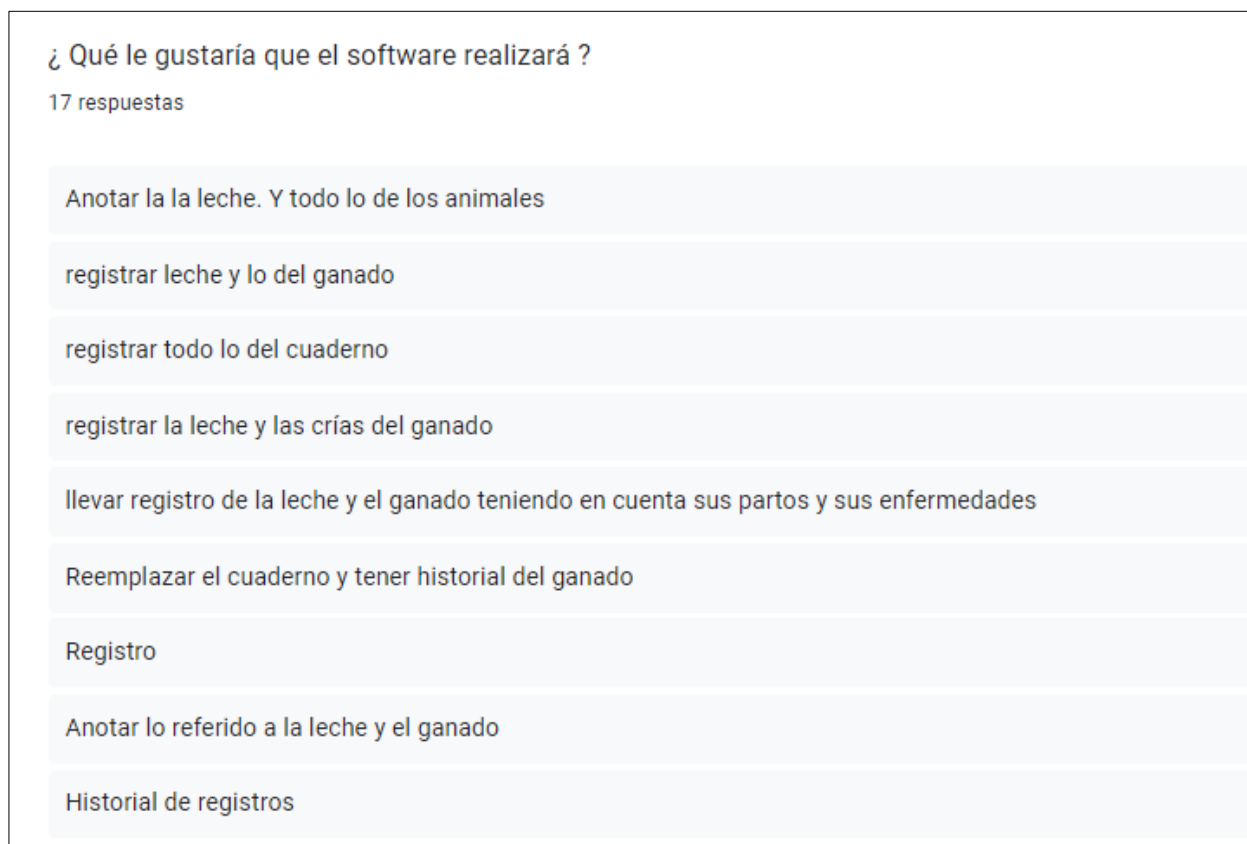
## Evidencia de respuesta



*Nota.* Diseño realizado por el equipo de trabajo en el cual se muestra que los usuarios apoyan para que el proyecto se lleve a cabo.

En esta pregunta se evidencia que los ganaderos están dispuestos a usar un sistema tecnológico en la asociación para un mejor orden y control facilitando así cada uno de los procesos.

En la última pregunta los ganaderos comparten las expectativas que tienen al momento de implementar el sistema lo que comentan que les gustaría llevar los registros de sus ganados, leche de una manera organizada y se pueda ver el historial en el momento que se requiera. De igual manera poder registrar cada uno de sus animales en el sistema

**Figura 90***Evidencia de respuesta*

*Nota.* Diseño realizado por el equipo de trabajo en el cual se muestran las perspectivas de implementar un software.

