

DEPARTAMENTO DE INFORMÁTICA  
CARRERA: INGENIERÍA INFORMÁTICA

TRABAJO DE DIPLOMA PARA OPTAR POR EL TÍTULO DE  
INGENIERO INFORMÁTICO

**Aplicación web para la gestión de los datos asociados a los cultivos de la Empresa La Quinta**

**Web application for the management of data associated with the crops of Empresa La Quinta**

**Autor: Sandra Pérez Madrigal**

Tutores:

MSc. Bernardo León Ávila

Dr. Luis Quintero Domínguez

Este documento es Propiedad Patrimonial de la Universidad de Sancti Spíritus «José Martí Pérez», y se encuentra depositado en los fondos del Centro de Recursos para el Aprendizaje y la Investigación «Raúl Ferrer Pérez», subordinado a la Dirección General de Desarrollo 3 de la mencionada casa de altos estudios.

Se autoriza su utilización bajo la licencia siguiente:

**Atribución- No Comercial- Compartir Igual**



Para cualquier información, contacte con:

Centro de Recursos para el Aprendizaje y la Investigación “Raúl Ferrer Pérez”. Comandante Manuel Fajardo s/n, esquina a Cuartel, Olivos 1. Sancti Spíritus. Cuba. CP. 60100

Teléfono: **41-334968**

A mi familia y amigos, en especial a mis padres.

## **AGRADECIMIENTOS**

A mis padres y mi familia en general.

A mis amigos por toda la ayuda y paciencia.

A mis profesores, sin ellos no podría haber llegado hasta aquí.

A mis tutores por su confianza y apoyo durante todo este proceso.

A todos los que contribuyeron a la culminación de la tesis.

**GRACIAS.**

## **RESUMEN**

La agricultura en los últimos años ha sufrido las consecuencias del cambio climático, especialmente en lo que a la producción agrícola se refiere. La siembra dentro de invernaderos, vinculado a la agricultura inteligente podría contribuir a la solución que se necesita. Con la utilización de las Tecnologías de la Información y las Comunicaciones (TIC), se hace posible mejorar la producción, ya que se puede implementar la recolección, procesamiento y análisis de datos. En el presente trabajo se detalla la creación de una aplicación web que implementa el proceso de gestión de datos referentes a cultivos dentro de los invernaderos de la Empresa La Quinta. La misma hace posible la gestión de invernaderos, especies y variedades cultivables, tipos de suelo, cultivos y todas las acciones fito-técnicas que se les puedan realizar, así como las futuras cosechas de los mismos. Como resultado de este sistema se espera que apoye en la toma de decisiones dentro de los invernaderos, intentando así lograr el máximo potencial de la producción agrícola.

Palabras Clave: invernadero; gestión de datos; aplicación web.

## **ABSTRACT**

Agriculture in recent years has suffered the consequences of climate change, especially in terms of agricultural production. Planting inside greenhouses, linked to smart agriculture could contribute to the solution needed. With the use of Information and Communication Technologies (ICT), it becomes possible to improve production, since data collection, processing and analysis can be implemented. This paper details the creation of a web application that implements the process of managing crop data within the greenhouses of La Quinta Company. It makes possible the management of greenhouses, cultivable species and varieties, soil types, crops and all the phyto-technical actions that can be performed on them, as well as their future harvests. As a result of this system, it is expected to support decision making within the greenhouses, thus trying to achieve the maximum potential of agricultural production.

Keywords: greenhouse; data management; web application.

