

# **Proyecto final Ingeniería del Software**

Desarrollo de un sistema de información Web para la gestión y trazabilidad del levante de gallinas ponedoras de granjas avícolas - SIGEAVI

## **Presentado por:**

Nicolás David Fajardo Acuña - 201920898

Cristian David Parada Martínez - 201921141

Daniel Felipe Suarez Bohórquez - 201920229

Ingeniería de Sistemas y Computación – VII Semestre

## **Presentado a:**

Ing. Julián Monsalve

Universidad Pedagógica Y Tecnológica de Colombia  
Facultad de ingeniería  
Escuela de Ingeniería de Sistemas y Computación

## Tabla de contenido

<b>Proyecto final Ingeniería del Software</b>	<b>1</b>
<b>Título de proyecto</b>	<b>4</b>
<b>Objetivos del proyecto</b>	<b>4</b>
<i>Objetivo General</i>	<i>4</i>
<i>Objetivos Específicos</i>	<i>4</i>
Justificación	5
Problema	5
Metodología	6
<b>Proceso de elicitación de requerimientos</b>	<b>9</b>
<b>Modelos del sistema</b>	<b>10</b>
Diagrama de contexto	10
<b>Diagramas de Casos de Uso</b>	<b>11</b>
<b>Diagramas de actividades</b>	<b>12</b>
<b>Modelo E/R</b>	<b>13</b>
Diagrama de clases	14
Pruebas realizadas al proceso y al producto desarrollado	15
Plan de Pruebas:	15
Pruebas unitarias	16
Pruebas de Integración	31
<b>Conclusiones</b>	<b>33</b>
<b>Referencias</b>	<b>34</b>
<b>Anexos:</b>	<b>35</b>
Anexo 1: Entrevista primer encuentro:	35
Anexo 2: Documento IEEE 830	37
Documento IEEE 830 para un Sistema de información Web para la gestión y trazabilidad de granjas avícolas	37
1. Introducción	37

2. Propósito	37
3. Alcance	37
4. Personal involucrado	38
5. Definiciones, acrónimos y abreviaturas	38
6. Referencias	38
7. Resumen	39
8. Descripción general	39
9. Perspectiva del producto	39
10. Funcionalidad del producto	40
11. Características de los usuarios	41
12. Restricciones	41
13. Suposiciones y dependencias	41
14. Requisitos específicos	42
14.1 Requisitos comunes de las interfaces	50
14.1.1 Interfaces de usuario	50
14.1.2 Interfaces de hardware	50
14.1.3 Interfaces de software	50
14.1.4 Interfaces de comunicación	51
14.2 Requisitos funcionales	51
14.3 Requisitos no funcionales	52
14.3.1 Requisitos de Interfaz	52
14.3.2 Ayuda en el uso del sistema	52
14.3.3 Encabezado del sistema.	53
14.3.4 Diseño de la interfaz a la característica de la tienda tecnológica.	53
14.3.5 Desempeño	53
14.3.6 Nivel de usuario	53
14.3.7 Confiabilidad continua del sistema	53
14.3.8 Seguridad de información	54
15. Priorización de requisitos	54

## **Título de proyecto**

“Desarrollo de un sistema de información Web para la gestión y trazabilidad del levante de gallinas ponedoras de granjas avícolas - SIGEAVI”

## **Objetivos del proyecto**

### ***Objetivo General***

Sistematizar el proceso de gestión del espacio, control de alimentación y la trazabilidad del crecimiento de gallinas ponedoras desde el primer día de nacidas hasta la semana 17 de vida, con el fin de facilitar y optimizar el trabajo de los administradores y trabajadores de la granja.

### ***Objetivos Específicos***

- Analizar, examinar y recolectar los requerimientos del software para la gestión y trazabilidad del levante de gallinas ponedoras utilizando el estándar IEEE 830.
- Proponer y construir diagramas de clases, casos de uso, actividad y entidad relación para el diseño de un software que permita modelar el sistema propuesto utilizando el estándar UML
- Codificar e implementar el software propuesto para obtener un sistema orientado a la Web que cumpla con los diseños y los requerimientos de los objetivos anteriores, utilizando técnicas y herramientas de ingeniería de software.
- Aplicar un plan de pruebas donde se pueda evidenciar el correcto funcionamiento de cada uno de los requerimientos del software realizando pruebas unitarias y de integración.

## **Justificación**

Este proyecto surge de la necesidad por parte de algunos administradores de granjas avícolas de manejar un sistema que les permitiera optimizar, facilitar y potenciar las actividades que se realizan en la granja, tales como el manejo del inventario de alimentos, el control de mortalidad de las aves, el seguimiento de la vacunación de las mismas, la gestión del espacio y por último los análisis que permitían visualizar a los administradores cual es el estado de los lotes y en qué aspectos se tiene que mejorar para aumentar la productividad de la granja. Otro factor crucial que lleva a desarrollar un sistema tecnológico es la alta competitividad que se encuentra actualmente en el mercado donde es crucial tener datos correctos y aves completamente saludables que sean capaces de producir huevos de altísima calidad y garantizarles a los clientes un excelente servicio.

## **Problema**

Se identificó una deficiencia en la gestión y control de la trazabilidad del levante de gallinas ponedoras en las granjas avícolas de la región causadas por un manejo de la información en planillas físicas, error humano en el almacenamiento y cálculo de la información, incumplimiento de las normas del espacio necesario para las aves y por una falta en el manejo de inventario de alimentos y de las vacunas. Lo anterior tiene un efecto negativo, ya que ocasiona pérdida de la información, falta de seguridad en la misma, un aumento en la mortalidad de las aves, inconsistencia en la contabilidad y pérdidas económicas ya que no hay una claridad en los costos del levante de las aves.

## Metodología

Para el desarrollo del proyecto se hizo uso de la metodología SCRUM, debido a que esta es el marco de trabajo que mejor se adapta a las necesidades del proyecto, ya que al ser una metodología ágil está pensada para proyectos en los cuales es más fundamental el funcionamiento del sistema final que la documentación excesiva (lo que aplica debido al poco tiempo), también, al no contar con mucho tiempo para el desarrollo del proyecto, posiblemente en el transcurso del desarrollo del mismo iban a haber muchos cambios respecto a lo planeado inicialmente, lo cual se ajusta mucho a las metodologías ágiles como SCRUM. Inicialmente en una reunión hecha por el equipo de trabajo se definió el Sprint Backlog del proyecto el cual quedó dividido en cuatro Sprints, de los cuales para el sprint 1 se definió un tiempo de desarrollo de 2 semanas mientras que para los sprint 2, 3 y 4 se les definió un tiempo de desarrollo de una semana para cada uno. Se da inicio al proyecto el 31 de octubre del 2022 y definiendo la entrega del mismo para el día 5 de diciembre del 2022. En la reunión realizada por el equipo se definió que se iba a utilizar el software JIRA para la gestión del proyecto ya que esta herramienta es ideal para trabajar este marco de trabajo y en la misma reunión se definieron las actividades para cada uno de los sprint dividiéndolas de la siguiente manera:

Sprint 1:	31/10/2022 - 14/11/2022
<ul style="list-style-type: none"><li>● Epic 1: Levantamiento de requerimientos:<ul style="list-style-type: none"><li>○ Tarea 1: Licitación</li><li>○ Tarea 2: Desarrollo de documento IEEE 830</li><li>○ Tarea 3: Modelado UML</li></ul></li><li>● Epic 2: Autenticación:<ul style="list-style-type: none"><li>○ Tarea 1: Registro de usuarios :<ul style="list-style-type: none"><li>■ Subtarea 1: Diseño</li><li>■ Subtarea 2: Desarrollo</li><li>■ Subtarea 3: Pruebas</li></ul></li><li>○ Tarea 2: Inicio de sesión (Login)<ul style="list-style-type: none"><li>■ Subtarea 1: Diseño</li><li>■ Subtarea 2: Desarrollo</li><li>■ Subtarea 3: Pruebas</li></ul></li><li>○ Tarea 3: Recuperación de contraseña:<ul style="list-style-type: none"><li>■ Subtarea 1: Diseño</li><li>■ Subtarea 2: Desarrollo</li><li>■ Subtarea 3: Pruebas</li></ul></li></ul></li></ul>	<p>Encargado Nicolas Fajardo Encargado : Nicolas Fajardo Encargado : Daniel Suarez</p> <p>Encargado : Cristian Parada Encargado : Nicolas Fajardo Encargado : Daniel Suarez</p> <p>Encargado : Cristian Parada Encargado : Nicolas Fajardo Encargado : Daniel Suarez</p> <p>Encargado : Cristian Parada Encargado : Daniel Suarez Encargado : Nicolas Fajardo</p>
Sprint 2:	14/11/2022 -21/11/2022
<ul style="list-style-type: none"><li>● Epic 3: Gestión de granjas:<ul style="list-style-type: none"><li>○ Tarea 1: CRUD granjas<ul style="list-style-type: none"><li>■ Subtarea 1: Diseño</li><li>■ Subtarea 2: Desarrollo</li><li>■ Subtarea 3: Pruebas</li></ul></li></ul></li></ul>	<p>Encargado : Cristian Parada Encargado : Daniel Suarez Encargado : Nicolas Fajardo</p>

- Tarea 2: CRUD galpones
  - Subtarea 1: Diseño Encargado : Cristian Parada
  - Subtarea 2: Desarrollo Encargado : Daniel Suarez
  - Subtarea 3: Pruebas Encargado : Nicolas Fajardo
- Tarea 3: CRUD lotes
  - Subtarea 1: Diseño Encargado : Cristian Parada
  - Subtarea 2: Desarrollo Encargado: Daniel Suarez
  - Subtarea 3: Pruebas Encargado: Nicolas Fajardo
- Tarea 4: Calcular cantidad de aves para un determinado espacio y calcular el espacio necesario para una determinada cantidad de aves
  - Subtarea 1: Diseño Encargado : Cristian Parada
  - Subtarea 2: Desarrollo Encargado : Daniel Suarez
  - Subtarea 3: Pruebas Encargado: Nicolas Fajardo

### Sprint 3

21/11/2022 - 28/11/2022

- Epic 4: Gestión Alimenticias de las aves:
  - Tarea 1: CRUD inventario
    - Subtarea 1: Diseño Encargado : Cristian Parada
    - Subtarea 2: Desarrollo Encargado : Nicolas Fajardo
    - Subtarea 3: Pruebas Encargado: Daniel Suarez
  - Tarea 2: Control de peso semanal
    - Subtarea 1: Diseño Encargado : Cristian Parada
    - Subtarea 2: Desarrollo Encargado: Daniel Suarez
    - Subtarea 3: Pruebas Encargado: Nicolas Fajardo
  - Tarea 3: Control de alimentación diaria
    - Subtarea 1: Diseño Encargado : Cristian Parada
    - Subtarea 2: Desarrollo Encargado: Nicolas Fajardo
    - Subtarea 3: Pruebas Encargado: Daniel Suarez
  - Tarea 4: Control de consumo de agua
    - Subtarea 1: Diseño Encargado : Cristian Parada
    - Subtarea 2: Desarrollo Encargado : Nicolas Fajardo
    - Subtarea 3: Pruebas Encargado : Daniel Suarez

### Sprint 4

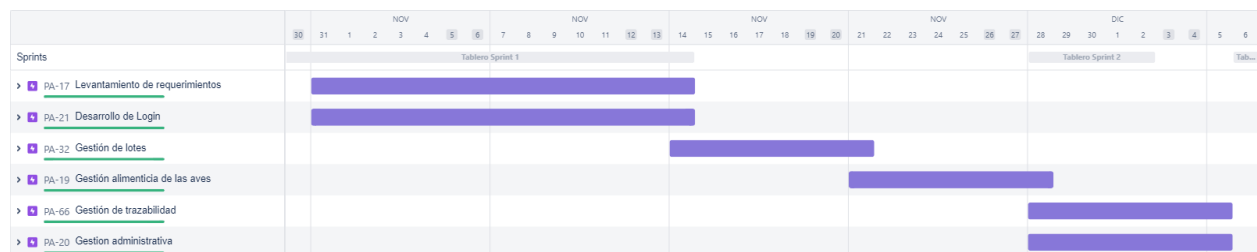
28/11/2022 - 05/12/2022

- Epic 5: Gestión de Trazabilidad
  - Tarea 1: Control de mortalidad diaria
    - Subtarea 1: Diseño Encargado : Cristian Parada
    - Subtarea 2: Desarrollo Encargado ; Nicolas Fajardo
    - Subtarea 3: Pruebas Encargado : Daniel Suarez
  - Tarea 2: Plan de vacunación
    - Subtarea 1: Diseño Encargado : Cristian Parada
    - Subtarea 2: Desarrollo Encargado : Daniel Suarez
    - Subtarea 3: Pruebas Encargado : Nicolas Fajardo

- Epic 6: Gestión administrativa
  - Tarea 1: Generación de reportes de consumo de alimentos
    - Subtarea 1: Diseño Encargado : Cristian Parada
    - Subtarea 2: Desarrollo Encargado : Daniel Suarez
    - Subtarea 3: Pruebas Encargado : Nicolas Fajardo
  - Tarea 1: Generación de reportes de consumo de agua
    - Subtarea 1: Diseño Encargado : Cristian Parada
    - Subtarea 2: Desarrollo Encargado : Daniel Suarez
    - Subtarea 3: Pruebas Encargado : Nicolas Fajardo
  - Tarea 3: Reporte de trazabilidad (mortalidad)
    - Subtarea 1: Diseño Encargado : Cristian Parada
    - Subtarea 2: Desarrollo Encargado : Daniel Suarez
    - Subtarea 3: Pruebas Encargado : Nicolas Fajardo

**NOTA:** Debido al poco tiempo para el desarrollo del proyecto, se limitaron las funcionalidades entregables a las descritas anteriormente. Las funcionalidades de control de costos, gestión de historial de vacunación y sus reportes podrán ser implementadas en futuras versiones.

Cronograma de actividades:



Arquitectura de desarrollo:

Para el desarrollo del proyecto se usó una arquitectura cliente/servidor dividiendo en dos capas el Frontend, el Backend. El Frontend fue desarrollado en JavaScript usando el framework de Angular para su desarrollo. El Backend fue desarrollado en JavaScript usando el entorno de ejecución de Node Js. La base de datos está relacionada directamente con el backend y fue desarrollada en MySQL y fué desplegada en el servicio en la nube de db4free.net.



## Proceso de elicitación de requerimientos

Para esta sección del documento se hablará sobre los encuentros, reuniones o llamadas que el equipo de desarrollo realizó con diferentes administradores y granjeros de la región para recolectar la información necesaria para el desarrollo del proyecto. Donde se discutieron temas acerca del modelo de negocios que se maneja en la granja, lo que se espera que haga el sistema, las personas que van a hacer uso e interactuarán con el sistema y los permisos que deberían tener estos usuarios.

En una primera reunión realizada en la ciudad de Moniquirá (Boyacá) el día 1 de noviembre de 2022, se le realizaron las siguientes preguntas con respecto al sistema y su modelo de negocio, en donde se obtuvieron las respuestas registradas en el Anexo 1.

Una vez culminada esta entrevista, se tenía gran claridad sobre lo que se quería y se debía hacer, sin embargo, al no conocer al detalle la información que se debía solicitar en cada parte del sistema se realizó una llamada de manera más informal con los interesados en donde se platicó sobre estos aspectos, esta llamada se realizó el jueves 3 de noviembre de 2022, en donde se concretó en definidas cuentas la siguiente información:

- ✓ La información que se debe tener de cada lote es la fecha de llegada, la cantidad y el galpón asignado.
- ✓ La información que se debe almacenar para cada granjero es el nombre, el apellido, el número de documento, el nombre de usuario, la contraseña del usuario.
- ✓ La información que se debe almacenar de la granja es el nombre, la dirección, el teléfono de contacto.
- ✓ La información que se debe almacenar para cada gasto es la fecha en que se generó, la información del usuario que la registró, la descripción del gasto, el monto y el lote al que se le destinó el gasto.
- ✓ La información del inventario en cuenta a la entrada de alimentos debe incluir la cantidad, la fecha de ingreso, el tipo de alimento y el lote al que va destinado. En cuanto a la salida se debe registrar cuántos gramos de comida se proporcionan diariamente.

Además, nos comentaron que les gustaría que el sistema les permitirá generar un reporte en un lapso de tiempo ya que es una práctica que se hace continuamente para tener una mayor claridad de la situación actual de la granja, permitiendo así compararlas con las tablas de rendimientos de los estándares del mercado. Si el sistema incluye todo esto ayudará en la toma de decisiones y permitirá ver cómo evoluciona la granja y si está dentro de los parámetros de producción rentables.