## **Mockups**

Yeimy Constanza Ángel Sánchez y Jairo Orlando Pascagaza Barriga

Programa de ingeniería de sistemas, Universidad de Cundinamarca

Trabajo de grado

Director

Cristian Eduardo Cano López

Junio 2, 2023

## Índice de Figuras

<b>Figura 1</b>	Registros de producción lechera.....	16
<b>Figura 2</b>	Diagrama Burn Down.....	44
<b>Figura 3</b>	Metodología.....	45
<b>Figura 4</b>	Evidencia de respuesta.....	76
<b>Figura 5</b>	Diagrama de contexto.....	94
<b>Figura 6</b>	Caso de uso Inicio sesión .....	95
<b>Figura 7</b>	Diagrama de casos de Usos de presidente .....	96
<b>Figura 8</b>	Diagrama caso de uso socio.....	97

## Figura 91

### *Mockups de Sign Up y login*



*Nota.* Diseño realizado por el equipo de trabajo que muestra el diseño de los campos para registrarse como usuario de “Asovapp”.

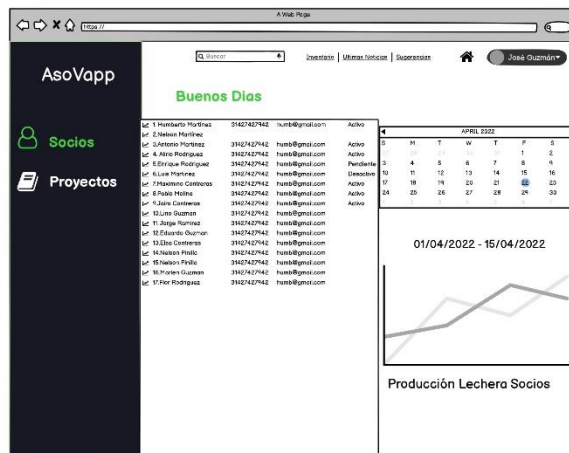
En estos mockups se realiza el diseño de los campos y requisitos para poder el usuario final registrarse en el sistema con su nombres, apellidos, telefono, correo electrónico y la respectiva contraseña donde sera guardado en la base de datos la cual con el login valida cada uno de los datos guardados con su usuario y contraseña podra iniciar sesión y disponer de cada uno de los servicios que este ofrece.

### **Módulo de presidente**

En este diseño se muestra el primer mockup de dicho módulo en el cual se encuentra el menú de opciones que tiene el presidente y su página de inicio con información relevante de su asociación como lo la producción lechera de los socios, por último, tiene la opción de cerrar sesión.

**Figura 92**

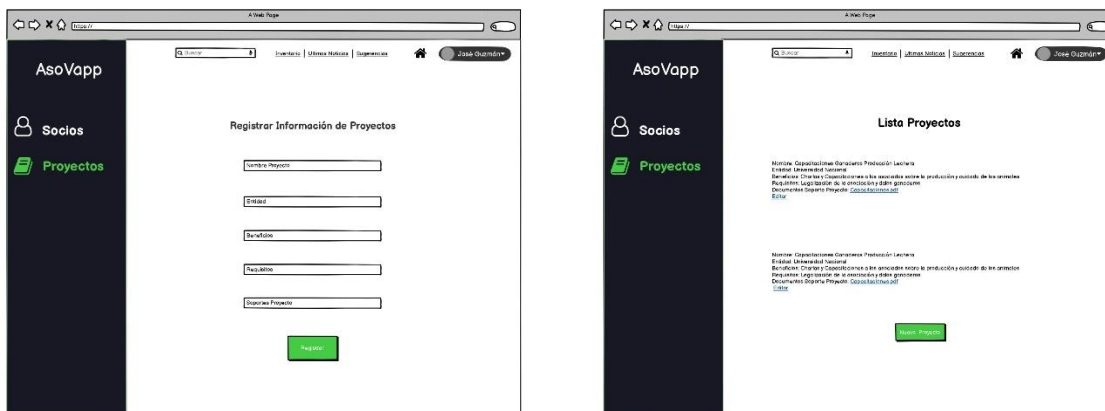
*Mockup inicial del módulo presidente*



*Nota.* Diseño realizado por el equipo de trabajo en el cual se muestra el diseño de la ventana inicial del módulo de presidente perteneciente al sistema “Asovapp”.