# ÍNDICE

<b>FIGURAS</b>	<b>X</b>
<b>TABLAS</b>	<b>XII</b>
<b>INTRODUCCIÓN</b>	<b>1</b>
<b>1. Fundamentos teóricos, metodológicos y tecnológicos que sustentan la creación de un sistema informático para la gestión de los datos asociados a los cultivos de la Empresa La Quinta</b>	<b>6</b>
1.1. Agricultura y proceso de cultivos agrícolas . . . . .	7
1.1.1. Agricultura . . . . .	7
1.1.2. Cultivo de la tierra y producción de alimentos . . . . .	7
1.1.3. Factores influyentes en el resultado de los cultivos . . . . .	8
1.1.3.1. Suelo . . . . .	8
1.1.3.2. Fertilizantes . . . . .	9
1.1.3.3. Riego . . . . .	9
1.1.4. Agricultura en la actualidad . . . . .	10
1.2. Gestión de información de cultivos agrícolas . . . . .	11
1.2.1. Agricultura inteligente . . . . .	11

1.2.2. Agricultura de precisión . . . . .	11
1.2.3. Agricultura de precisión dentro de los invernaderos . . . . .	12
1.2.4. Recolección de datos . . . . .	13
1.3. Metodologías y herramientas . . . . .	14
1.3.1. Tipos de aplicaciones . . . . .	14
1.3.2. Metodologías de desarrollo . . . . .	14
1.3.3. Lenguajes de programación . . . . .	15
1.3.4. Sistemas de bases de datos . . . . .	16
1.3.5. Frameworks . . . . .	17
1.4. Conclusiones parciales . . . . .	18
<b>2. Análisis y diseño de una aplicación web para la gestión de los datos asociados a los cultivos de la Empresa La Quinta</b>	<b>20</b>
2.1. Modelación del negocio . . . . .	20
2.1.1. Identificación de los procesos del negocio . . . . .	21
2.1.2. Reglas del negocio . . . . .	21
2.1.3. Actores y trabajadores del negocio . . . . .	21
2.1.4. Diagrama de casos de uso del negocio . . . . .	22
2.1.5. Diagrama de actividades de los casos de uso del negocio . . . . .	23
2.1.6. Diagrama de clases del modelo de objetos . . . . .	23
2.2. Necesidades y cualidades del sistema . . . . .	25
2.2.1. Requerimientos funcionales . . . . .	25
2.2.2. Requerimientos no funcionales . . . . .	27
2.2.3. Modelo de casos de uso del sistema . . . . .	29
2.3. Análisis y diseño del sistema . . . . .	32



2.3.1. Diagramas de Colaboración . . . . .	33
2.3.2. Diagramas de Clases del Diseño . . . . .	33
2.4. Diseño de la base de datos . . . . .	35
2.5. Conclusiones parciales . . . . .	37
<b>3. Validación del correcto funcionamiento de una aplicación web para la gestión de los datos asociados a los cultivos de la empresa La Quinta</b>	<b>38</b>
3.1. Ayuda, tratamiento de errores y seguridad . . . . .	38
3.1.1. Ayuda . . . . .	39
3.1.2. Tratamiento de errores . . . . .	39
3.1.3. Seguridad . . . . .	39
3.2. Prototipos de interfaz de usuarios . . . . .	40
3.3. Modelo de implementación . . . . .	43
3.4. Pruebas . . . . .	45
3.4.1. Pruebas al <i>backend</i> utilizando Postman . . . . .	45
3.4.2. Pruebas de rendimiento utilizando Firebug . . . . .	47
3.4.3. Pruebas de aceptación . . . . .	49
3.5. Conclusiones parciales . . . . .	50
<b>CONCLUSIONES</b>	<b>53</b>
<b>RECOMENDACIONES</b>	<b>54</b>
<b>REFERENCIAS</b>	<b>55</b>
<b>ANEXOS</b>	<b>59</b>
<b>A. Requerimientos funcionales detallados</b>	<b>60</b>

<b>B.</b>	<b>66</b>
B.1. Diagramas de colaboración (Caso de uso "Gestionar cultivos") . . . . .	66
B.2. Diagramas de clases de diseño (Caso de uso "Gestionar cultivos") . . . . .	66