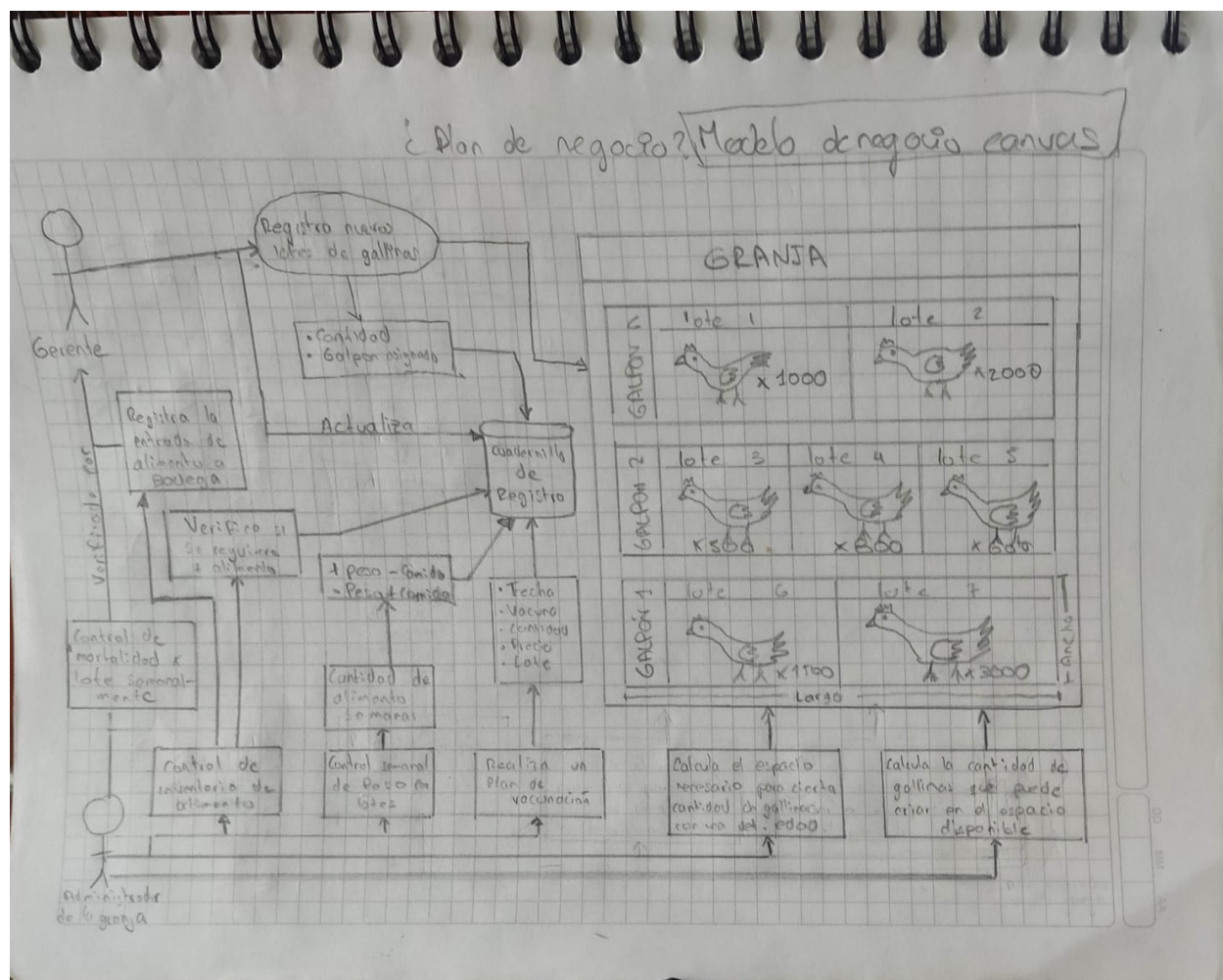
Con toda la información recopilada hasta este momento, ya se tenía un gran dominio sobre el tema y sobre lo que se esperaba del sistema por lo que se empezaron a realizar Mockups, sobre cómo interactuaría el sistema con su entorno.

El documento IEEE 830 y la priorización de los requerimientos se encuentran en el anexo 2.

## Modelos del sistema

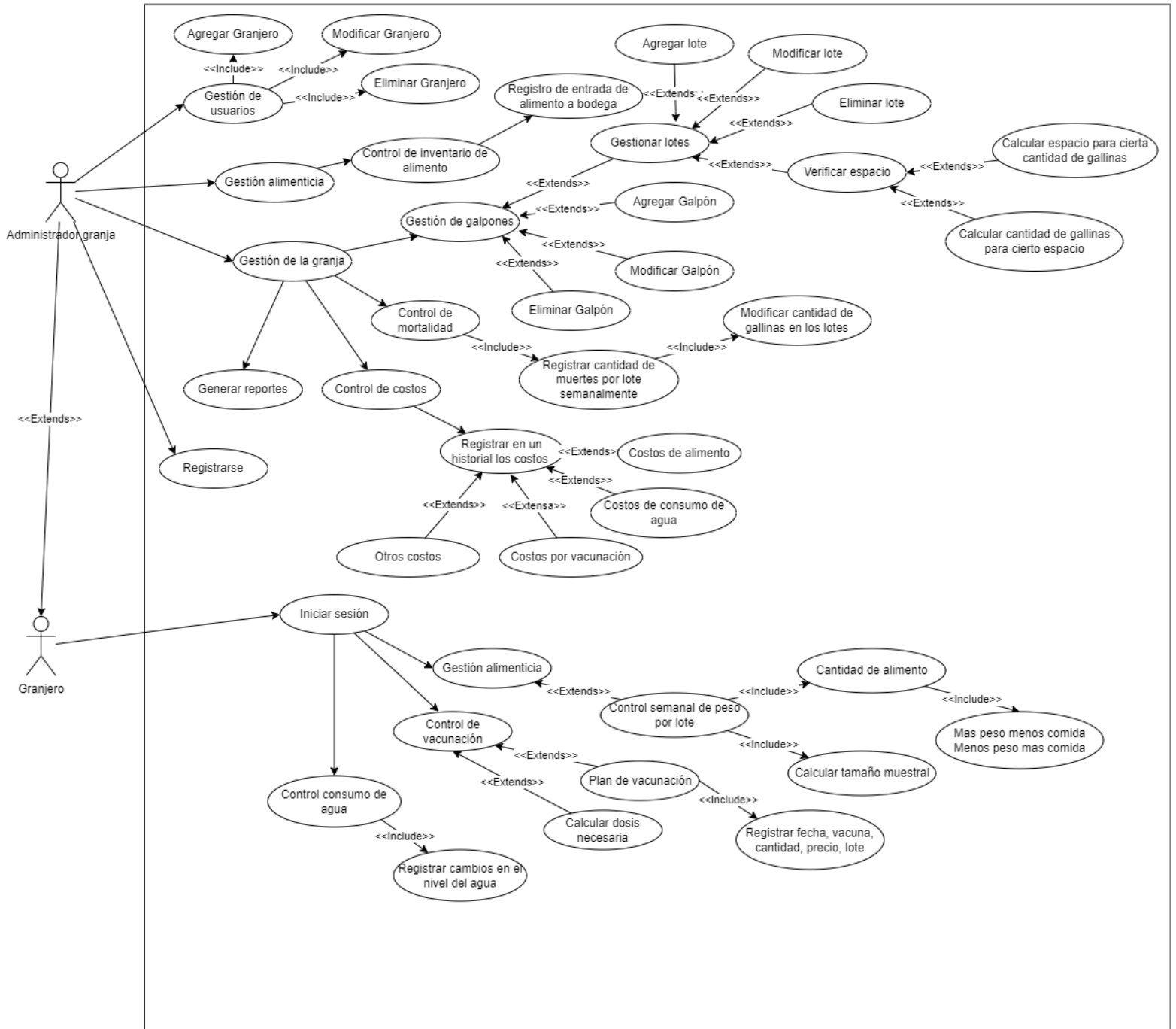
### Diagrama de contexto

En este diagrama se realiza un primer bosquejo con ayuda de la recopilación de la información obtenida de las encuestas, entrevistas y llamadas, en el cual se quiere representar una aproximación de cómo se comporta el sistema de la tienda con su entorno.



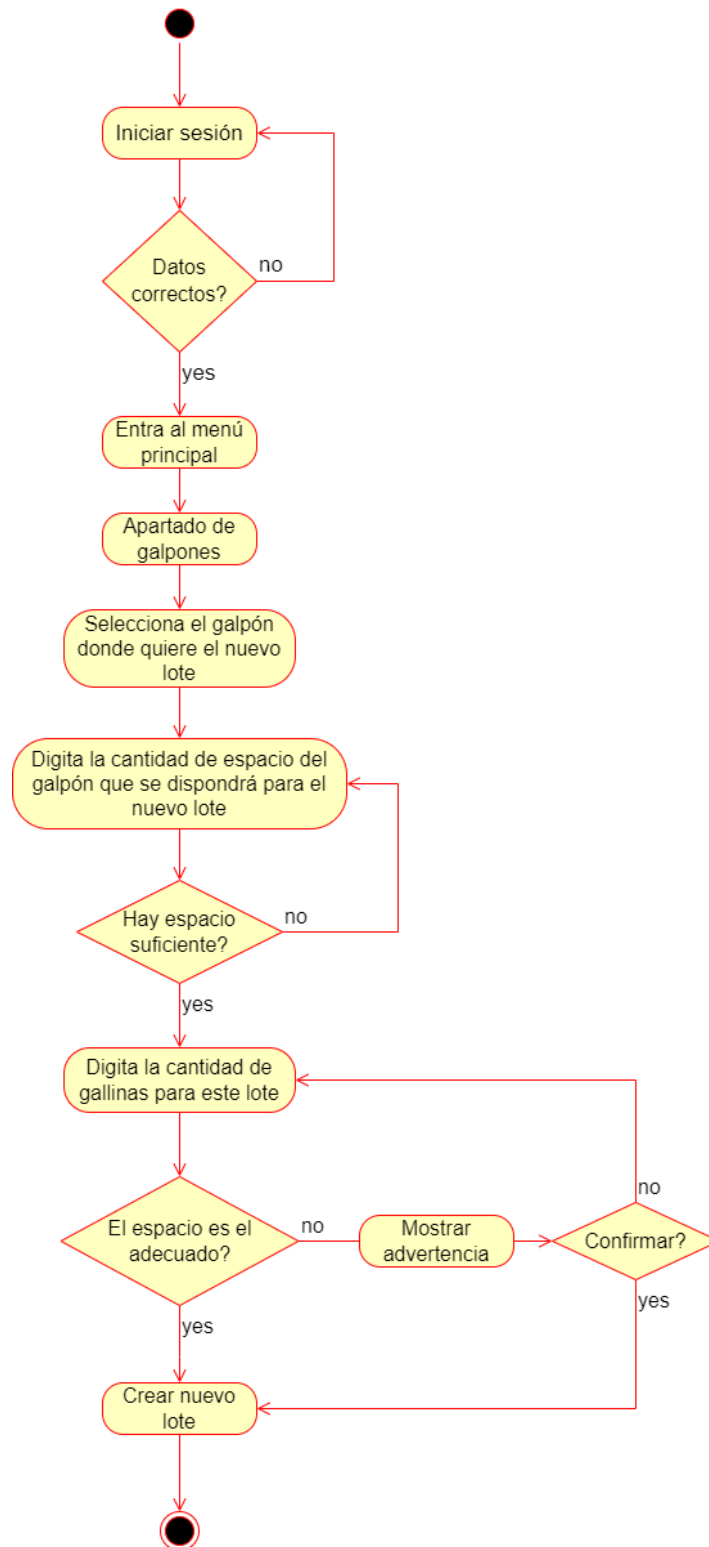
## Diagramas de Casos de Uso

### Sistema de gestión avícola

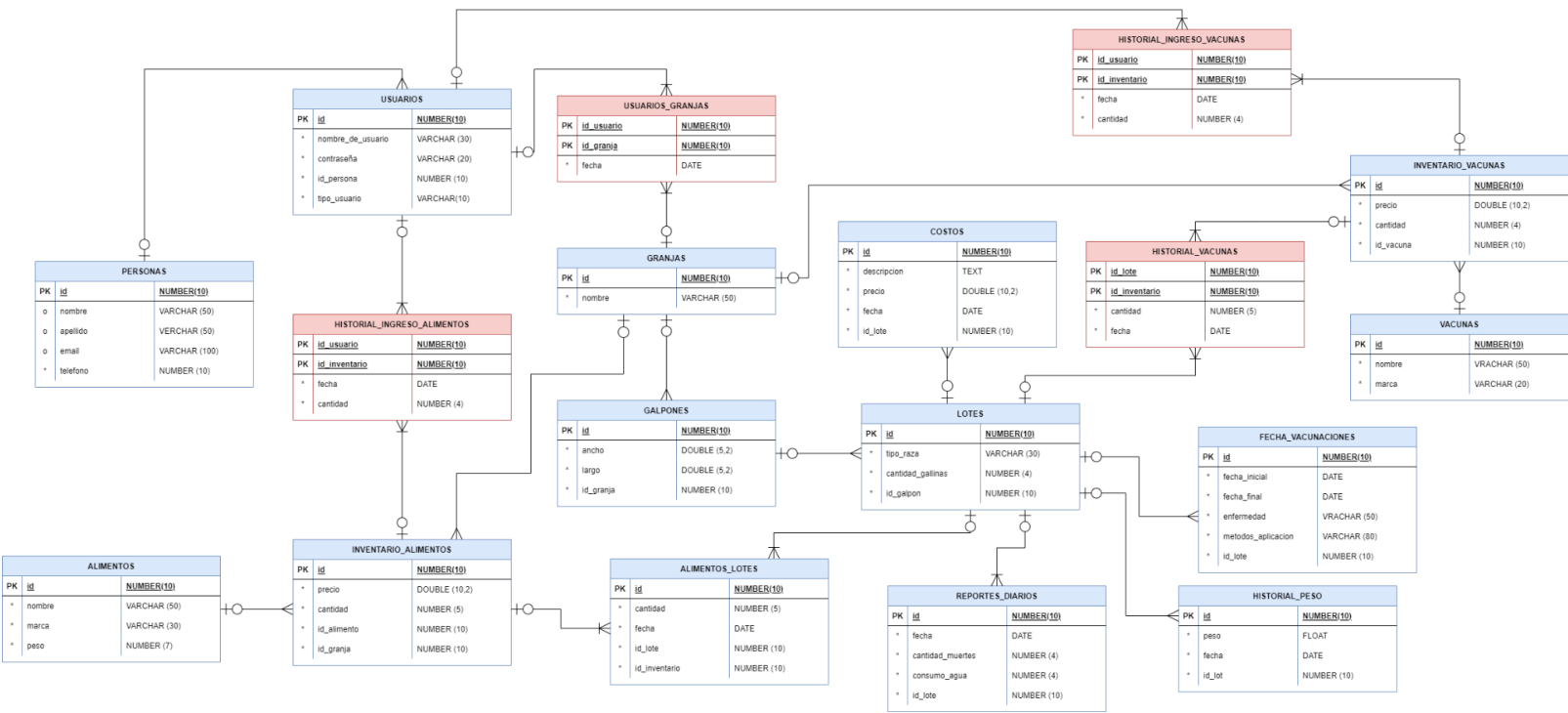


## Diagramas de actividades

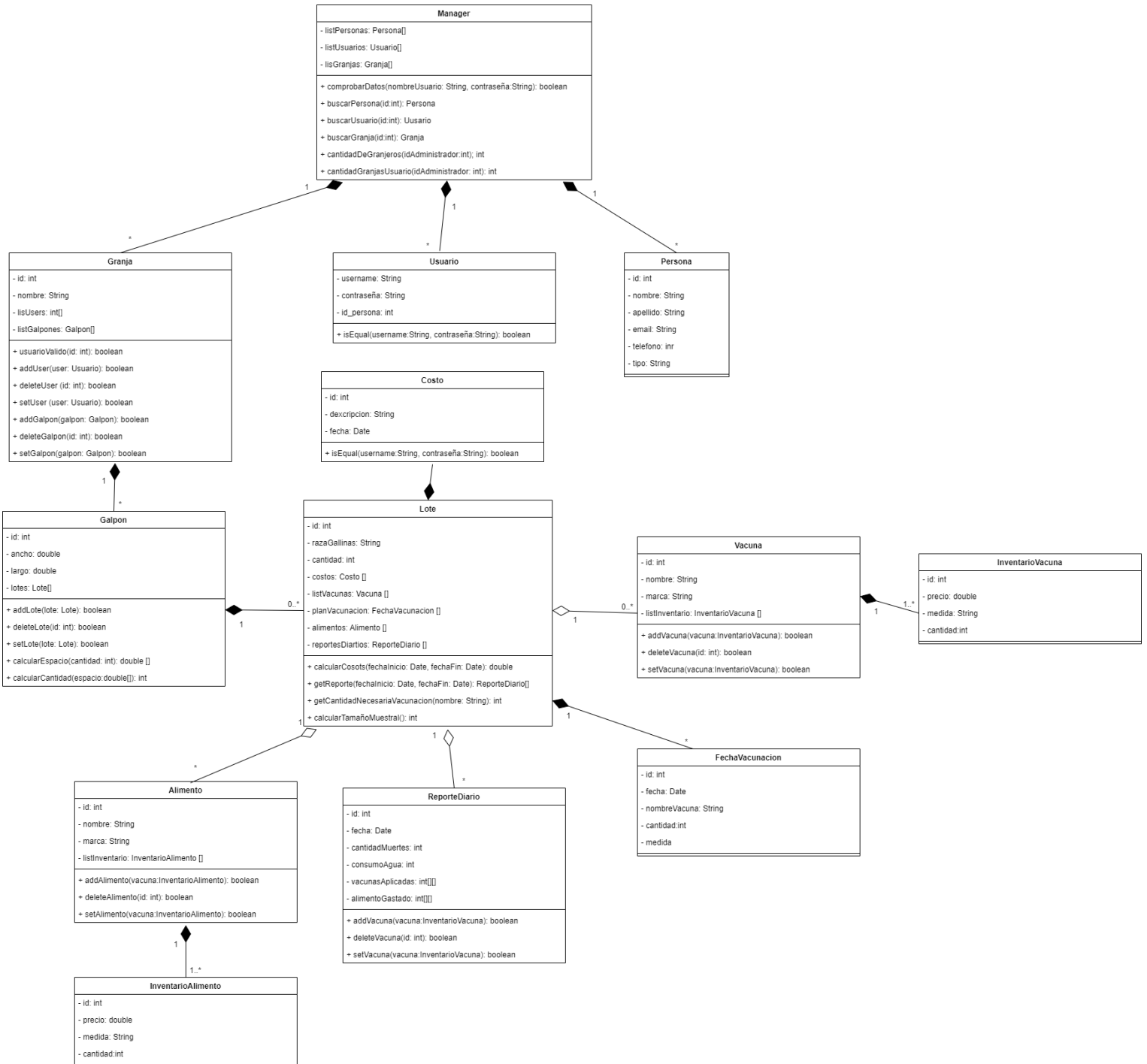
Administrador  
granja



### Modelo E/R



## Diagrama de clases



## **Pruebas realizadas al proceso y al producto desarrollado**

### ***Plan de Pruebas:***

El plan de pruebas que se realizaría se enfocará en realizar los siguientes pasos:

1. Analizar los requerimientos de desarrollo de software.
2. Identificar las funcionalidades nuevas a probar.
3. Identificar las funcionalidades de sistemas existentes que deben probarse.
4. Definir la estrategia de pruebas.