**Figura 93**

*Mockups Opción registro e historial de proyectos*



*Nota.* Diseño realizado por el equipo de trabajo en el cual se muestra el diseño de la opción registro e historial de proyectos en el módulo de presidencia perteneciente al sistema “Asovapp”.

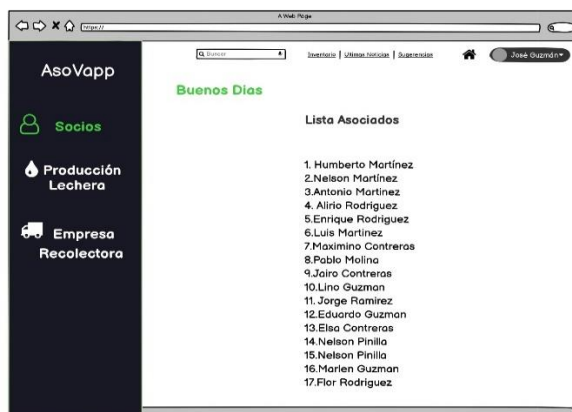
La opción de este módulo permite al usuario registrar la información de cada uno de los proyectos en los que la asociación participa por medio de un formulario y luego ver el historial de los registros y así mismo poder editarlos o borrarlos.

### Mockups Módulo de Administradora de tanque

Al iniciar Sesión el sistema lleva a la administradora del tanque a una ventana correspondiente a la primera opción de este módulo, en la cual encuentra un menú de opciones que para este rol son dos, adicionalmente encuentra la lista de asociados en dicha entidad con el fin de poder realizar el registro de la leche recibida de una forma individual, y un formulario en el cual se selecciona la fecha en el que se va a realizar el registro.

### Figura 94

#### *Submódulo Opción Leche diaria*



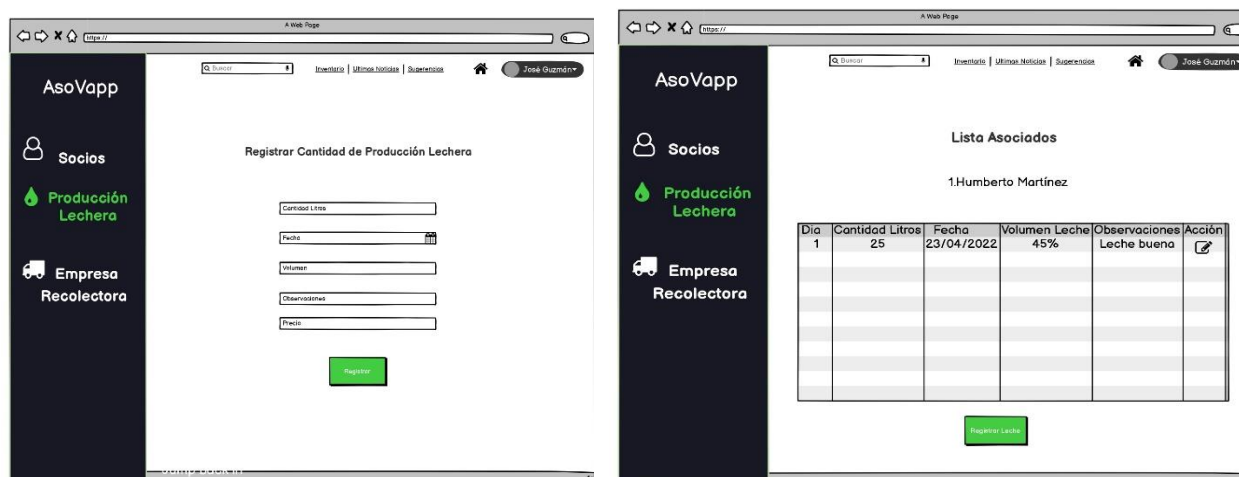
*Nota.* Diseño realizado por el equipo de trabajo en el cual se muestra el diseño de la ventana de la opción leche diaria en el módulo administradora el tanque perteneciente al sistema “Asovapp”.