# FIGURAS

2.1. Casos de uso del negocio . . . . .	23
2.2. Diagrama de actividades . . . . .	24
2.3. Modelo de objetos . . . . .	24
2.4. Modelo de casos de uso del sistema . . . . .	33
2.5. Diagrama de colaboración «Gestionar cultivos». Escenario «Insertar cultivo» . .	34
2.6. Diagrama de diseño «Insertar cultivo» . . . . .	35
2.7. Modelo de entidad-relación . . . . .	36
2.8. Modelo físico de la base de datos . . . . .	37
3.1. Página de inicio . . . . .	40
3.2. Interfaz "Insertar invernadero" . . . . .	41
3.3. Interfaz "Eliminar invernadero" . . . . .	41
3.4. Interfaz "Insertar cultivos" . . . . .	42
3.5. Interfaz "Gestionar cultivos" . . . . .	42
3.6. Interfaz "Gestionar acciones fitosanitarias" . . . . .	43
3.7. Interfaz "Insertar labor de cultivo (Drenaje)" . . . . .	44
3.8. Interfaz "Mostrar labor de cultivo (Drenaje)" . . . . .	44
3.9. Diagrama de despliegue . . . . .	45
3.10. Respuesta a la petición "Mostrar cultivos" . . . . .	46
3.11. Respuesta a la petición "Insertar cultivo" proporcionando parámetros correctos .	46
3.12. Respuesta a la petición "Modificar cultivo" proporcionando parámetros correctos	47
3.13. Respuesta a la petición "Eliminar cultivo" . . . . .	47
3.14. Respuesta a la petición "Autenticar usuario" . . . . .	48
3.15. Respuesta al caso de uso "Mostrar cultivos" . . . . .	48

---

3.16. Parámetros insertados para el caso de uso "Insertar cultivo" . . . . .	48
3.17. Respuesta ante los parámetros insertados para el caso de uso "Insertar cultivo" .	49
3.18. Parámetros insertados para el caso de uso "Modificar cultivo" . . . . .	49
3.19. Respuesta ante los parámetros insertados para el caso de uso "Modificar cultivo"	49
3.20. Respuesta ante credenciales inválidas para el caso de uso "Autenticar usuarios" .	50
3.21. Respuesta ante credenciales correctas para el caso de uso "Autenticar usuarios" .	50
3.22. Resultado de la ejecución para las entradas propuestas . . . . .	51
3.23. Ejemplos de diferentes mensajes de la aplicación . . . . .	52
B.1. Diagrama de colaboración «Gestionar cultivos». Escenario «Modificar cultivo» .	67
B.2. Diagrama de colaboración «Gestionar cultivos». Escenario «Eliminar cultivo» .	68
B.3. Diagrama de colaboración «Gestionar cultivos». Escenario «Mostrar cultivo» . .	68
B.4. Diagrama de diseño «Modificar cultivo» . . . . .	69
B.5. Diagrama de diseño «Eliminar cultivo» . . . . .	69
B.6. Diagrama de diseño «Mostrar cultivo» . . . . .	69

# TABLAS

- 2.1. Actores del negocio . . . . . 22
- 2.2. Trabajadores del negocio . . . . . 22
- 2.3. Requisitos funcionales . . . . . 26
- 2.4. Actores del sistema . . . . . 29
- 2.5. Descripción de los casos de uso del sistema . . . . . 29
- 2.5. Descripción de los casos de uso del sistema . . . . . 30
- 2.5. Descripción de los casos de uso del sistema . . . . . 31
- 2.5. Descripción de los casos de uso del sistema . . . . . 32
  
- 3.1. Caso de prueba para el caso de uso "Gestionar cultivos". Escenario "Insertar cultivo" . . . . . 50