Las pruebas unitarias se realizan haciendo peticiones al servidor y verificando que los datos enviados y recibidos sean los esperados, así como su actualización correcta en la base de datos.

Las pruebas de integración se realizan verificando que la navegabilidad en la interfaz sea la indicada y que no se presenten errores en el proceso

5. Definir los criterios de inicio, aceptación y suspensión de pruebas.
6. Identificar los entornos (ambientes) requeridos.

Las pruebas unitarias se realizan en el software Postman para probar la conexión con el servidor y el ingreso de datos correctos en la base de datos.

Las pruebas de integración se realizan con el software Test Studio para probar las funcionalidades de la interfaz de usuario

7. Determinar necesidades de personal.

En cada sprint se determina el responsable de realizar las pruebas

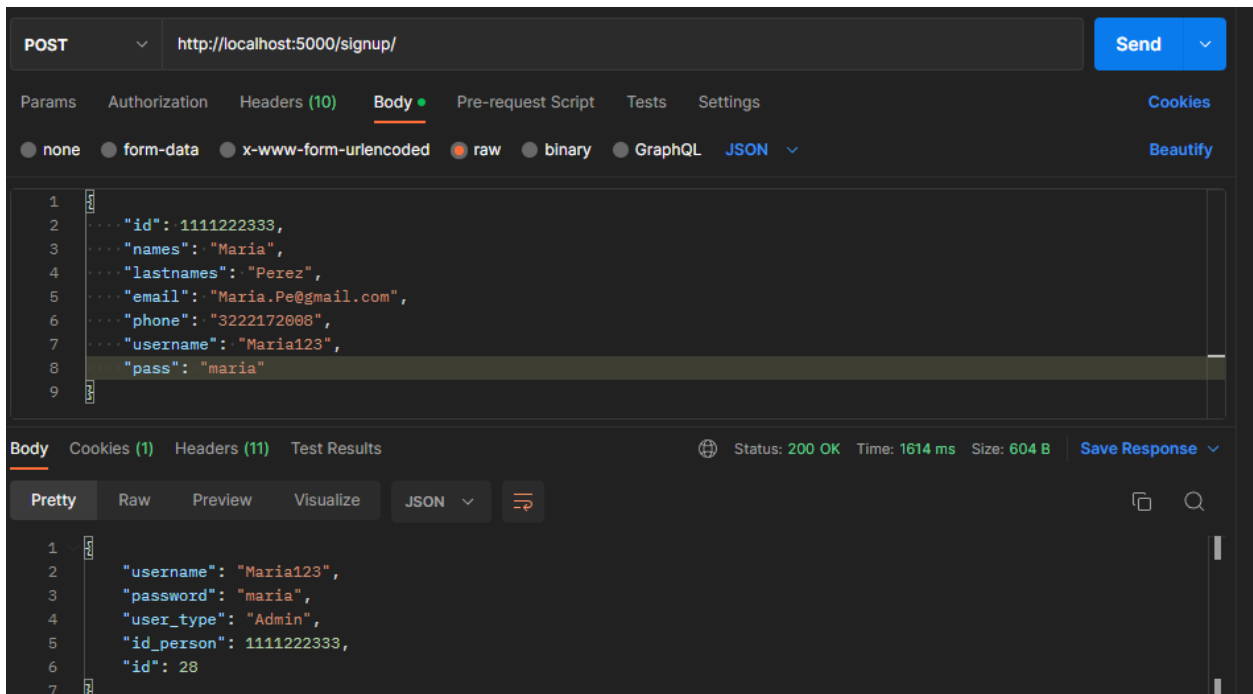
8. Establecer la metodología y procedimientos de prueba.
9. Elaborar la planificación de las pruebas.
10. Identificar los riesgos y definir planes de respuesta.

## Pruebas unitarias

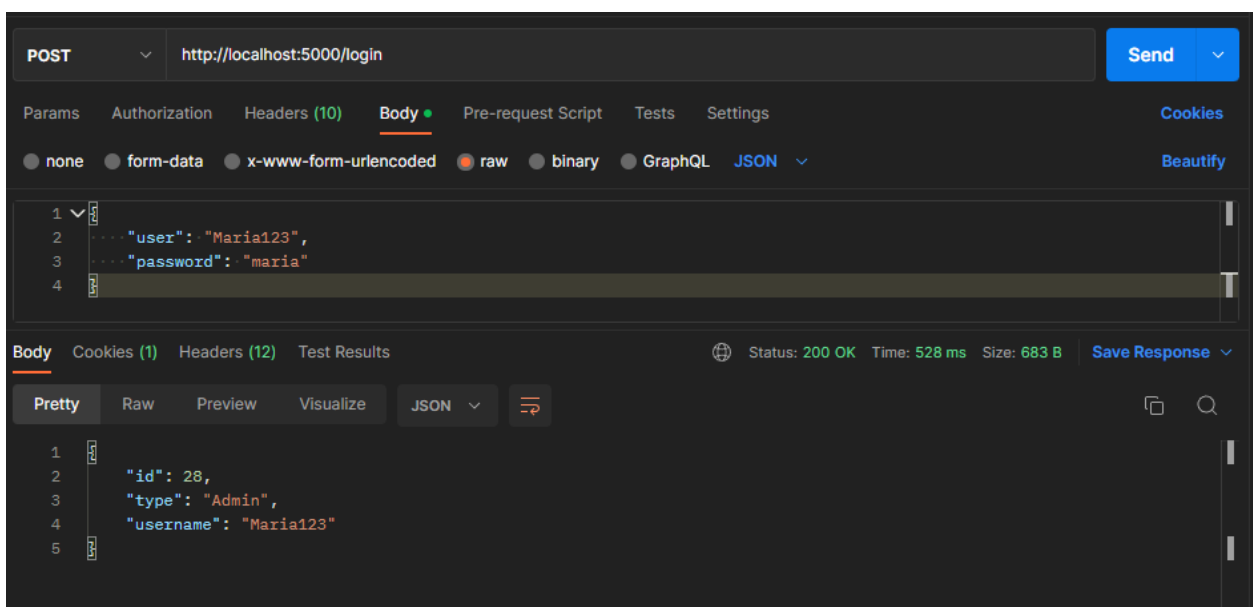
Sprint 1:

31/10/2022 - 14/11/2022

- Epic 1: Levantamiento de requerimientos - No requiere
- Epic 2: Autenticación:
  - Tarea 1: Registro de usuarios



- Tarea 2: Inicio de sesión (Login)