El registro se realiza por medio de un formulario en el que la administradora debe llenar una serie de campos con los datos, a su vez esta opción permite ver un historial de todos los asociados el que permite realizar modificaciones o borrar cada uno de los registros.

Al seleccionar el nombre de uno de los asociados se obtiene la siguiente ventana en la cual se puede registrar la leche diaria que el socio entrega a la asociación, además permite ver el total de los litros entregados en la quincena.

## Figura 95

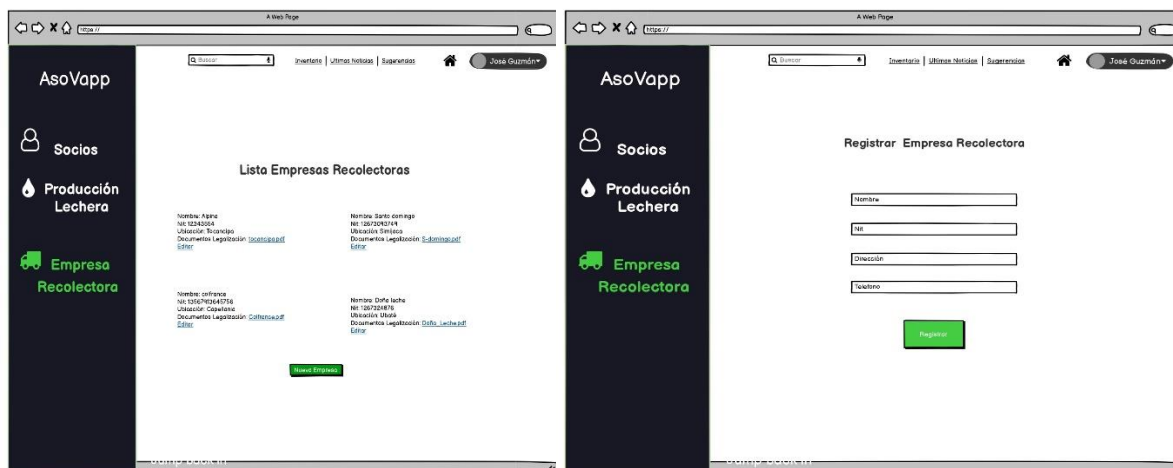
### Submódulo de Registro Leche diaria



*Nota.* Diseño realizado por el equipo de trabajo en el cual se muestra el diseño de la ventana de registro leche diaria en el módulo administradora el tanque perteneciente al sistema “Asovapp”.

**Figura 96**

*Submódulo de empresa recolectora*



*Nota.* Diseño realizado por el equipo de trabajo en el cual se muestra el diseño de la ventana de registro leche diaria en el módulo administradora el tanque perteneciente al sistema “Asovapp”.

Este modulo permite al usuario el registro de la leche total recibida por parte de los asociados y la leche total entregada a la empresa recolectora despues de enfriarla, cabe resaltar que el proceso e enfriamiento de la leche disminuye su cantidad de 3 a 7 litros para llevar este control el sistema realiza una resta entre la leche recogida y la entregada, dicho resultado se ubica al frente del registro diario, adicionalmente muestra la suma total de los tres items tenidos en cuenta en esta ventana, por ultimo el sistema permite generar un pdf de esta información.

**Figura 97**

*Submódulo de registro de leche entregada*

The screenshot shows a web browser window with the AsoVapp logo and navigation menu on the left. The main content area is titled 'Registrar Litros de leche vendidos' and 'Santo Domingo'. It features a table with the following data:

Fecha	Cantidad Litros	Precio
23/04/2022	3500	\$ 2000

Below the table is a green button labeled 'Registrar'.

*Nota.* Diseño realizado por el equipo de trabajo en el cual se muestra el diseño de la ventana de registro leche diaria recogida y entregada en el módulo administradora el tanque perteneciente al sistema “Asovapp”.