# INTRODUCCIÓN

La agricultura, que se define como el arte de cultivar la tierra es la actividad que se ocupa de la producción de cultivo del suelo, el desarrollo y recogida de las cosechas ([Erazo Neira y Sánchez Escudero, 2022](#)). Debido al impacto generado por el cambio climático, la misma ha sufrido daños severos en su producción agrícola anual, lo cual ha generado un alto costo de mantenimiento y sostenibilidad, y, según estimaciones, para el año 2050 se espera un aumento de la población mundial, por lo que se requiere incrementar la producción de alimentos. Con las técnicas de agricultura convencional difícilmente se podrán satisfacer las nuevas demandas de alimentos, por lo que las granjas deben estar en capacidad de producir más alimentos con la misma o mejor calidad; y, para esto, los agricultores deben usar y aprovechar las Tecnologías de la Información y Comunicación (TIC).

En la actualidad se destaca la gestión de la información en los procesos agrícolas y el uso de TIC con el propósito de mejorar la producción a partir de la recolección, procesamiento y análisis de datos, y el uso de TIC es la base de la llamada Agricultura Inteligente o *Smart Farming*, que permite mejorar la producción y calidad de sus productos agropecuarios, así como gestionar y facilitar la ejecución de sus procesos ([Quintero Pulgar, 2021](#)).

Con el desarrollo de nuevas tecnologías orientadas hacia la agroindustria surge el concepto de agricultura de precisión (AP). La AP se define como el uso de un conjunto de tecnologías con el propósito de monitorear las condiciones físicas, químicas y climáticas del cultivo. El fin es optimizar el uso de recursos en el proceso productivo y la calidad final de los productos, tanto de las cosechas como sus derivados ([Amondaray et al., 2018](#)). Según plantea [Ofori y El-Gayar \(2021\)](#), la AP garantiza el uso eficiente de los recursos agrícolas para aumentar la productividad

y la sostenibilidad con el uso de la tecnología.

Una manera para lograrlo es mediante Granjas Inteligentes que produzcan alimentos nutritivos y de calidad, optimizando el uso de los recursos disponibles ([Quintero Pulgar, 2021](#)), y, de hecho, es una de las principales áreas donde se aplica la AP. El objetivo principal de estas instalaciones es proteger a las plantas de la incidencia de la intensa radiación solar y de las fuertes lluvias propias de los países tropicales. En ellas se intenta lograr que las plantas se desarrollen en un medio más favorable que el existente al aire libre ([Amondaray et al., 2018](#)). Como explican [Erazo Neira y Sánchez Escudero \(2022\)](#), son lugares cerrados, adecuados herméticamente, destinados al cultivo de todo tipo de plantas, y dentro de los invernaderos se puede hablar de ambientes controlados, lo que se refiere al control de variables que puedan afectar al cultivo, ya que un invernadero automatizado es un lugar accesible y diseñado para conservar un buen estado climático y del suelo para el cultivo.

No obstante, incluso contando con invernaderos, el hecho de no contar con el registro y control sistematizado de las variables ambientales provoca pérdidas debido a que los cultivos no alcanzan su máximo potencial si no se dispone de las condiciones adecuadas dentro del invernadero, y actualmente no se dispone de mecanismos efectivos y de bajo costo que contrarresten esta necesidad, limitando el máximo potencial de producción agrícola ([Martínez Pérez y López de Jiménez, 2021](#)).

Especialmente en las casas de cultivo protegido, el control de parámetros generalmente se realiza a través de la experticia de sus trabajadores y por estimaciones empíricas. Contar con un sistema de supervisión que ayude a lograr un máximo de seguridad, ahorro de recursos y apoye en la toma de decisiones en el interior de los invernaderos permitiría incrementar las producciones ([Madruga Peláez et al., 2019](#)). Y es que en la actualidad el mundo está dirigiendo todos los negocios y organizaciones hacia el uso de la analítica de datos para obtener más beneficios. Lo mismo ocurre con la agricultura inteligente, y para tratar los datos, los investigadores se enfrentan a una serie de retos ([Akhter y Sofi, 2021](#)).