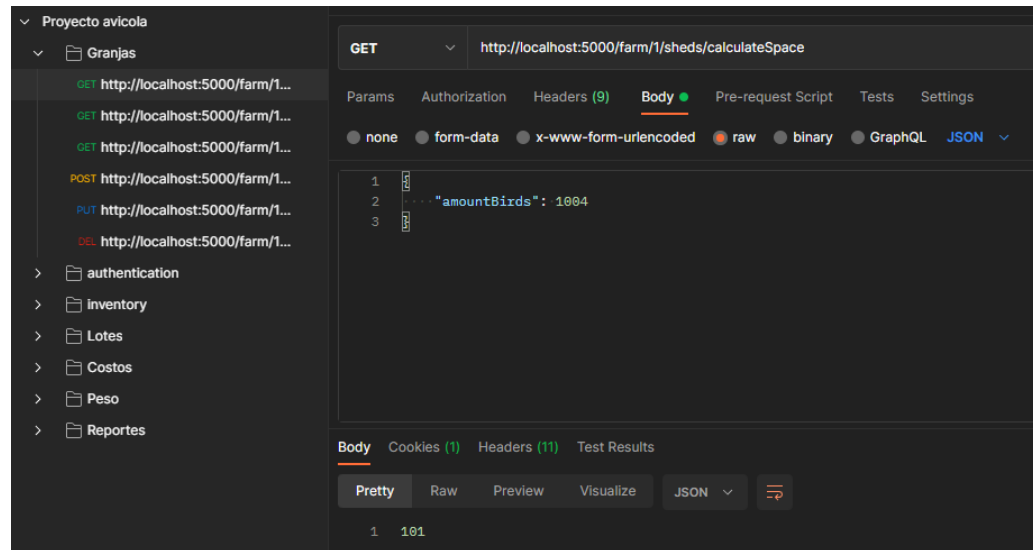


## Sprint 2:

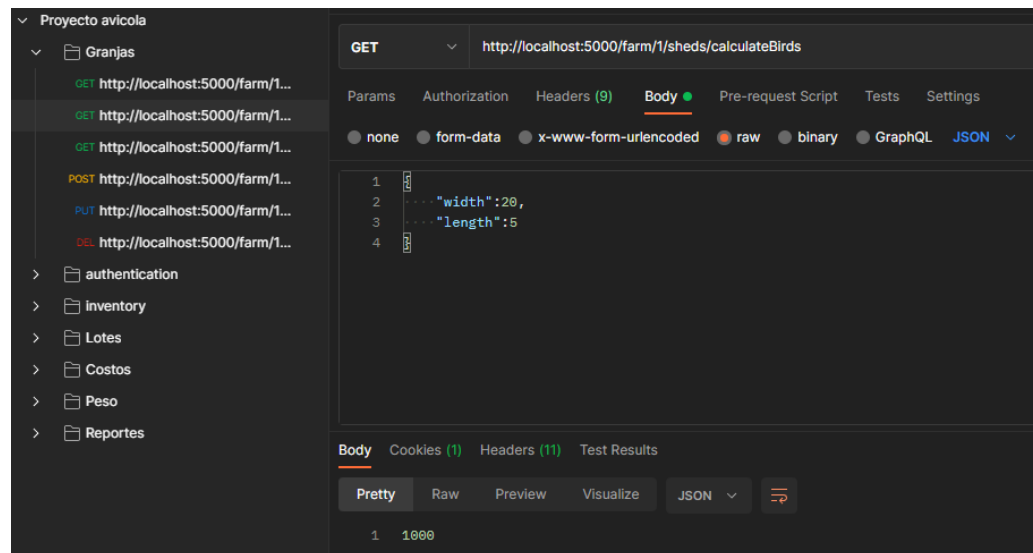
14/11/2022 -21/11/2022

- Epic 3: Gestión de granjas:
  - Tarea 1: CRUD granjas

### Obtener granjas

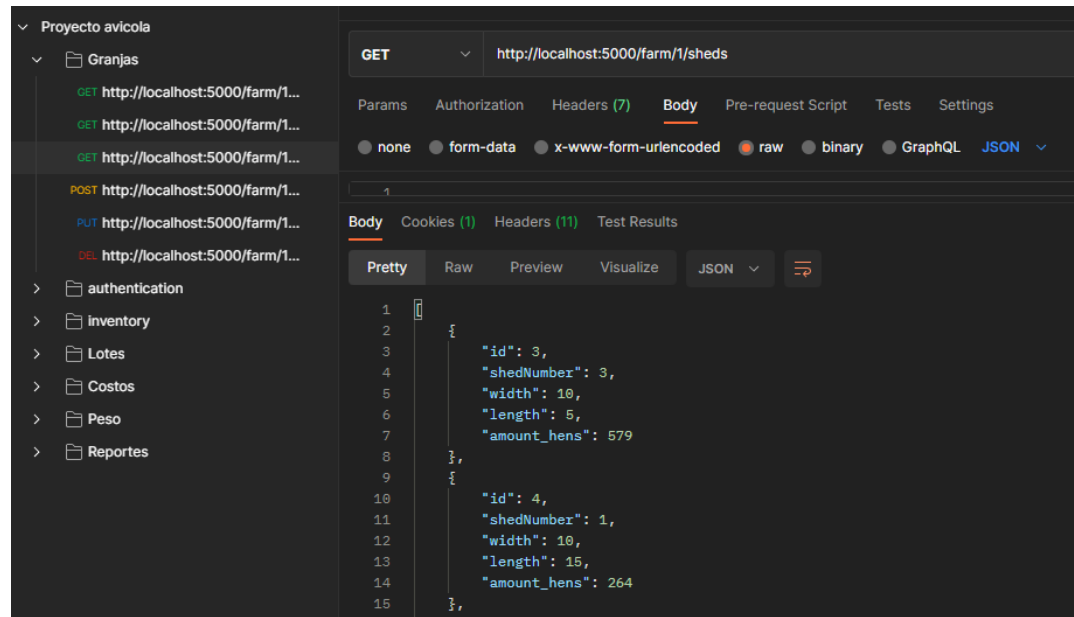


### Actualizar granja

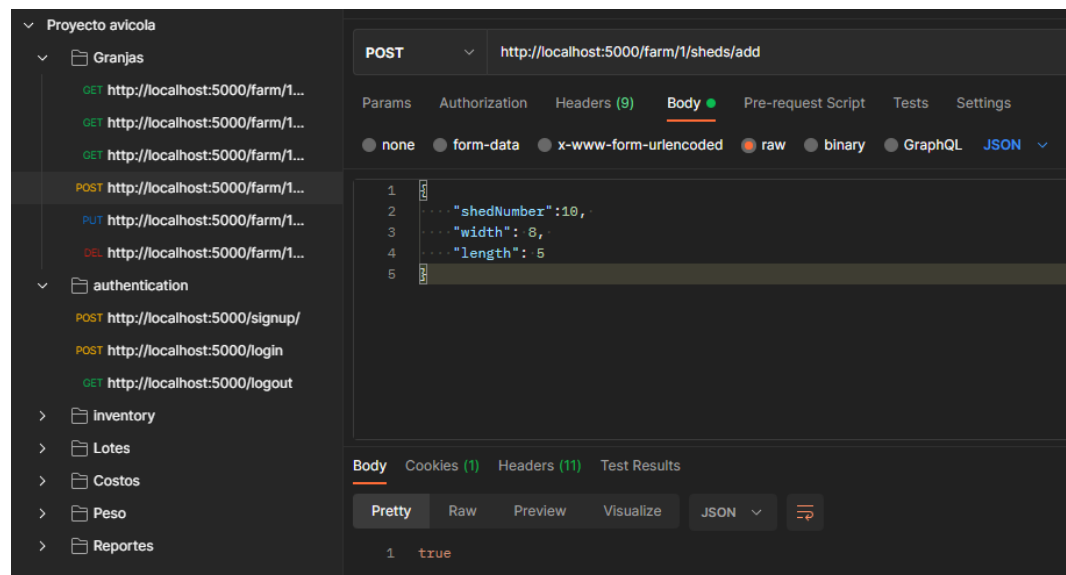


- Tarea 2: CRUD galpones

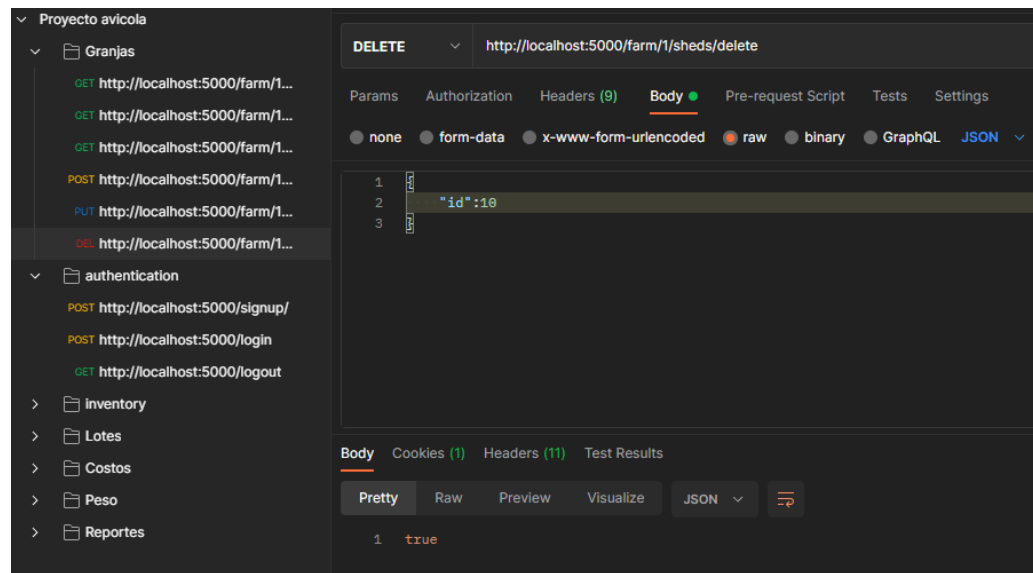
### Obtener Galpones



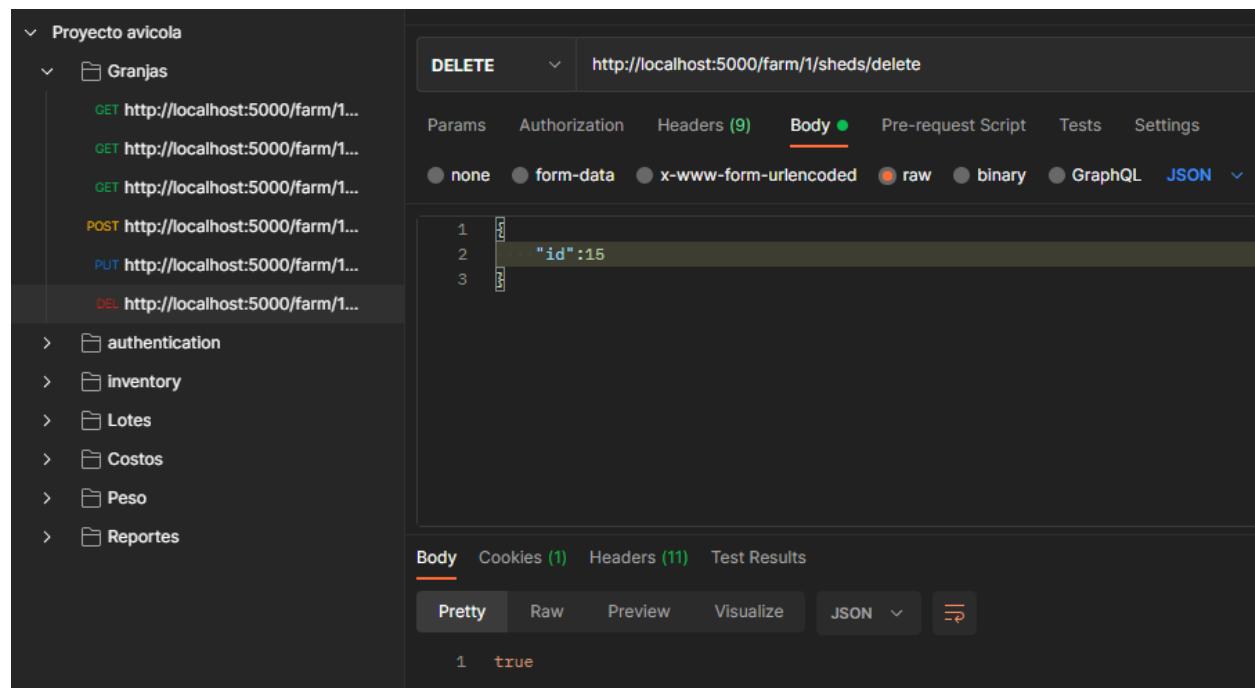
## Crear Galpon



## Actualizar Galpon



## Eliminar Galpon



- Tarea 3: CRUD lotes

## Obtener Lotes

Proyecto avicola

Granjias

authentication

inventory

Lotes

Costos

Peso

Reportes

GET http://localhost:5000/farm/1...

POST http://localhost:5000/farm/1...

PUT http://localhost:5000/farm/1...

DEL http://localhost:5000/farm/1...

GET http://localhost:5000/farm/1/3/lots

ParamsAuthorizationHeaders (7)BodyPre-request ScriptTestsSettings

Query Params

KEY	VALUE
Key	Value

BodyCookies (1)Headers (11)Test Results

PrettyRawPreviewVisualizeJSON

```
1  {
2    "id": 13,
3    "race": "HyLine Brown",
4    "amount_hens": 105,
5    "lotNumber": 4,
6    "dateOfBirth": "2022-06-14T05:00:00.000Z",
7    "id_shed": 3
8  },
9  {
10    "id": 15,
11    "race": "HyLine Brown",
12    "amount_hens": 351,
13    "lotNumber": 2,
14    "dateOfBirth": "2022-12-05T05:00:00.000Z",
15    "id_shed": 3
16  },
17  {
18    "id": 41,
19    "race": "HyLine Brown",
20    "amount_hens": 123,
21    "lotNumber": 6,
22    "dateOfBirth": "2022-12-07T05:00:00.000Z",
23    "id_shed": 3
24  }
25 }
```

## Crear Lote

Proyecto avicola

Granjias

authentication

inventory

Lotes

Costos

Peso

Reportes

GET http://localhost:5000/farm/1...

GET http://localhost:5000/farm/1...

GET http://localhost:5000/farm/1...

POST http://localhost:5000/farm/1...

PUT http://localhost:5000/farm/1...

DEL http://localhost:5000/farm/1...

POST http://localhost:5000/farm/1/15/lots/add

ParamsAuthorizationHeaders (9)BodyPre-request ScriptTestsSettings

noneform-datax-www-form-urlencodedrawbinaryGraphQLJSON

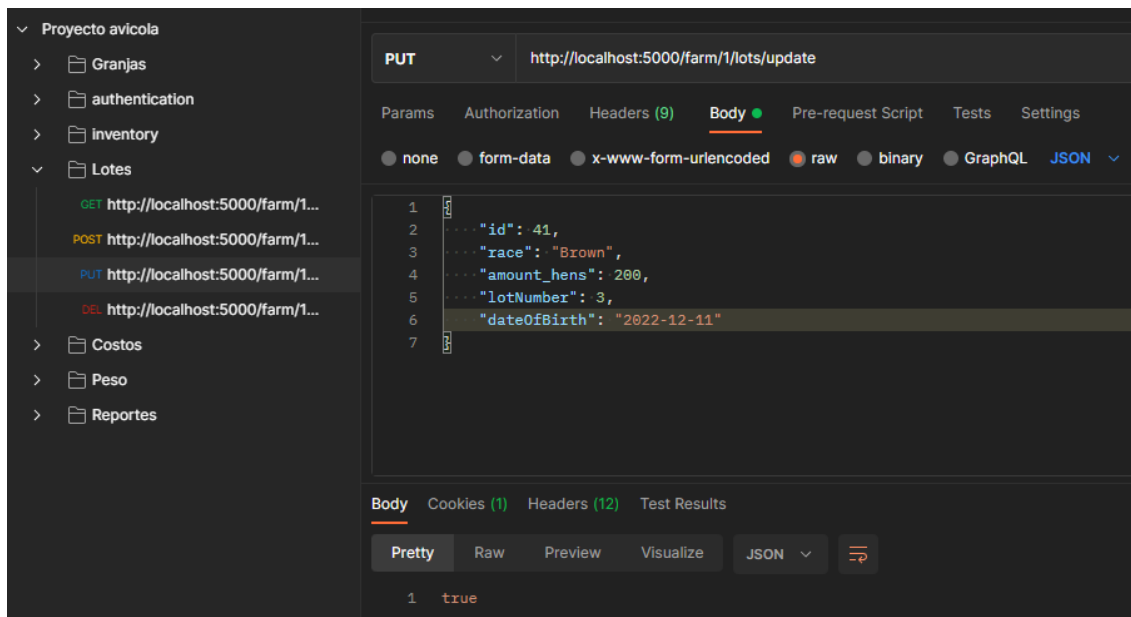
1 {
2 "race": "Brown",
3 "amount\_hens": 700,
4 "lotNumber": 7,
5 "dateOfBirth": "2022-12-5"
6 }

BodyCookies (1)Headers (11)Test Results

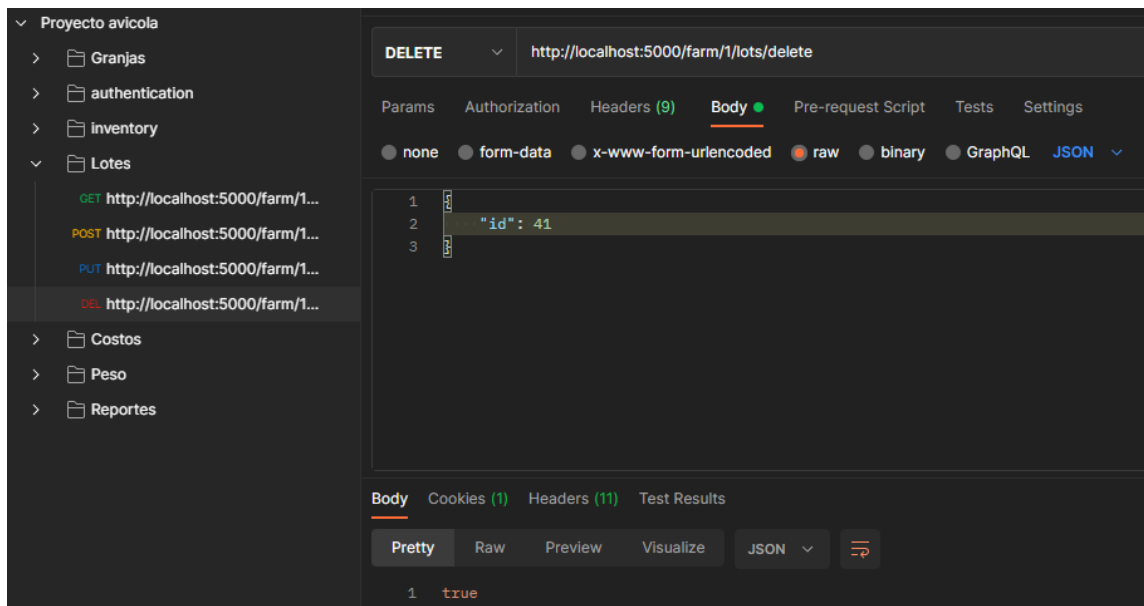
PrettyRawPreviewVisualizeJSON

```
1  true
```

## Actualizar Lote

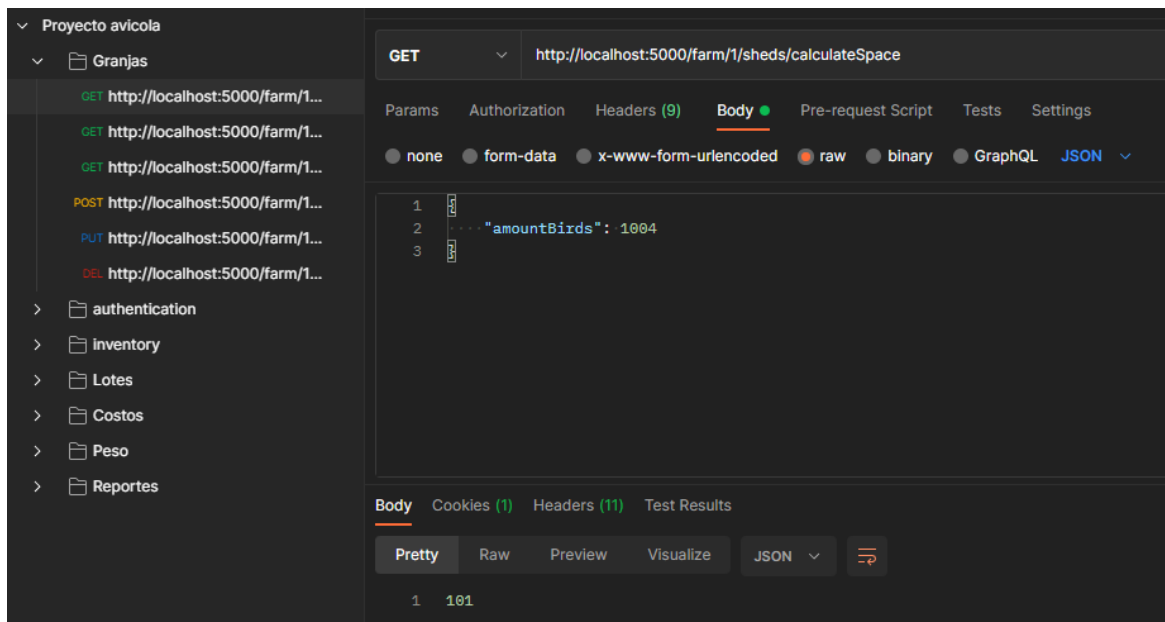


## Eliminar Lote

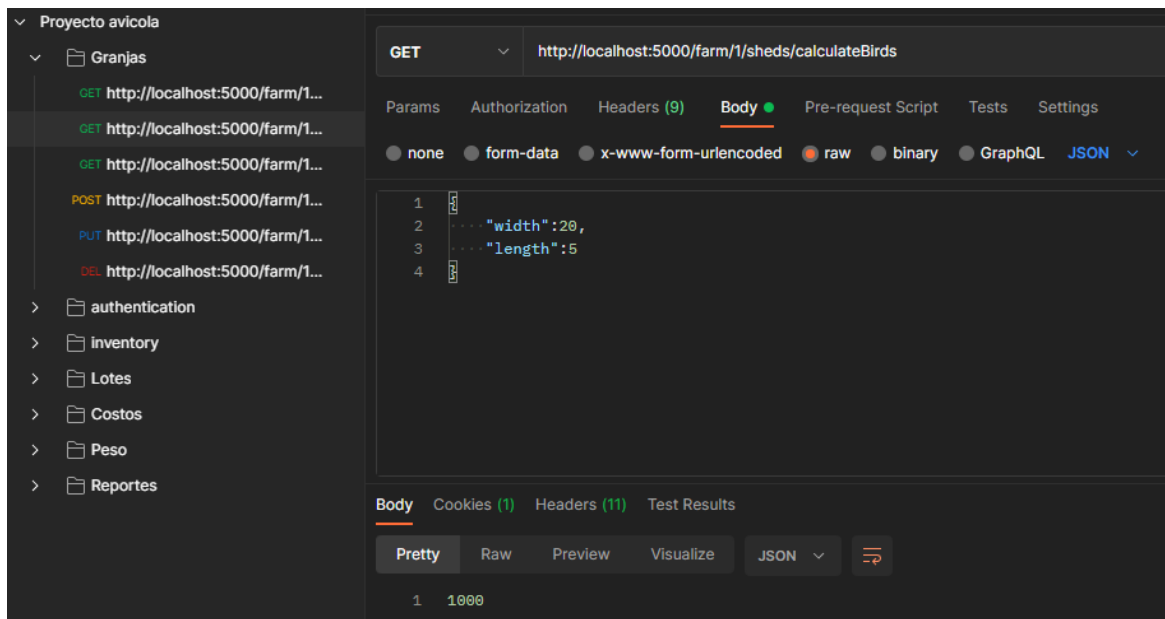


- Tarea 4: Calcular cantidad de aves para un determinado espacio y calcular el espacio necesario para una determinada cantidad de aves