**Figura 98**

*Submódulo animales*

The left screenshot shows the 'Animales' sub-module with a 'Buenos Días' greeting and a cow image. The right screenshot shows the 'Registrar Animal' form with the following fields:

- Nombre
- Raza
- Fecha de nacimiento
- Adjuntar imagen
- Observaciones

Below the form is a green button labeled 'Registrar'.

*Nota.* Diseño realizado por el equipo de trabajo en el cual se muestra el diseño de la ventana Opción Animales en la cual se encuentran ventanas que son historial, registro animal y añadir nuevas observaciones en el módulo ganadero perteneciente al sistema “Asovapp”.

El tercer módulo de este sistema se denomina ganadero el cual cuenta con dos opciones la primera es animales la cual permite registrar por medio de un formulario los animales con los que el ganadero cuenta la finca asimismo puede registrar por medio de otro formulario las observaciones de cada animal ya sean enfermedades vacunas o cualquier tipo de procedimiento que se le realiza a el animal

## **Resultados pruebas**

Yeimy Constanza Ángel Sánchez y Jairo Orlando Pascagaza Barriga

Programa de ingeniería de sistemas, Universidad de Cundinamarca

Trabajo de grado

Director

Cristian Eduardo Cano López

Junio 2, 2023

Se realizaron algunas preguntas el día que se realizó la visita a las instalaciones de la asociación donde se le aplicó **a tres socios y a la administradora de tanque** una muestra de la población objetivo del sistema de información AsoVapp.

- ¿El sistema de información es de fácil uso o encuentra alguna dificultad para utilizarlo?
- ¿Encuentra que el sistema AsoVapp es útil para llevar un registro y seguimiento de la producción lechera?
- ¿Ha encontrado algún problema o error mientras hace uso del sistema?
- ¿El sistema cumple con las funcionalidades que espera? ¿Por qué?

**Primera entrevista fue realizada al socio Antonio Martínez:**

¿El sistema de información es de fácil uso o encuentra alguna dificultad para utilizarlo?

Respuesta: Al principio se hace difícil por lo que no estoy diariamente haciendo uso de este tipo de aplicaciones, pero el estilo que tiene me permite familiarizarme y así querer aprender a usarlo bien y sacarle el mejor provecho ya que es de gran ayuda para nosotros.

¿Encuentra que el sistema AsoVapp es útil para llevar un registro y seguimiento de la producción lechera?

Respuesta: Si es útil para llevar un registro y seguimiento de la producción lechera, ya que me permite tener un control más preciso de las cantidades de mi producción lechera.

¿Ha encontrado algún problema o error mientras hace uso del sistema?

Respuesta: No, pero al momento de registrar es un poco lento, pero más por la señal del internet que llega hasta aquí.

¿El sistema cumple con las funcionalidades que espera? ¿Por qué?

Respuesta: creo que, si porque se puede llevar un registro detallado de la producción lechera, gestionar la información de manera eficiente y generar informes.

### **Segunda entrevista fue realizada al socio Jairo Contreras:**

¿El sistema de información es de fácil uso o encuentra alguna dificultad para utilizarlo?

Respuesta: Si es de fácil uso ya que el sistema se da a entender y le va enseñando a uno mismo día a día.

¿Encuentra que el sistema AsoVapp es útil para llevar un registro y seguimiento de la producción lechera?

Respuesta: Si ya que puedo registrar mis animales y asi mismo ver la cantidad de producción lechera que la señora encargada del tanque me anotar cada día.

¿Ha encontrado algún problema o error mientras hace uso del sistema?

Respuesta: No, pero al momento de ingresar es lento porque la señal de los datos no es muy buena.

¿El sistema cumple con las funcionalidades que espera? ¿Por qué?

Respuesta: si claro, porque no conocía que estos sistemas pudieran hacer todo eso y me ha dejado sorprendido, contento con estas tecnologías.

### **Tercera entrevista fue realizada al socio Álvaro Fresneda:**

¿El sistema de información es de fácil uso o encuentra alguna dificultad para utilizarlo?

Respuesta: se me hace un poco difícil, pero si practico cada día, eso aprendo es a usar la aplicación es rápido y asi aprovechar este sistema ya que es muy interesante y beneficioso para nosotros.

¿Encuentra que el sistema AsoVapp es útil para llevar un registro y seguimiento de la producción lechera?