Como explica [Akhter y Sofi \(2021\)](#), la recogida de datos es una de las tareas básicas y fundamentales cuando se va a realizar el análisis de datos, y esta recolección de información, junto a la supervisión y evaluación de un sistema de cultivo resulta determinante para una efectiva toma de decisiones ([Madruga Peláez et al., 2019](#)).

En Cuba con el objetivo de desarrollar una agricultura sostenible y productiva se han desarrolla-

do programas de inversiones para la fabricación y adquisición de equipamientos modernos que integran las nuevas tecnologías de automatización, control y comunicaciones (Madruga Peláez et al., 2019). A pesar de todos los esfuerzos, el desarrollo de novedosas tecnologías relacionadas con la AP y la administración (manejo o gestión) de la misma, es aún incipiente.

Al momento de investigación, son pocos, y prácticamente desconocidos los métodos de registro de datos de los cultivos que permitan realizar comparaciones entre una temporada y otra, para así optimizar este proceso, ya que los productores agrícolas no guardan registros, sino que se basan en su experiencia, y lograr niveles adecuados de calidad es fundamental, en ese sentido controlar aquellos procesos o etapas que inciden tanto en los costos como en la calidad es vital.

La Empresa La Quinta es una empresa asociada al sistema de agricultura urbana de la provincia de Sancti Spíritus, dedicada al cultivo de plantas tanto en entornos semiprotegidos como en entornos protegidos, es decir, invernaderos. Actualmente la empresa cuenta con más de 15 casas de cultivos, y se dedican a la venta de plantas en moneda nacional, para el sector del turismo y para la exportación, siendo sus principales productos el Chile Habanero, el pimiento y el pepino, aunque tienen un catálogo mucho más amplio.

En la empresa La Quinta, se presenta una **situación problemática** evidenciada por las siguientes manifestaciones:

- Los datos tienden a presentar incongruencias e inconsistencia, ya que la recolección de datos se realiza manualmente.
- La información queda registrada en formato físico en las oficinas de la empresa, lo que en ocasiones provoca pérdidas de información a causa del extravío o deterioro de los documentos.
- La generación de reportes se dificulta producto a la gran cantidad de datos manejados.
- La toma de decisiones se ve afectada a causa de no tener la capacidad de análisis de la información almacenada en el formato actual.

Con la situación problemática descrita anteriormente, podemos definir entonces como **problema de investigación**:

- ¿Cómo contribuir a la gestión de datos asociados a cultivos de la Empresa La Quinta?

Para dar respuesta al problema de investigación se formuló el siguiente **objetivo general**:



- Desarrollar un sistema informático para la gestión de los datos asociados a los cultivos de la empresa La Quinta.

Para dar solución al problema de investigación se definieron las siguientes **preguntas de investigación**:

1. ¿Cuáles son los fundamentos teóricos, metodológicos y tecnológicos que sustentan la creación de un sistema informático para la gestión de los datos asociados a los cultivos de la empresa La Quinta?
2. ¿Cómo diseñar y desarrollar un sistema informático para la gestión de los datos asociados a los cultivos de la empresa La Quinta?
3. ¿Cómo validar el correcto funcionamiento de un sistema informático para la gestión de los datos asociados a los cultivos de la empresa La Quinta?

En correspondencia con las preguntas de investigación se plantean las siguientes **tareas de investigación**:

1. Determinar los fundamentos teóricos, explicando los antecedentes bibliográficos, las bases teóricas y la definición de términos necesarios para comprender los cimientos del desarrollo de un sistema informático para la gestión de información.
2. Determinar los elementos metodológicos que contribuyan al desarrollo ágil, robusto y escalable de un software de gestión de información.
3. Determinar los fundamentos tecnológicos que permitan garantizar la usabilidad, seguridad y calidad de la solución que se desea desarrollar.
4. Diseñar y desarrollar un sistema informático utilizando la metodología seleccionada para facilitar la gestión de los datos asociados a los cultivos de la empresa La Quinta.
5. Validar el correcto funcionamiento del sistema informático para la gestión de los datos asociados a los cultivos de la empresa La Quinta.