## Espacio según cantidad de aves



## Cantidad de aves según espacio

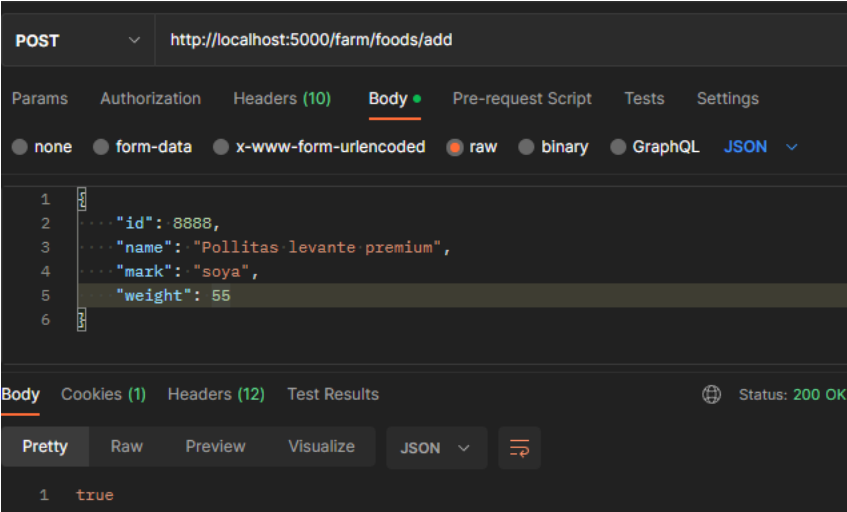


Sprint 3

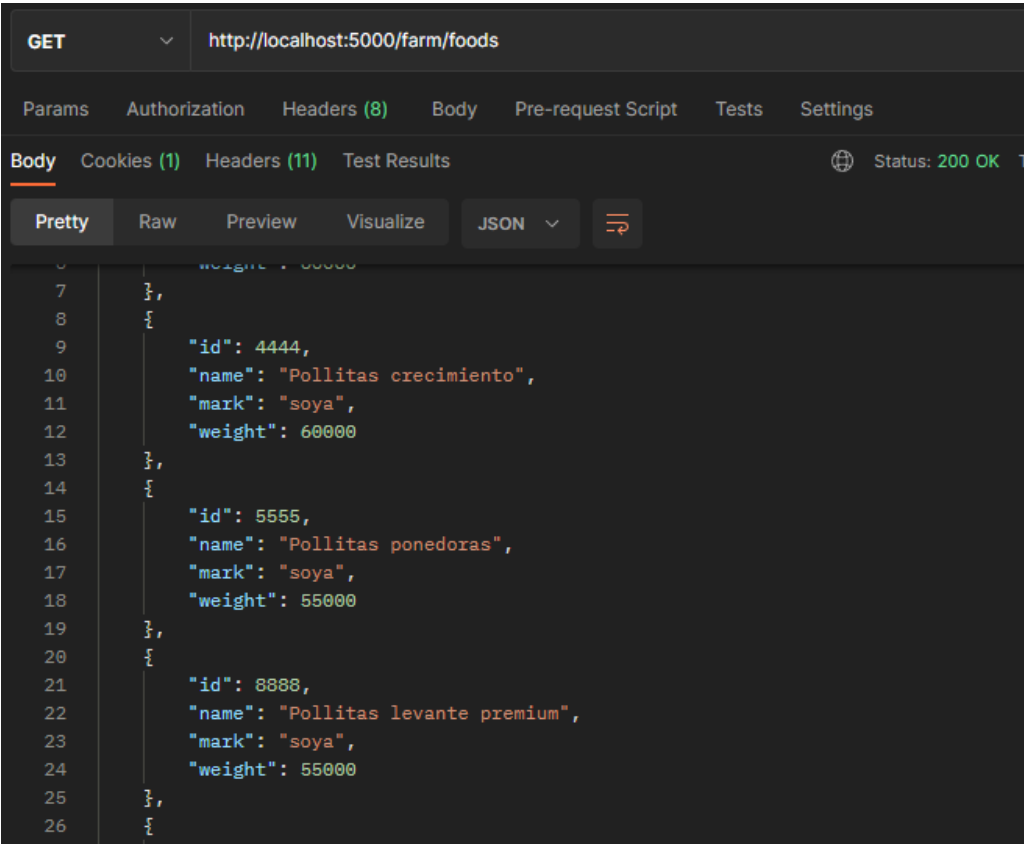
21/11/2022 - 28/11/2022

- Epic 4: Gestión Alimenticias de las aves:
  - Tarea 1: Inventario

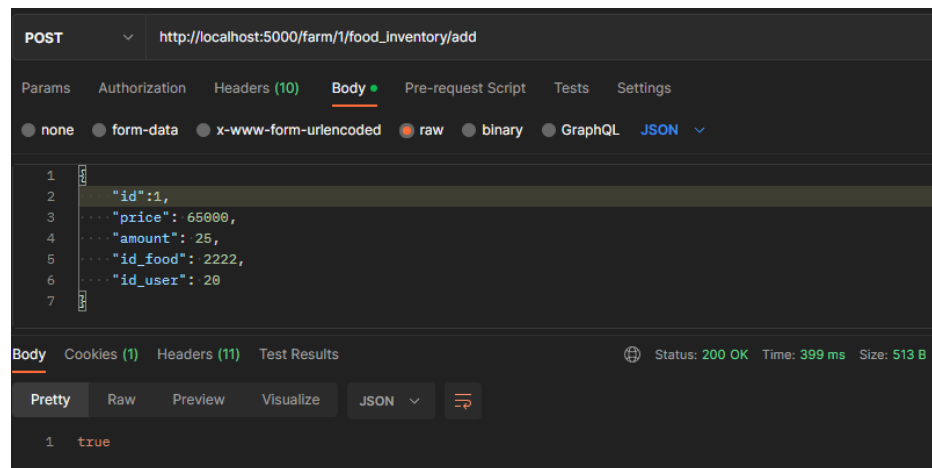
Añadir alimento



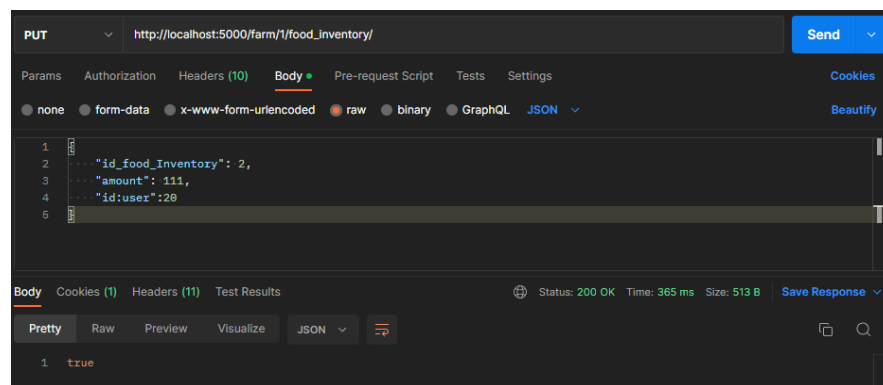
Obtener lista de alimentos



## Añadir alimento a inventario

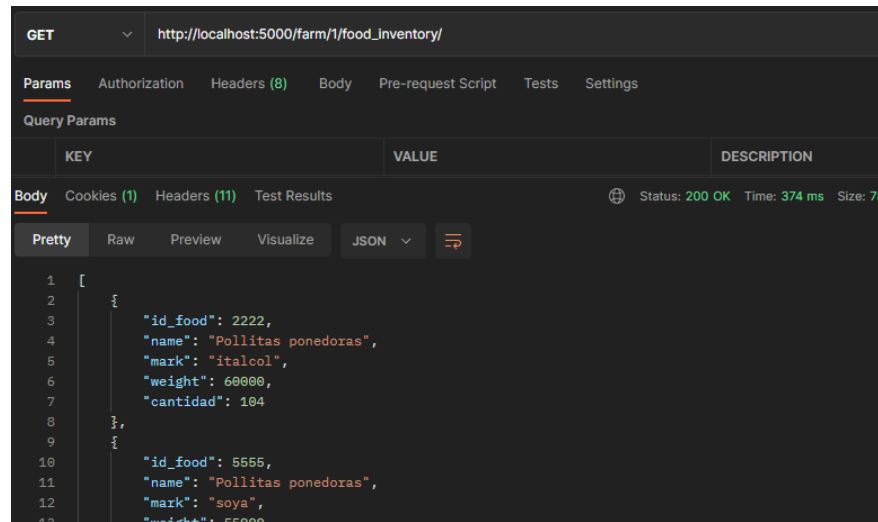


## Actualizar inventario



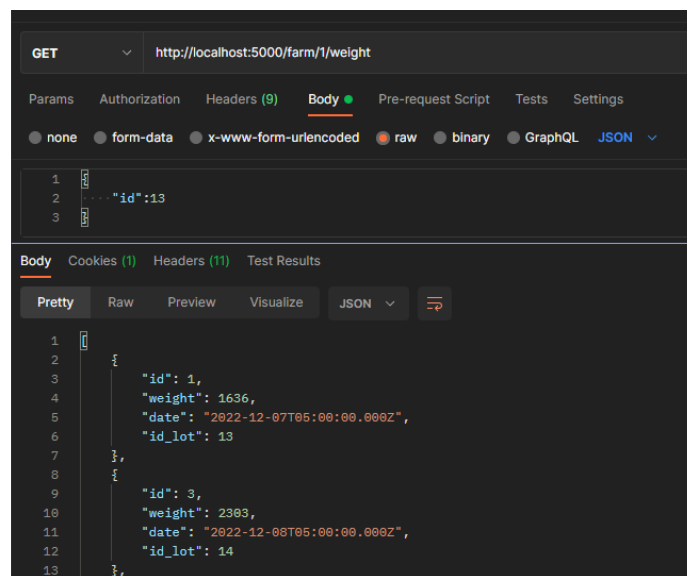
## Ver inventario



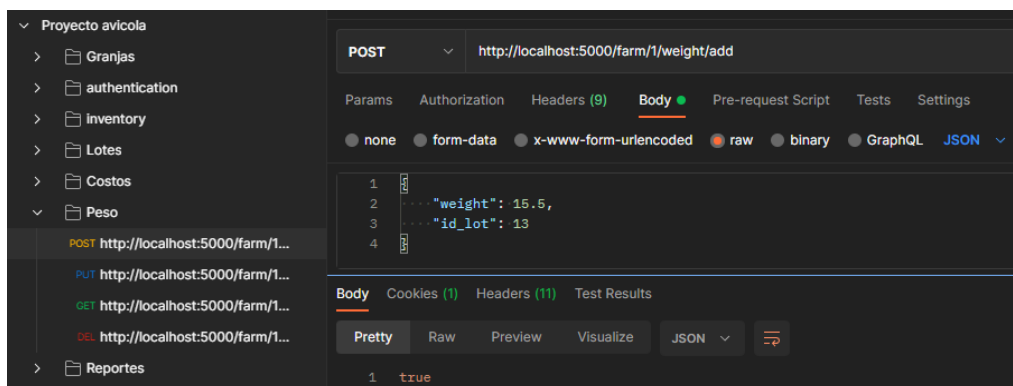


## ○ Tarea 2: Control de peso

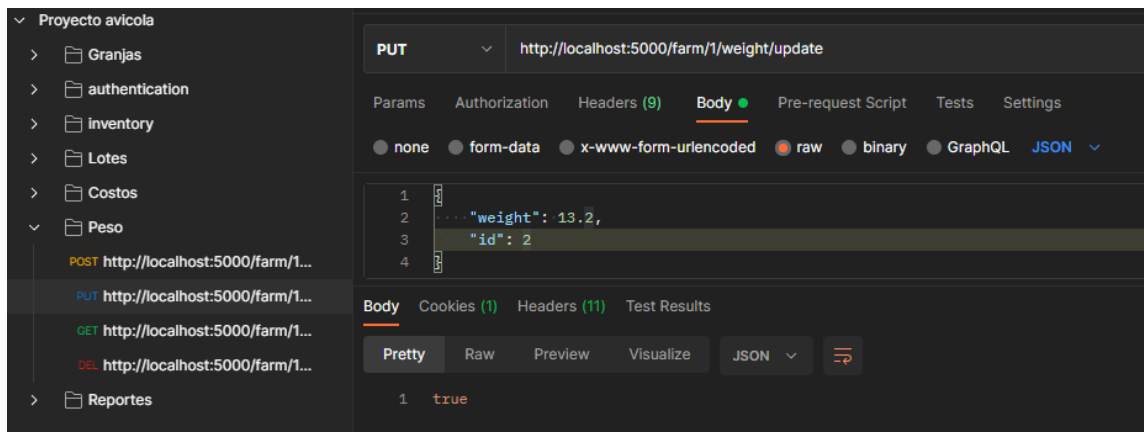
### Obtener control de peso



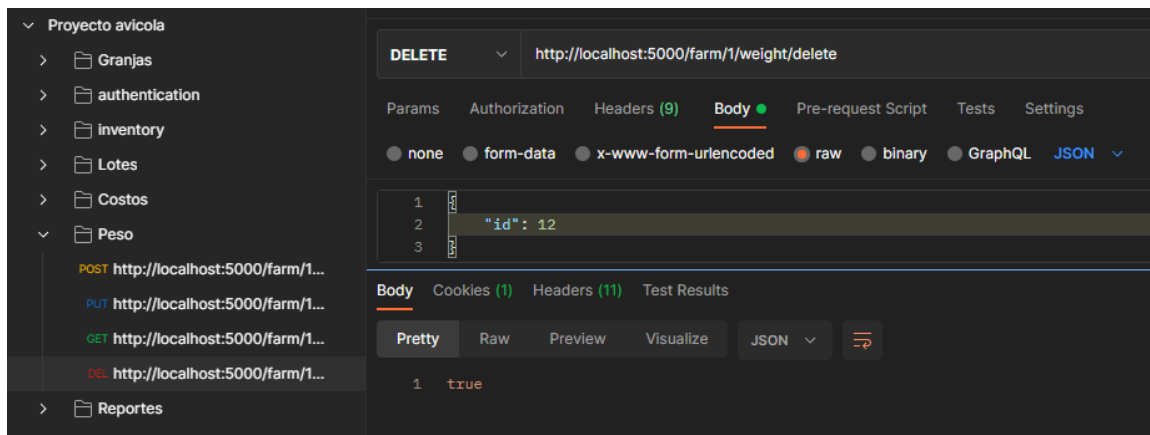
### Registrar peso



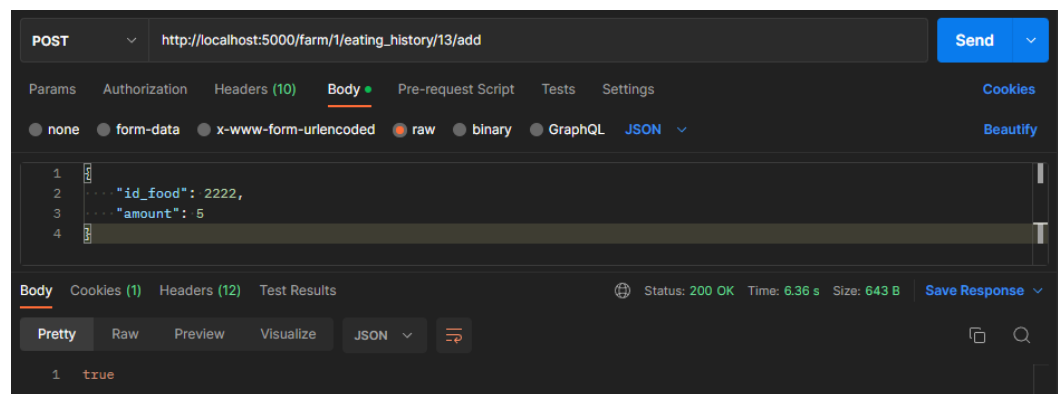
## Actualizar registro de peso



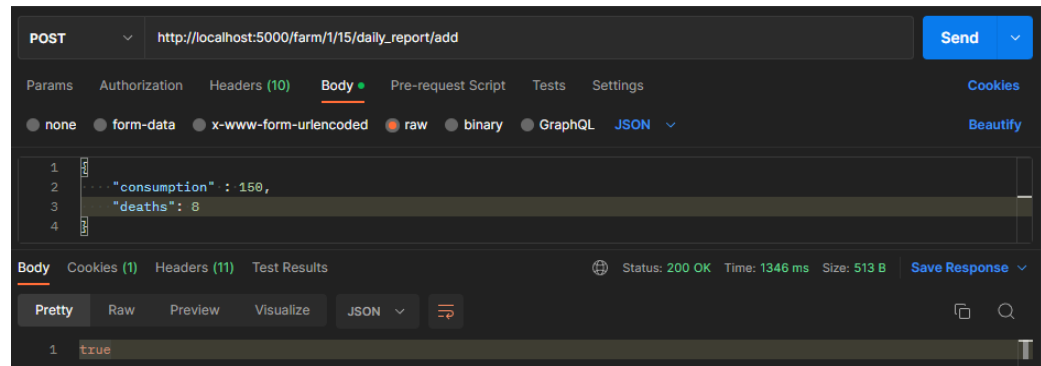
## Eliminar registro de peso



- Tarea 3: Control de alimentación diaria



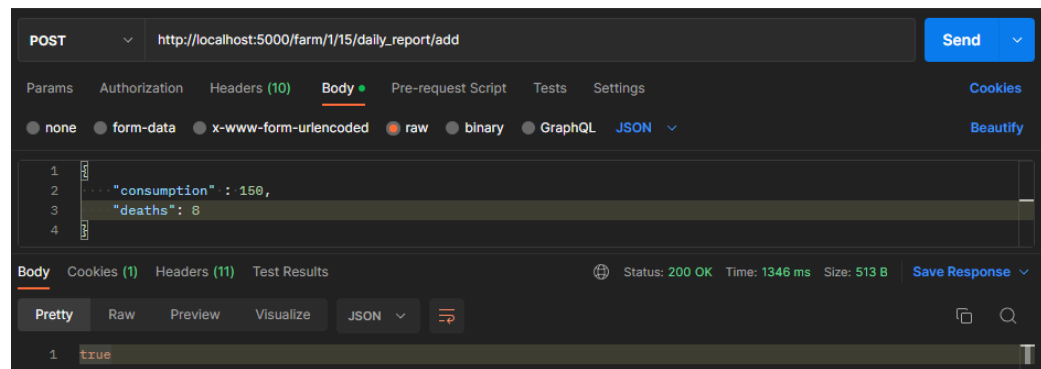
- Tarea 4: Control de consumo de agua



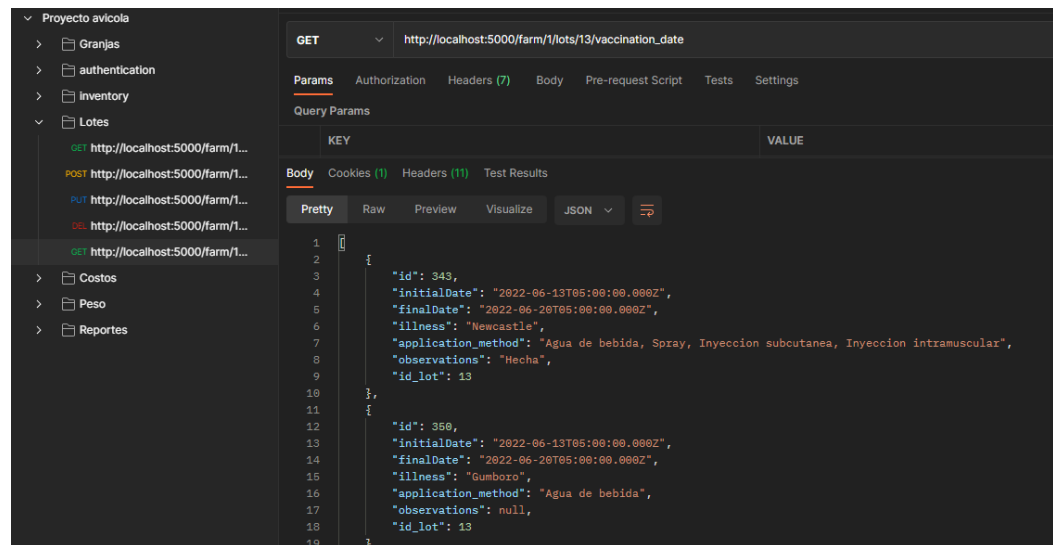
## Sprint 4

28/11/2022 - 05/12/2022

- Epic 5: Gestión de Trazabilidad
  - Tarea 1: Control de mortalidad diaria



- Tarea 2: Plan de vacunación



- Epic 6: Gestión administrativa
  - Tarea 1: Generación de reportes de consumo de alimentos

Obtener reporte diario consumo de alimentos

Proyecto avicola

Granjias

authentication

inventory

Lotes

Costos

Peso

Reportes

GET http://localhost:5000/farm/1...