Respuesta: claro, me permite consultar los litros de la producción lechera que han sido vendidos a la asociación y así cada quincena me permite rectificar que estén *anotados* bien.

¿Ha encontrado algún problema o error mientras hace uso del sistema?

Respuesta: No ninguno está claro cada sección.

¿El sistema cumple con las funcionalidades que espera? ¿Por qué?

Respuesta: sí me parece que se ve completo con lo que habíamos planeado en la primera visita que nos hicieron ustedes.

**Certificado de Registro en la Dirección Nacional de Derechos de Autor**

Yeimy Constanza Ángel Sánchez y Jairo Orlando Pascagaza Barriga

Programa de ingeniería de sistemas, Universidad de Cundinamarca

Trabajo de grado

Director

Cristian Eduardo Cano López

Junio 2, 2023

 <b>MINISTERIO DEL INTERIOR</b> <b>DIRECCIÓN NACIONAL DE DERECHO DE AUTOR</b> <b>UNIDAD ADMINISTRATIVA ESPECIAL</b> <b>OFICINA DE REGISTRO</b>		Libro - Tomo - Partida <b>13-94-331</b> Fecha Registro <b>30-may.-2023</b>
<b>CERTIFICADO DE REGISTRO DE SOPORTE LOGICO - SOFTWARE</b>		
Page 1 of 2		
<b>1. DATOS DE LAS PERSONAS</b>		
<b>AUTOR</b>		
Nombres y Apellidos	CRISTIAN EDUARDO CANO LOPEZ	No de identificación CC
Nacional de	COLOMBIA	Ciudad:
Dirección		
<b>AUTOR</b>		
Nombres y Apellidos	YEIMY CONSTANZA ANGEL SANCHEZ	No de identificación CC
Nacional de	COLOMBIA	Ciudad:
Dirección		
<b>AUTOR</b>		
Nombres y Apellidos	JAIRO ORLANDO PASCAGAZA BARRIGA	No de identificación CC
Nacional de	COLOMBIA	Ciudad:
Dirección		
<b>PRODUCTOR</b>		
Nombres y Apellidos	CRISTIAN EDUARDO CANO LOPEZ	No de identificación CC
Nacional de	COLOMBIA	Ciudad:
Dirección		
<b>PRODUCTOR</b>		
Nombres y Apellidos	YEIMY CONSTANZA ANGEL SANCHEZ	No de identificación CC
Nacional de	COLOMBIA	Ciudad:
Dirección		
<b>PRODUCTOR</b>		
Nombres y Apellidos	JAIRO ORLANDO PASCAGAZA BARRIGA	No de identificación CC
Nacional de	COLOMBIA	Ciudad:
Dirección		
<b>2. DATOS DE LA OBRA</b>		
Título Original	SISTEMA DE INFORMACIÓN PARA SEGUIMIENTO Y CONTROL DE LOS PROCESOS ADMINISTRATIVOS DE LA ASOCIACIÓN ASOAGROPROLECHE DEL MUNICIPIO DE CARMEN DE CARUPA ASOVAPP	
Año de Creación	2023	Pais de Origen COLOMBIA Año Edición
CLASE DE OBRA	INEDITA	
CARACTER DE LA OBRA	OBRA ORIGINARIA	
CARACTER DE LA OBRA	OBRA EN COLABORACION	
ELEMENTOS APORTADOS DE SOPORTE LOGICO	PROGRAMA DE COMPUTADOR	
<b>3. DESCRIPCIÓN DE LA OBRA</b>		
SISTEMA DE INFORMACIÓN EL CUAL LE PERMITE A LOS USUARIOS REALIZAR SEGUIMIENTO Y CONTROL A LAS ACTIVIDADES ADMINISTRATIVAS QUE DESEMPEÑA DENTRO DE LA ASOCIACIÓN DE GANADEROS ASOAGROPROLECHE, MEDIANTE EL REGISTRO DE LOS DATOS QUE RECOLECTA O GENERA AL REALIZARLA, FACILITANDO ASÍ LA ORGANIZACIÓN DE CADA UNO DE LOS PROCESOS.		
<b>4. OBSERVACIONES GENERALES DE LA OBRA</b>		