El resto del informe de tesis queda estructurado en tres capítulos, arribándose a conclusiones concretas, seguido de las recomendaciones correspondientes y referencias bibliográficas.

En el Capítulo 1 se recoge el marco teórico referencial y los principales conceptos que constituyen la base teórica de la investigación, abordando temas como agricultura, proceso de cultivos

agrícolas, y la gestión de la información de los cultivos agrícolas. También en el primer capítulo se analizan las principales tendencias tecnológicas con el objetivo de seleccionar las adecuadas para el desarrollo del sistema informático requerido. En el Capítulo 2 se caracteriza el objeto de estudio y se describe el desarrollo del software a través de la metodología seleccionada, desarrollando la modelación del negocio, definiendo las necesidades y cualidades del sistema, y mostrando el diagrama entidad-relación así como el modelo físico de la base de datos. En el Capítulo 3 se realiza la validación de la solución propuesta a través de pruebas de aceptación de software mostrando las principales interfaces y funcionalidades del prototipo inicial. Para finalizar, un apartado de conclusiones 3.5 donde se verifica el cumplimiento de los objetivos trazados al inicio de la investigación, así como recomendaciones 3.5, donde se plasman una serie de propuestas encaminadas a la continuidad de esta investigación.

# **CAPÍTULO 1**

## **Fundamentos teóricos, metodológicos y tecnológicos que sustentan la creación de un sistema informático para la gestión de los datos asociados a los cultivos de la Empresa La Quinta**

En este capítulo se recoge el marco teórico referencial y los principales conceptos que constituyen la base teórica de la investigación, abordando temas como agricultura, proceso de cultivos agrícolas, y la gestión de la información de los cultivos agrícolas. Además, se analizan las principales tendencias tecnológicas con el objetivo de seleccionar las adecuadas para el desarrollo del sistema informático requerido.

### 1.1. Agricultura y proceso de cultivos agrícolas

#### 1.1.1. Agricultura

La RAE define la agricultura como «cultivo o labranza de la tierra; conjunto de técnicas y conocimientos relativos al cultivo de la tierra». Agricultura es la palabra más completa que se utiliza para denotar las muchas formas en que las plantas de cultivo sostienen a la población humana mundial proporcionando alimentos y otros productos. Es una forma de uso de la tierra y de economía que resultó de la combinación del cultivo (un conjunto de acciones humanas centradas en la preparación del suelo y la plantación, el cuidado y la cosecha de las plantas) y la domesticación (un conjunto de cambios genéticos y morfológicos que han aumentado la capacidad de las plantas para adaptarse al cultivo) ([Harris y Fuller, 2014](#)).

La agricultura se considera la base de la vida de la especie humana, ya que es la principal fuente de granos y otros alimentos. Como explica [Gondchawar y Kawitkar \(2016\)](#), el crecimiento del sector agrícola desempeña un papel fundamental dentro de cada país en lo que al crecimiento de la economía se refiere, y proporciona amplias oportunidades de empleo a la población.