GET http://localhost:5000/farm/1...

GET http://localhost:5000/farm/1...

GET http://localhost:5000/farm/1...

GEThttp://localhost:5000/farm/1/report/dailyFood

ParamsAuthorizationHeaders (9)Body●Pre-request ScriptTestsSettings

Query Params

KEY	VALUE
-----	-------

BodyCookies (1)Headers (11)Test Results

PrettyRawPreviewVisualizeJSON▼

```
1  {
2    {
3      "id_lot": 13,
4      "shedNumber": 1,
5      "lotNumber": 4,
6      "foodDate": "2022-12-09T22:06:42.000Z",
7      "alimento_consumido": 600000
8    },
9    {
10     "id_lot": 13,
11     "shedNumber": 1,
12     "lotNumber": 4,
13     "foodDate": "2022-12-09T22:06:43.000Z",
14     "alimento_consumido": 720000
15   },
16 }
```

Obtener reporte consumo total alimentos

Proyecto avicola

Granjias

authentication

inventory

Lotes

Costos

Peso

Reportes

GET http://localhost:5000/farm/1...

GET http://localhost:5000/farm/1...

GET http://localhost:5000/farm/1...

GET http://localhost:5000/farm/1...

GEThttp://localhost:5000/farm/1/report/foodAll

ParamsAuthorizationHeaders (9)Body●Pre-request ScriptTestsSettings

Query Params

KEY	VALUE
-----	-------

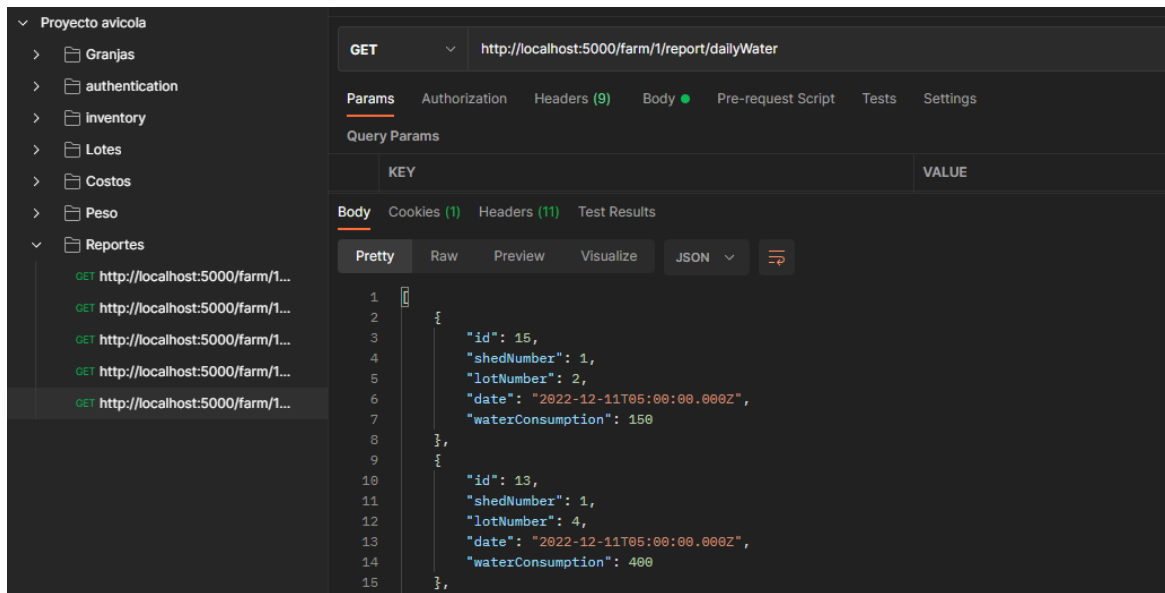
BodyCookies (1)Headers (11)Test Results

PrettyRawPreviewVisualizeJSON▼

```
1  {
2    {
3      "id_lot": 13,
4      "shedNumber": 1,
5      "lotNumber": 4,
6      "total_consumido": 3300000,
7      "promedio": 31428.5714
8    },
9    {
10     "id_lot": 14,
11     "shedNumber": 1,
12     "lotNumber": 1,
13     "total_consumido": 2890000,
14     "promedio": 23495.935
15   },
16 }
```

- Tarea 1: Generación de reportes de consumo de agua

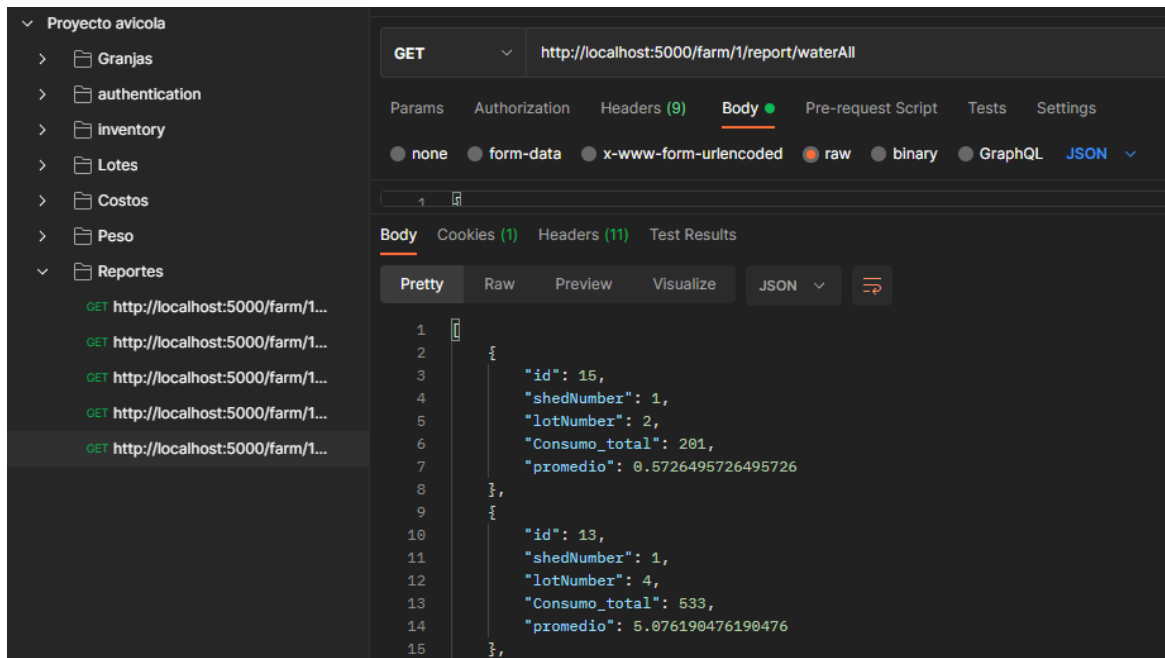
Obtener reporte diario consumo de agua



The screenshot shows a REST client interface with a sidebar on the left containing a project tree. The main panel displays a GET request to `http://localhost:5000/farm/1/report/dailyWater`. The response is shown in the 'Body' tab, formatted as JSON. The response is an array of two objects, each representing a farm's daily water consumption report.

```
1 {
2   {
3     "id": 15,
4     "shedNumber": 1,
5     "lotNumber": 2,
6     "date": "2022-12-11T05:00:00.000Z",
7     "waterConsumption": 150
8   },
9   {
10    "id": 13,
11    "shedNumber": 1,
12    "lotNumber": 4,
13    "date": "2022-12-11T05:00:00.000Z",
14    "waterConsumption": 400
15  },
16 }
```

Obtener reporte consumo total agua



The screenshot shows a REST client interface with a sidebar on the left containing a project tree. The main panel displays a GET request to `http://localhost:5000/farm/1/report/waterAll`. The response is shown in the 'Body' tab, formatted as JSON. The response is an array of two objects, each representing a farm's total water consumption report.

```
1 {
2   {
3     "id": 15,
4     "shedNumber": 1,
5     "lotNumber": 2,
6     "Consumo_total": 201,
7     "promedio": 0.5726495726495726
8   },
9   {
10    "id": 13,
11    "shedNumber": 1,
12    "lotNumber": 4,
13    "Consumo_total": 533,
14    "promedio": 5.076190476190476
15  },
16 }
```

- Tarea 3: Reporte de trazabilidad (mortalidad)

Obtener reporte diario mortalidad

GET <http://localhost:5000/farm/1/report/dailyDeath>

Params Authorization Headers (9) Body ● Pre-request Script Tests Settings

Query Params

KEY	VALUE
-----	-------

Body Cookies (1) Headers (11) Test Results

Pretty Raw Preview Visualize JSON

```
1 {
2   {
3     "id": 15,
4     "shedNumber": 1,
5     "lotNumber": 2,
6     "date": "2022-12-10T05:00:00.000Z",
7     "numberOfDeaths": 28
8   },
9   {
10    "id": 13,
11    "shedNumber": 1,
12    "lotNumber": 4,
13    "date": "2022-12-10T05:00:00.000Z",
14    "numberOfDeaths": 13
15  },
16 }
```

Obtener reporte muertes totales

GET <http://localhost:5000/farm/1/report/deathAll>

Params Authorization Headers (9) Body ● Pre-request Script Tests Settings

Query Params

KEY	VALUE
-----	-------

Body Cookies (1) Headers (11) Test Results

Pretty Raw Preview Visualize JSON

```
1 {
2   {
3     "id": 13,
4     "shedNumber": 1,
5     "lotNumber": 4,
6     "Muertes_totales": 89
7   },
8   {
9     "id": 14,
10    "shedNumber": 1,
11    "lotNumber": 1,
12    "Muertes_totales": 67
13  },
14 }
```

## Pruebas de Integración

Para las pruebas de integración se usa el software Telerik Test Studio para web tests, en donde se le indica a las pruebas que realicen ciertas acciones, para que luego automaticamente el software ejecute los pasos que se le indiquen

### Inicio de sesión

Pass - 12 passed out of total 12 executed.				Show step result details	View log	Clear results
1	✓	✓	✗	Navigate to : 'http://localhost:4200/'	▼	✎
2	✓	✓	✗	Click 'IniciarSesionButtonTag'	▼	✎
3	✓	✓	✗	Enter text 'rocio@gmail.com' in 'Text'	▼	✎
4	✓	✓	✗	Enter text 'Crispo219' in 'Password'	▼	✎
5	✓	✓	✗	Click 'Submit'	▼	✎
6	✓	✓	✗	Click 'ButtonTag'	▼	✎
7	✓	✓	✗	Click 'ButtonTag'	▼	✎
8	✓	✓	✗	Click 'Div'	▼	✎
9	✓	✓	✗	Click 'Div'	▼	✎
10	✓	✓	✗	Click 'UsernameiconImage'	▼	✎
11	✓	✓	✗	Click 'LogoutButtonButtonTag'	▼	✎
12	✓	✓	✗	Click 'CerrarSesiónButtonTag'	▼	✎

### Creación de Lotes

Pass - 20 passed out of total 20 executed.				Show step result details	View log	Clear results
8	✓	✓	✗	Click 'RocioGranjaSpan'	▼	✎
9	✓	✓	✗	Click 'GallinasPTag'	▼	✎
10	✓	✓	✗	Click 'GestiónDeSpan'	▼	✎
11	✓	✓	✗	Click 'NumeroDeloteTableHeader'	▼	✎
12	✓	✓	✗	Click 'FechaDeTableHeader'	▼	✎
13	✓	✓	✗	Click 'AñadirLotePTag'	▼	✎
14	✓	✓	✗	Enter text 'L-1' in 'x7Text'	▼	✎
15	✓	✓	✗	Enter text '12092022'	▼	✎
16	✓	✓	✗	Click 'x8Text'	▼	✎
17	✓	✓	✗	Select 'ByText' option 'G-1' on 'x9Select'	▼	✎
18	✓	✓	✗	Enter text '15' in 'x10Number'	▼	✎
19	✓	✓	✗	Enter text 'HyLine-brown' in 'x11Text'	▼	✎
20	✓	✓	✗	Click 'AñadirPTag'	▼	✎

Añadir +

### Creación de galpones

Pass - 17 passed out of total 17 executed.				Show step result details	View log	Clear results
1	✓		✎ Navigate to : 'http://localhost:4200/'	✓	✎	✎
2	✓		✎ Click 'IniciarSesionButtonTag'	✓	✎	✎
3	✓		✎ Click 'FormTag'	✓	✎	✎
4	✓		✎ Enter text 'rocio@gmail.com' in 'Text'	✓	✎	✎
5	✓		✎ Enter text 'Crispo219' in 'Password'	✓	✎	✎
6	✓		✎ Click 'Submit'	✓	✎	✎
7	✓		✎ Click 'Div'	✓	✎	✎
8	✓		✎ Click 'MisGranjasPTag'	✓	✎	✎
9	✓		✎ Click 'RocioGranjaSpan'	✓	✎	✎
10	✓		✎ Click 'GalponesPTag'	✓	✎	✎
11	✓		✎ Click 'GestiónSpan'	✓	✎	✎
12	✓		✎ Click 'Div'	✓	✎	✎
13	✓		✎ Click 'AñadirGalpónPTag'	✓	✎	✎
14	✓		✎ Enter text 'G-1' in 'x7Text'	✓	✎	✎
15	✓		✎ Enter text '10' in 'x11Number'	✓	✎	✎
16	✓		✎ Enter text '10' in 'x12Number'	✓	✎	✎
> 17	✓		✎ Click 'AñadirPTag'	✓	✎	✎

## Mortalidad

Pass - 19 passed out of total 19 executed.				Show step result details	View log	Clear results
1	✓		✎ Navigate to : 'http://localhost:4200/'	✓	✎	✎
2	✓		✎ Click 'IniciarSesionButtonTag'	✓	✎	✎
3	✓		✎ Enter text 'rocio@gmail.com' in 'Text'	✓	✎	✎
4	✓		✎ Enter text 'Crispo219' in 'Password'	✓	✎	✎
5	✓		✎ Click 'Submit'	✓	✎	✎
6	✓		✎ Click 'Div'	✓	✎	✎
7	✓		✎ Click 'Logolmage'	✓	✎	✎
8	✓		✎ Click 'RocioGranjaSpan'	✓	✎	✎
9	✓		✎ Click 'Logolmage'	✓	✎	✎
10	✓		✎ Click 'MortalidadSpan'	✓	✎	✎
11	✓		✎ Click 'Div'	✓	✎	✎
12	✓		✎ Click 'LotesTableHeader'	✓	✎	✎
13	✓		✎ Enter text 'L-1' in 'SearchText'	✓	✎	✎
14	✓		✎ Click 'L1TableCell'	✓	✎	✎
15	✓		✎ Desktop command: Drag & Drop Div to Window Target	✓	✎	✎
16	✓		✎ Click 'GuardarButtonTag'	✓	✎	✎
17	✓		✎ Click 'UsernameonImage'	✓	✎	✎
18	✓		✎ Click 'LogoutButtonButtonTag'	✓	✎	✎
> 19	✓		✎ Click 'CerrarSesiónButtonTag'	✓	✎	✎



## Conclusiones

1. Se logró cumplir con los objetivos evidenciando que el uso de un marco de trabajo y metodologías ágiles como scrum y kanban son herramientas poderosas para llevar a cabo el desarrollo de proyectos en poco tiempo. Además de la importancia de un buen levantamiento de requerimientos y gestión del proyecto que permita distribuir eficientemente la carga de trabajo y los recursos necesarios para lograr cumplir con los objetivos planteados.
2. Durante el desarrollo del proyecto se pudo evidenciar la importancia de llevar soluciones tecnológicas al sector agropecuario, en este caso, esta herramienta será de gran ayuda para los granjeros de la región. Se espera que con ello logren mejorar su productividad y mantener un control más exacto de las actividades avícolas.
3. Se comprobó la importancia de una correcta planificación, estructura, y excelente trabajo en equipo para lograr cumplir satisfactoriamente los objetivos. En el proceso de desarrollo, la parte que más se dificultó fue la unión de la vista con la lógica del negocio. Por lo que fue necesaria una buena comunicación, paciencia y enfoque de parte de todos los miembros del equipo para lograr resolver los problemas que implicó este reto.

## Referencias

- [1] (S/f). Lohmann-breeders.com. Recuperado de [https://lohmann-breeders.com/media/2021/06/LB\\_MG\\_LB-Classic\\_ESP.pdf](https://lohmann-breeders.com/media/2021/06/LB_MG_LB-Classic_ESP.pdf)
- [2] Db4free.net - MySQL database for free. (s/f). Db4free.net. Recuperado de <https://www.db4free.net/>
- [3] Inicio - Agroshow. (s/f). Agroshow.info. Recuperado de <https://agroshow.info/productos/avicultura/software/>
- [4] Las disposiciones del ICA son claves para poder iniciar una granja avícola. (s/f). Agronegocios.co. Recuperado de <https://www.agronegocios.co/aprenda/las-disposiciones-del-ica-son-claves-para-poder-iniciar-un-a-granja-avicola-2923712>

## **Anexos:**

### **Anexo 1: Entrevista primer encuentro:**

1. ¿Cuál es la actividad económica de su negocio?
  - Nuestro negocio consiste en la crianza de gallinas ponedoras desde el primer día de nacidas hasta cuando cumplen aproximadamente 17 semanas de vida, en este tiempo nos encargamos del levante de las aves, esto consiste en alimentarlas, administrar el espacio necesario para su cuidado y la vacunación. Después de estas semanas se venden o se llevan a otras granjas para la producción de huevo.
2. ¿Cómo llevan el control del espacio para las gallinas?
  - Para ello hay que tener en cuenta la cantidad de gallinas, la raza y el espacio disponible, por ejemplo, la raza Brown que es la más usada en esta región, requiere un espacio de 1 metro cuadrado por cada 10 gallinas. Entonces lo que hay que hacer primero es calcular cuántas gallinas puedo tener en el espacio disponible. Una vez hecho esto se alistan los galpones y se adecuan para las nuevas pollitas, a pesar que al principio no se vaya a usar todo el espacio, es necesario dejarlo reservado ya que se le va ampliando a medida que van creciendo.
3. ¿Cómo manejan los grupos de gallinas que sean de diferente edad?
  - Cada grupo de gallinas es llamado lote, y a cada lote se le asigna un espacio específico y delimitado en un galpón, no se pueden revolver gallinas de diferentes lotes. En un galpón puede haber varios lotes de gallinas y estos lotes se enumeran y se lleva su control en un cuadernillo.
4. ¿Qué tipo de información maneja el cuadernillo?
  - En el cuadernillo va el registro semanal de cada lote, este registro incluye: el peso promedio de las aves, el consumo de agua, la cantidad de comida que se les está dando y se calcula la cantidad de comida que se les debe dar la siguiente semana de acuerdo a su edad y peso, además se lleva un registro de las aves muertas. El cuadernillo también incluye las fechas de vacunación y despique de las aves. Todo esto con el objetivo de llevar un control de costos por Lotes.
5. ¿Cómo calculan el peso promedio de las aves y el alimento que se les debe dar?
  - Se toma una muestra significativa de la población de aves de un lote, por ejemplo, para un lote de 2000 aves, se toma el peso de aproximadamente 350 aves, luego se promedian y se registra el peso en el cuadernillo. De acuerdo a la edad y con ayuda de un cuadro de pesos que contienen el rango en el que las aves deben estar, se toma la decisión de cuánto alimento se les debe dar la siguiente semana.
6. ¿Cómo calculan el consumo de agua?

- El agua se distribuye a través de un sistema de dispensación automática, a cada lote se le asigna un tanque de agua independiente y se calcula de acuerdo al nivel del agua en el tanque.

7. ¿Cómo se realiza el proceso de vacunación?

- Se realiza un plan de vacunación en el cual se definen la semana en que debe realizarse y la vacuna a aplicar, se realiza el pedido a los proveedores de acuerdo a la cantidad de aves que se tengan. Se contrata personal para desarrollar este proceso.

8. ¿Cómo se lleva el inventario de alimentos?

- Se piden al proveedor y se almacenan en una bodega, Se pide de acuerdo a la cantidad de alimento que se debe proporcionar a cada lote semanal o quincenalmente.

9. ¿Se ha implementado anteriormente un sistema parecido, si – no, cuál?

- Realmente no, pero sería muy útil tener un sistema que me ayude a manejar mi negocio debido a que he conocido diferentes casos en donde he visto que un sistema como este le da un plus o un valor adicional al negocio.

10. ¿Qué plataformas se utilizan para estas aplicaciones?

- Hemos investigado, y existen muy pocas opciones al respecto, además son aplicaciones que toca descargar en un ordenador por lo que para administrar la granja tendría que tener acceso a ese único dispositivo. La idea de una aplicación web, dónde pueda acceder desde cualquier lugar y dispositivo es algo muy atractivo.

11. ¿Algo adicional que debamos saber?

- Si, algo importante que olvidé mencionar, es que en el cuadernillo hay cosas que pueden escribir directamente los granjeros y otras que sólo las podemos hacer los administradores o dueños de la granja, por ejemplo, el control de mortalidad y el manejo del inventario de alimento llevan siempre nuestras firmas.

## **Anexo 2: Documento IEEE 830**

### **Documento IEEE 830 para un Sistema de información Web para la gestión y trazabilidad de granjas avícolas**

Ficha del documento

<b>Fecha</b>	<b>Revisión</b>	<b>Autor</b>	<b>Verificado dep. Calidad.</b>
05/11/2022		Nicolás David Fajardo Acuña	
05/11/2022		Cristian David Parada Martínez	
05/11/2022		Daniel Felipe Suarez Bohórquez	

#### ***1. Introducción***

Este documento es una Especificación de Requisitos Software (ERS) para un sistema de información Web para la gestión y trazabilidad de granjas avícolas. Esta especificación se ha estructurado basándose en las directrices dadas por el estándar IEEE Práctica Recomendada para Especificaciones de Requisitos Software ANSI/IEEE 830, 1998.

#### ***2. Propósito***

El presente documento tiene como propósito definir las especificaciones funcionales, no funcionales para el desarrollo de un sistema de información web que permitirá sistematizar la gestión de una granja avícola. Este sistema será utilizado por el administrador y los cajeros de la tienda.

#### ***3. Alcance***

Esta especificación de requisitos está dirigida a los usuarios del sistema (administrador y granjeros), para continuar con el desarrollo de aplicaciones de este tipo sobre las microempresas y para profundizar en la automatización de ésta, la cual tiene por objetivo principal el gestionar los distintos procesos administrativos (Inventario, control de costos, gestión de espacios, trazabilidad y reportes) de una granja avícola, aunque el sistema podría ampliarse hasta la producción de huevos, el alcance del desarrollo de éste proyecto se delimitará a la gestión de la granja hasta cuando las aves tengan 17 semanas de vida que es el punto en que son levantadas e inician la postura, tema el cuál podría incluirse en siguientes versiones.

#### 4. Personal involucrado

<b>Nombre</b>	Nicolás David Fajardo Acuña
<b>Rol</b>	Analista y programador
<b>Categoría Profesional</b>	Ingeniero de sistemas y computación.
<b>Responsabilidad</b>	Análisis de información y programación.
<b>Información de contacto</b>	nicolas.fajardo01@uptc.edu.co

<b>Nombre</b>	Cristian David Parada Martínez
<b>Rol</b>	Diseñador y programador
<b>Categoría Profesional</b>	Ingeniero de sistemas y computación.
<b>Responsabilidad</b>	Diseño y programación.
<b>Información de contacto</b>	Cristian.parada@uptc.edu.co

<b>Nombre</b>	Daniel Felipe Suarez Bohórquez
<b>Rol</b>	Analista y programador
<b>Categoría Profesional</b>	Ingeniero de sistemas y computación.
<b>Responsabilidad</b>	Análisis de información y programación.
<b>Información de contacto</b>	Daniel.suarez09@uptc.edu.co

#### 5. Definiciones, acrónimos y abreviaturas

<b>Nombre</b>	<b>Descripción</b>
<b>Usuario</b>	Persona que usará el sistema para gestionar procesos
<b>SIS-I</b>	Sistema de Información para la Gestión de Procesos de la granja avícola.
<b>ERS</b>	Especificación de Requisitos Software
<b>RF</b>	Requerimiento Funcional
<b>RNF</b>	Requerimiento No Funcional
<b>FTP</b>	Protocolo de Transferencia de Archivos

#### 6. Referencias

<b>Título del Documento</b>	<b>Referencia</b>
Standard IEEE 830 - 1998	IEEE

## ***7. Resumen***

Este documento consta de tres partes. En la primera parte se realiza una introducción al mismo y se proporciona una visión general de la especificación de recursos del sistema.

En la segunda parte del documento se realiza una descripción general del sistema, con el fin de conocer las principales funciones que éste debe realizar, los datos asociados, los factores, restricciones, supuestos y dependencias que afectan al desarrollo, sin entrar en excesivos detalles.

Por último, la tercera parte del documento es aquella en la que se definen detalladamente los requisitos que debe satisfacer el sistema.

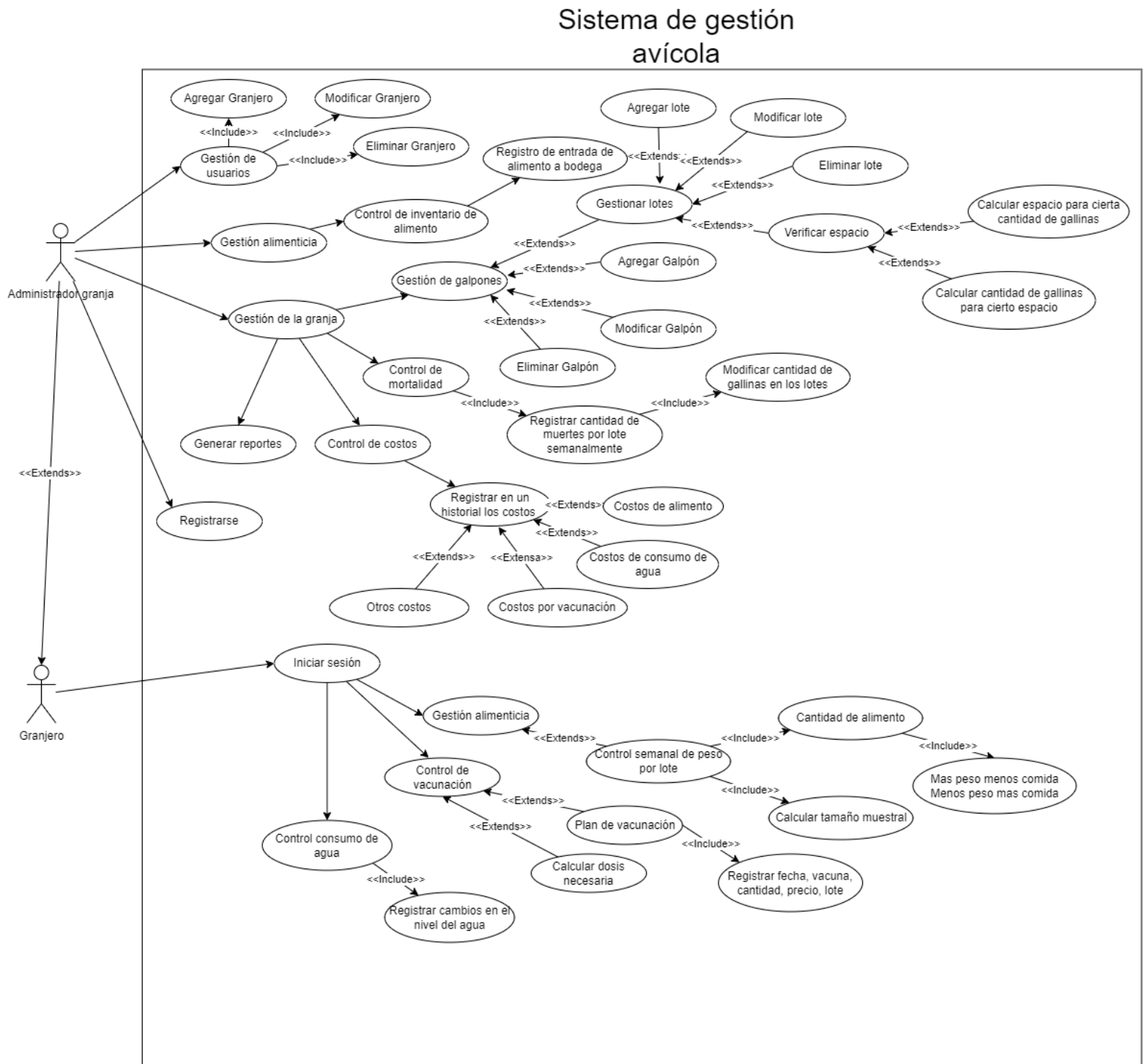
## ***8. Descripción general***

SIGEAVI será un software avícola para gallinas ponedoras multi-granja diseñado para gestionar los registros de levante, trazabilidad y de rendimiento de esta. Será una poderosa herramienta ofimática con la que podrá contar el granjero y los administradores en la toma de decisiones, para observar la evolución de la granja y así advertir de si está dentro de los parámetros de producción rentables. El sistema maneja la información en reportes permitiendo así compararlas con las tablas de rendimientos de los estándares del mercado.

## ***9. Perspectiva del producto***

El sistema SIS-I será un producto diseñado para trabajar en entornos web, lo que permitirá que tenga un mejor manejo ya que su usabilidad será más rápida y eficaz, para los usuarios encargados de manejar el sistema, además permitirá a los usuarios del sistema tener acceso desde cualquier dispositivo sin necesidad de instalar alguna aplicación.

## 10. Funcionalidad del producto





### ***11. Características de los usuarios***

<b>Tipo de usuario</b>	Administrador
<b>Formación</b>	Básica secundaria
<b>Actividades</b>	Control y manejo del sistema en general, gestión de lotes, registró de mortalidad, control de inventario de alimentos, gestión de costos, generación de reportes.

<b>Tipo de usuario</b>	Granjero
<b>Formación</b>	Básica secundaria
<b>Actividades</b>	Registro de consumo de agua y alimentos, registro de peso.

### ***12. Restricciones***

- Lenguajes y tecnologías en uso: JavaScript, NodeJs, Angular.
- Interfaz para ser usada en la web.
- El sistema se diseñará según una arquitectura en capas .
- Los servidores deben ser capaces de atender consultas concurrentemente.
- El sistema deberá tener un diseño e implementación sencilla, independiente de la plataforma o del lenguaje de programación.