#### 1.1.2. Cultivo de la tierra y producción de alimentos

El cultivo representa un cambio importante en la estrategia humana, ya que las personas comienzan a manipular el suelo y la composición de las comunidades vegetales para mejorar el rendimiento de determinadas plantas más adelante. Es una actividad a través de la cual los humanos se involucran directamente en la gestión de la vida y los ciclos vitales de ciertas plantas. En términos abstractos, esto puede considerarse un cambio desde un enfoque de subsistencia en gran medida de recolección hacia uno altamente regulador. En la práctica, el cultivo implica la manipulación del suelo, el agua y otros componentes del entorno vegetal. En su forma más básica, implica la siembra de semillas en un suelo que ha sido limpiado de otra vegetación ([Harris y Fuller, 2014](#)). La tierra cultivable se está reduciendo lentamente debido a las construcciones (casas, presas, fábricas, industrias, etc.) y a la reducción natural de la tierra cultivable (inundaciones, terremotos, desprendimientos, etc.) ([Akhter y Sofi, 2021](#)). Aún así, como explica [Gondchawar y Kawitkar \(2016\)](#) muchos agricultores siguen utilizando los métodos tradicionales de cultivo, lo que se traduce en un bajo rendimiento de las cosechas y los frutos. Pero allí donde se ha imple-

## 1.1 Agricultura y proceso de cultivos agrícolas

---

mentado la automatización y se ha sustituido a los seres humanos por maquinarias automáticas, el rendimiento ha mejorado.

Según [Harris y Fuller \(2014\)](#), los inicios de la producción de alimentos representan un cambio estratégico en el comportamiento humano hacia la manipulación del entorno del suelo y a través de una influencia en la composición de las poblaciones de plantas que crecen en ese suelo, mediante la siembra y el cuidado de una o unas pocas especies.

### 1.1.3. Factores influyentes en el resultado de los cultivos

#### 1.1.3.1. Suelo

El cultivo de la tierra para la producción de alimentos suele implicar la preparación del suelo mediante el labrado ([Harris y Fuller, 2014](#)). El suelo es el elemento básico de toda práctica agraria. Sin suelo, la palabra cultivo no existe, por lo tanto, analizar el estado del suelo es el primer y principal paso hacia la mejor práctica agraria. Haciendo pruebas de suelo podemos analizar el estado físico, biológico y químico del suelo. En base a esto, los agricultores pueden tomar decisiones en consecuencia en relación con sus campos. Al hacer las pruebas de suelo podemos analizar los nutrientes del suelo que incluyen la información relacionada con las necesidades de fertilizantes, la historia de la cosecha, el tipo de suelo, el nivel de riego, etc ([Akhter y Sofi, 2021](#)).

Según la [FAO \(2022\)](#), existen diferentes tipos de suelo. Su clasificación consiste en agrupar suelos con propiedades similares (químicas, físicas y biológicas) en unidades que puedan georeferenciarse y cartografiarse. Son un recurso natural muy complejo, mucho más que el aire y el agua, contienen todos los elementos químicos naturales y se combinan simultáneamente en estado sólido, líquido y gaseoso. Además, el número de características físicas, químicas y biológicas y sus combinaciones son casi infinitas .

Los suelos son la base de la agricultura y el medio en el que crecen casi todas las plantas que producen alimentos. Los suelos sanos producen cultivos sanos que, a su vez, nutren a las personas y los animales. De hecho, la calidad del suelo está directamente relacionada con la calidad y cantidad de los alimentos. Los suelos aportan los nutrientes esenciales que nuestras plantas productoras de alimentos necesitan para crecer y prosperar. También sirven de amortiguador para proteger las delicadas raíces de las plantas de las drásticas fluctuaciones de temperatura. Un

## 1.1 Agricultura y proceso de cultivos agrícolas

---

suelo sano permite mejorar la producción de cultivos [FAO \(2015\)](#).

### 1.1.3.2. Fertilizantes

Otra variable importante es la adición de nutrientes al suelo por medios como el abono, los cultivos múltiples con especies fijadoras de nitrógeno (generalmente leguminosas de la familia de las fabáceas), o el uso de rotaciones de cultivos con leguminosas o períodos de barbecho. Esto representa un componente importante del cultivo, es decir, la programación de las estaciones de siembra y cosecha y los patrones interanuales en la rotación de cultivos y el descanso ([Harris y Fuller, 2014](#)).