### ***13. Suposiciones y dependencias***

- Se asume que los requisitos aquí descritos son estables
- Los equipos en los que se vaya a ejecutar el sistema deben cumplir los requisitos antes indicados para garantizar una ejecución correcta del mismo

#### 14. Requisitos específicos

- Requerimientos Funcionales

<b>Identificación del requerimiento:</b>	RF01
<b>Nombre del Requerimiento:</b>	Validación de Usuario.
<b>Características:</b>	Los usuarios deberán identificarse para acceder a cualquier parte del sistema.
<b>Descripción del requerimiento:</b>	El sistema podrá ser consultado por cualquier usuario dependiendo del módulo en el cual se encuentre y su nivel de permisos.
<b>Requerimiento NO funcional:</b>	<ul style="list-style-type: none"><li>● RNF01</li><li>● RNF02</li><li>● RNF06</li><li>● RNF07</li><li>● RNF08</li></ul>
<b>Prioridad del requerimiento:</b> Medio	

<b>Identificación del requerimiento:</b>	RF02
<b>Nombre del Requerimiento:</b>	Recuperación de contraseña
<b>Características:</b>	El sistema permitirá recuperar la contraseña de un usuario
<b>Descripción del requerimiento:</b>	El sistema permitirá a los usuarios recuperar su contraseña haciendo uso de su correo electrónico en donde se le enviará un código y éste tendrá que confirmarlo para cambiar su contraseña
<b>Requerimiento NO funcional:</b>	<ul style="list-style-type: none"><li>● RNF01</li><li>● RNF04</li><li>● RNF05</li><li>● RNF06</li></ul>
<b>Prioridad del requerimiento:</b> Baja	

<b>Identificación del requerimiento:</b>	RF03
<b>Nombre del Requerimiento:</b>	Gestión de Galpones
<b>Características:</b>	El sistema permitirá gestionar los galpones en los que se criarán las aves
<b>Descripción del requerimiento:</b>	El sistema permitirá al administrador añadir, modificar o eliminar los galpones de la granja, cada galpón está identificado por un número, y tiene dimensiones de largo y ancho.
<b>Requerimiento NO funcional:</b>	<ul style="list-style-type: none"> <li>● RNF01</li> <li>● RNF04</li> <li>● RNF05</li> <li>● RNF08</li> </ul>
<b>Prioridad del requerimiento:</b> Alta	

<b>Identificación del requerimiento:</b>	RF04
<b>Nombre del Requerimiento:</b>	Gestión de Lotes
<b>Características:</b>	El sistema permitirá gestionar los lotes en los que se criarán las aves
<b>Descripción del requerimiento:</b>	El sistema permitirá al administrador añadir, modificar o eliminar los lotes, cada lote está identificado por un número, y tiene información acerca de, la fecha de nacimiento de las aves, la cantidad de aves, el galpón asignado y la raza de las aves.
<b>Requerimiento NO funcional:</b>	<ul style="list-style-type: none"> <li>● RNF01</li> <li>● RNF02</li> <li>● RNF03</li> <li>● RNF05</li> </ul>
<b>Prioridad del requerimiento:</b> Alta	

<b>Identificación del requerimiento:</b>	RF05
<b>Nombre del Requerimiento:</b>	Control de consumo de agua
<b>Características:</b>	El sistema permitirá registrar el consumo de agua de las aves por lotes
<b>Descripción del requerimiento:</b>	El sistema permitirá al granjero registrar la cantidad de agua en litros que las aves consumen diariamente de acuerdo a los cambios del nivel de agua en los tanques.
<b>Requerimiento NO funcional:</b>	<ul style="list-style-type: none"> <li>• RNF01</li> <li>• RNF02</li> <li>• RNF05</li> <li>• RNF06</li> </ul>
<b>Prioridad del requerimiento:</b> Baja	

<b>Identificación del requerimiento:</b>	RF06
<b>Nombre del Requerimiento:</b>	Control de mortalidad
<b>Características:</b>	El sistema permitirá llevar un registro de las aves muertas.
<b>Descripción del requerimiento:</b>	El sistema permitirá al administrador registrar la cantidad de aves que hayan muerto en un determinado día, automáticamente se debe actualizar la cantidad de aves en el lote.
<b>Requerimiento NO funcional:</b>	<ul style="list-style-type: none"> <li>• RNF01</li> <li>• RNF04</li> <li>• RNF05</li> <li>• RNF06</li> </ul>
<b>Prioridad del requerimiento:</b> Alta	

<b>Identificación del requerimiento:</b>	RF07
<b>Nombre del Requerimiento:</b>	Control de peso
<b>Características:</b>	El sistema permitirá llevar un registro semanal del peso de las aves.
<b>Descripción del requerimiento:</b>	El sistema permitirá al granjero registrar el peso promedio de las aves en cada lote, para esto, el sistema sugerirá cuantas aves se deben pesar de acuerdo a la cantidad de aves en el lote. De acuerdo al peso obtenido y a la edad de las aves, el sistema sugerirá cuanto alimento se les debe proporcionar la siguiente semana.
<b>Requerimiento NO funcional:</b>	<ul style="list-style-type: none"> <li>• RNF01</li> <li>• RNF02</li> <li>• RNF03</li> <li>• RNF07</li> </ul>
<b>Prioridad del requerimiento:</b> Media	

<b>Identificación del requerimiento:</b>	RF08
<b>Nombre del Requerimiento:</b>	Inventario de alimentos
<b>Características:</b>	El sistema permitirá gestionar el inventario de alimentos.
<b>Descripción del requerimiento:</b>	El sistema permitirá al <b>administrador</b> gestionar el inventario de alimentos registrando entradas (compras) y salidas de alimentos, editando, eliminando y añadiendo nuevos alimentos. Cada alimento tiene un precio, una medida, nombre y marca. Cada lote tiene su propio inventario de alimentos. Al <b>granjero</b> , sólo se le permitirá registrar la salida de alimentos.
<b>Requerimiento NO funcional:</b>	<ul style="list-style-type: none"> <li>• RNF01</li> <li>• RNF02</li> <li>• RNF03</li> <li>• RNF07</li> </ul>
<b>Prioridad del requerimiento:</b> Media	

<b>Identificación del requerimiento:</b>	RF09
<b>Nombre del Requerimiento:</b>	Plan de vacunación
<b>Características:</b>	El sistema permitirá gestionar el plan de vacunación.
<b>Descripción del requerimiento:</b>	El sistema permitirá al granjero gestionar el plan de vacunación de las aves, para ello calcula la dosis necesaria para el lote. El plan de vacunación incluye la fecha, el nombre de la vacuna a aplicar, la cantidad necesaria para el lote, el precio y el lote al que se le va a aplicar.
<b>Requerimiento NO funcional:</b>	<ul style="list-style-type: none"> <li>● RNF01</li> <li>● RNF03</li> <li>● RNF04</li> <li>● RNF08</li> </ul>
<b>Prioridad del requerimiento:</b> Media	

<b>Identificación del requerimiento:</b>	RF10
<b>Nombre del Requerimiento:</b>	Control de costos
<b>Características:</b>	El sistema permitirá llevar un control de costos por lotes.
<b>Descripción del requerimiento:</b>	El sistema permitirá al administrador gestionar los costos por lote este incluye costos de alimentación, consumo de agua, adecuación, vacunación, tratamientos y otros costos que puedan surgir durante el periodo de crianza de las aves.
<b>Requerimiento NO funcional:</b>	<ul style="list-style-type: none"> <li>● RNF01</li> <li>● RNF03</li> <li>● RNF07</li> <li>● RNF08</li> </ul>
<b>Prioridad del requerimiento:</b> Bajo	

<b>Identificación del requerimiento:</b>	RF11
<b>Nombre del Requerimiento:</b>	Generación de reportes.
<b>Características:</b>	El sistema permitirá crear reportes semanales.
<b>Descripción del requerimiento:</b>	El sistema permitirá al administrador crear reportes de alimento consumido, mortalidad y peso corporal a través del tiempo.
<b>Requerimiento NO funcional:</b>	<ul style="list-style-type: none"> <li>• RNF01</li> <li>• RNF02</li> <li>• RNF05</li> <li>• RNF06</li> </ul>
<b>Prioridad del requerimiento:</b> Bajo	

- Requerimientos No Funcionales

<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>	El sistema debe tener una interfaz de uso intuitiva y sencilla.
<b>Prioridad del requerimiento:</b> Medio	

<b>Identificación del requerimiento:</b>	RNF02
<b>Nombre del Requerimiento:</b>	Ayuda en el uso del sistema.
<b>Características:</b>	La interfaz del usuario deberá de presentar un sistema de ayuda para que los mismos usuarios del sistema se les facilite el trabajo en cuanto al manejo del sistema.
<b>Descripción del requerimiento:</b>	La interfaz debe estar complementada con un buen sistema de ayuda (la administración puede recaer en personal con poca experiencia en el uso de aplicaciones informáticas).
<b>Prioridad del requerimiento:</b> Bajo	

<b>Identificación del requerimiento:</b>	RNF03
<b>Nombre del Requerimiento:</b>	Encabezado del sistema
<b>Características:</b>	El sistema deberá mostrar en todos sus encabezados la información del sistema. Además información del usuario que ha iniciado sesión, y de la granja.
<b>Descripción del requerimiento:</b>	El sistema deberá mostrar la información del software como su nombre, logotipo y eslogan. La información de la granja como nombre y ubicación. y la información del usuario como nombre, rol y última hora de conexión.
<b>Prioridad del requerimiento:</b> Alto	

<b>Identificación del requerimiento:</b>	RNF04
<b>Nombre del Requerimiento:</b>	Menu de navegación
<b>Características:</b>	El sistema deberá de tener un menú de navegación fácil de entender y de usar
<b>Descripción del requerimiento:</b>	La interfaz de usuario debe tener un menú de navegación que permita a los usuario desplazarse fácilmente entre cada una de las funcionalidades del sistema.
<b>Prioridad del requerimiento:</b> Alto	

<b>Identificación del requerimiento:</b>	RNF05
<b>Nombre del Requerimiento:</b>	Desempeño
<b>Características:</b>	El sistema garantizará a los usuarios un desempeño en cuanto a los datos almacenados en el sistema ofreciéndole una confiabilidad a esta misma.
<b>Descripción del requerimiento:</b>	La información almacenada o registros realizados podrán ser consultados y actualizados permanente y simultáneamente, sin que se afecte el tiempo de respuesta.
<b>Prioridad del requerimiento:</b> Medio	



<b>Identificación del requerimiento:</b>	RNF06
<b>Nombre del Requerimiento:</b>	Nivel de Usuario
<b>Características:</b>	Garantizara al usuario el acceso de información de acuerdo al nivel que posee.
<b>Descripción del requerimiento:</b>	Facilidades y controles para permitir el acceso a la información al personal autorizado dependiendo si es un administrador o un granjero.
<b>Prioridad del requerimiento:</b> Alta	

<b>Identificación del requerimiento:</b>	RNF07
<b>Nombre del Requerimiento:</b>	Confiabilidad continúa del sistema.
<b>Características:</b>	El sistema tendrá que estar en funcionamiento en todo momento cuando los usuarios lo dispongan.
<b>Descripción del requerimiento:</b>	La disponibilidad del sistema debe ser continua con un nivel de servicio para los usuarios, garantizando que el sistema esté disponible y pueda ser usado en cualquier momento.
<b>Prioridad del requerimiento:</b> Medio	

<b>Identificación del requerimiento:</b>	RNF08
<b>Nombre del Requerimiento:</b>	Seguridad en información
<b>Características:</b>	El sistema garantizará a los usuarios una seguridad en cuanto a la información que se procede en el sistema.
<b>Descripción del requerimiento:</b>	Garantizar la seguridad del sistema con respecto a la información y datos que se manejan tales sean documentos, archivos y contraseñas.
<b>Prioridad del requerimiento:</b> Medio	

## ***14.1 Requisitos comunes de las interfaces***

### ***14.1.1 Interfaces de usuario***

La interfaz con el usuario consistirá en un conjunto de vistas con botones, listas y campos de textos. Ésta deberá ser construida específicamente para el sistema propuesto y, será visualizada desde cualquier dispositivo con conexión a internet.

### ***14.1.2 Interfaces de hardware***

Cómo va a ser una aplicación web los requisitos mínimos de hardware están definidos por los necesario para ejecutar un navegador web y según el dispositivo se requieren:

- Para pc:
  - Procesador 2,60 GHz o superior
  - Memoria ram de 256 MB o superior
  - Almacenamiento disponible de 600 MB o superior
  - Mouse
  - Teclado
- Para dispositivos móviles:
  - Almacenamiento disponible de 100 MB o superior
  - Procesador Quad Core 1.2GHz o superior
  - Memoria ram de 256 MB o superior

### ***14.1.3 Interfaces de software***

Para ejecutar el sistema es necesario tener instalado un navegador web, se recomienda usar chrome, mozilla firefox o microsoft edge. Para ejecutar un navegador web se necesitan los siguientes sistemas operativos según el dispositivo a usar:

- Windows  
Windows 7, Windows 8, Windows 10 o una versión posterior
- Mac  
MacOS High Sierra 10.13 o una versión posterior
- Linux  
64-bit Ubuntu 18.04, Debian 10, openSUSE 15.2, Fedora Linux 32 o cualquier otra versión de 64 bits posterior a estas.
- Android:  
Android 7.0 Nougat o superior

#### ***14.1.4 Interfaces de comunicación***

El servidor y los usuarios se comunicarán entre sí, mediante protocolos estándares, siempre que sea posible. Por ejemplo, para transferir archivos o documentos deberán utilizarse protocolos existentes (FTP,TCP/IP).

#### ***14.2 Requisitos funcionales***

Requisito funcional 1

- **Validación de usuarios:** Los usuarios deberán identificarse para acceder a cualquier parte del sistema, dependiendo de su nivel de permisos.

Requisito funcional 2

- **Recuperación de contraseña:** El sistema permitirá a los usuarios recuperar su contraseña haciendo uso de su correo electrónico en donde se le enviará un código y éste tendrá que confirmarlo para cambiar su contraseña

Requisito funcional 3

- **Gestión de Galpones:** El sistema permitirá al administrador añadir, modificar o eliminar los galpones de la granja, cada galpón está identificado por un número, y tiene dimensiones de largo y ancho.

Requisito funcional 4

- **Gestión de Lotes:** El sistema permitirá al administrador añadir, modificar o eliminar los lotes, cada lote está identificado por un número, y tiene información acerca de, la fecha de nacimiento de las aves, la cantidad de aves, el galpón asignado y la raza de las aves.

Requisito funcional 5

- **Control de consumo de agua:** El sistema permitirá al granjero registrar la cantidad de agua en litros que las aves consumen diariamente de acuerdo a los cambios del nivel de agua en los tanques.

Requisito funcional 6

- **Control de mortalidad:** El sistema permitirá al administrador registrar la cantidad de aves que hayan muerto en un determinado día, automáticamente se debe actualizar la cantidad de aves en el lote.

Requisito funcional 7

- **Control de peso:** El sistema permitirá al granjero registrar el peso promedio de las aves en cada lote, para esto, el sistema sugerirá cuantas aves se deben pesar de acuerdo a la cantidad de aves en el lote. De acuerdo al peso obtenido y a la edad de las aves, el sistema sugerirá cuanto alimento se les debe proporcionar la siguiente semana.

#### Requisito funcional 8

- **Inventario de alimentos:** El sistema permitirá al administrador gestionar el inventario de alimentos registrando entradas (compras) y salidas de alimentos, editando, eliminando y añadiendo nuevos alimentos. Cada alimento tiene un precio, una medida, nombre y marca. Cada lote tiene su propio inventario de alimentos. Al granjero, sólo se le permitirá registrar la salida de alimentos.

#### Requisito funcional 9

- **Plan de vacunación:** El sistema permitirá al granjero gestionar el plan de vacunación de las aves, para ello calcula la dosis necesaria para el lote. El plan de vacunación incluye la fecha, el nombre de la vacuna a aplicar, la cantidad necesaria para el lote, el precio y el lote al que se le va a aplicar.

#### Requisito funcional 10

- **Control de costos:** El sistema permitirá al administrador gestionar los costos por lote este incluye costos de alimentación, consumo de agua, adecuación, vacunación, tratamientos y otros costos que puedan surgir durante el periodo de crianza de las aves.

#### Requisito funcional 11

- **Generación de reportes:** El sistema permitirá al administrador crear reportes de alimento consumido, mortalidad y peso corporal a través del tiempo.

### ***14.3 Requisitos no funcionales***

#### ***14.3.1 Requisitos de Interfaz***

- El sistema debe tener una interfaz de uso intuitiva y sencilla
- La interfaz de usuario debe ajustarse a las características de la lógica del negocio, dentro de la cual estará incorporado el sistema de gestión de procesos.

#### ***14.3.2 Ayuda en el uso del sistema***

- El sistema debe tener un apartado de ayuda, que le indique al usuario como se usa el sistema.
- El sistema no debe ser muy complicado de usar para evitar errores en los procesos y confusión por parte de los usuarios.

#### ***14.3.3 Encabezado del sistema.***

- El sistema deberá mostrar la información del software como su nombre, logotipo y eslogan.
- La información de la granja como nombre y ubicación. y la información del usuario como nombre, rol y última hora de conexión.

#### ***14.3.4 Diseño de la interfaz a la característica de la tienda tecnológica.***

- La interfaz de usuario debe tener un menú de navegación que permita a los usuario desplazarse fácilmente entre cada una de las funcionalidades del sistema.

#### ***14.3.5 Desempeño***

- El sistema deberá responder de manera rápida y eficiente a los procesos solicitados por los usuarios.
- El tiempo de respuesta del servidor debe ser menor a 5 segundos, pero dependiendo dependerá significativamente de la velocidad de conexión.

#### ***14.3.6 Nivel de usuario***

- El sistema deberá reconocer a los distintos niveles de usuario (Administrador o granjero).
- El Sistema deberá garantizar el acceso a la información dependiendo del nivel de usuario.

#### ***14.3.7 Confiabilidad continua del sistema***

- El sistema tendrá que estar en funcionamiento las 24 horas los 7 días de la semana.
- Garantiza un esquema adecuado que permita contar con un esquema de contingencia en una posible falla en cualquiera de sus componentes.

#### 14.3.8 Seguridad de información

- Garantizar la confiabilidad, la seguridad y 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, sin que se afecte el tiempo de respuesta.
- Garantizar la seguridad del sistema con respecto a la información y datos que se manejan tales sean documentos, archivos y contraseñas.
- Facilidades y controles para permitir el acceso a la información al personal autorizado, con la intención de consultar y subir información pertinente.

#### 15. Priorización de requisitos

Para esta parte del documento se usará una priorización de requisitos usando el método de escala ordinal donde se les asignará una escala de importancia a cada requisito usando la siguiente tabla.

Categoría de prioridad	Escala numérica
No esencial	1
Deseable	2
Obligatorio	3

Ahora se categoriza para cada requisito con la tabla de escala ordinal.

Requerimientos Funcionales	
Validación de usuarios	3
Recuperación de contraseña	1
Gestión de Galpones	3
Gestión de Lotes	3
Control de consumo de agua	1
Control de mortalidad	3
Control de peso	2
Inventario de alimentos	2
Plan de vacunación	2
Control de costos	1
Generación de reportes	1

Requisitos No funcionales	
Requisitos de Interfaz	2
Ayuda en el uso del sistema	1
Encabezado del sistema	3
Menu de navegación	3
Desempeño	1
Nivel de Usuario	2
Confiabilidad continua	2
Seguridad en la información	2