Los fertilizantes aportan los nutrientes al suelo que, a su vez, se transfieren a las plantas a través de las raíces. Los fertilizantes como el potasio (K), el fósforo (P), el nitrógeno (N), etc., tienen sus propios usos para mantener la salud de la flora. Las plantas necesitan principalmente macro nutrientes NPK para mantener la salud. El P se utiliza para mantener la salud de las raíces, las flores y ayuda al desarrollo de los frutos, el K se utiliza para el crecimiento del tallo y para el buen funcionamiento del xilema, de modo que el agua pueda transferirse a todas las partes de la planta, y el N ayuda al crecimiento de las hojas. Por lo tanto, hay que evitar cualquier tipo de deficiencia de fertilizantes NPK en las plantas aunque el uso excesivo de los mismos también puede conducir a malos resultados e incluso al desequilibrio del ecosistema ([Akhter y Sofi, 2021](#)).

### 1.1.3.3. Riego

El agua es un insumo clave en cualquier sistema de cultivo, y en algunas regiones tuvo un papel central en los orígenes de la agricultura. Los sistemas de riego son una forma notable y generalizada de crear paisajes agrícolas distintivos. El control del agua puede centrarse tanto en su eliminación (drenaje) como en la adición de agua a zonas que, de otro modo, serían localmente secas para permitir el cultivo cuando las precipitaciones son insuficientes para mejorar la productividad ([Harris y Fuller, 2014](#)). El riego de los campos de cultivo en base a las necesidades es una de las mejores maneras de generar más y más cosechas ([Akhter y Sofi, 2021](#)).

### 1.1.4. Agricultura en la actualidad

La agricultura 4.0 actualiza los métodos de producción tradicionales y las estrategias de la agricultura mundial hacia una cadena de valor optimizada utilizando una serie de tecnologías emergentes que potencian las soluciones disruptivas en todas las etapas de la cadena de producción agrícola (da Silveira et al., 2021). He ahí la necesidad de aplicar la ciencia y la tecnología modernas en el sector agrícola para aumentar el rendimiento. La mayoría de los artículos señalan el uso de redes de sensores inalámbricos que recogen los datos de diferentes tipos de sensores y los envían al servidor principal mediante un protocolo inalámbrico. Los datos recogidos proporcionan información sobre diferentes factores ambientales que, a su vez, ayudan a controlar el sistema.

El control de los factores ambientales no es una solución suficiente y completa para mejorar el rendimiento de los cultivos. Hay otros factores que afectan a la productividad en gran medida. Estos factores incluyen el ataque de insectos y plagas, que pueden controlarse rociando el cultivo con insecticidas y pesticidas adecuados. En segundo lugar, el ataque de animales salvajes y pájaros cuando la cosecha crece. También existe la posibilidad de que se produzcan robos cuando la cosecha se encuentra en la fase de recogida. Incluso después de la cosecha, los agricultores también se enfrentan a problemas en el almacenamiento de los cultivos cosechados. Por lo tanto, con el fin de proporcionar soluciones a todos estos problemas, es necesario desarrollar un sistema integrado que se encargue de todos los factores que afectan a la productividad en todas las etapas como: el cultivo, la cosecha y el almacenamiento posterior a la cosecha (Gondchawar y Kawitkar, 2016).

En la actualidad se está produciendo una profunda transformación social, económica y política motivada por el desarrollo de las TIC. Se está asistiendo a un proceso de grandes cambios tecnológicos que permiten configurar la denominada Sociedad de la Información (Rodríguez et al., 2016). Hoy en día, cuando las tierras agrícolas y los recursos hídricos son más valiosos que nunca, es muy importante integrar las tecnologías de comunicación inalámbrica con soluciones más realistas y holísticas en las aplicaciones de agricultura inteligente (Cicioğlu y Çalhan, 2021).

## 1.2. Gestión de información de cultivos agrícolas

### 1.2.1. Agricultura inteligente

Más detalladamente, la agricultura inteligente es un concepto de gestión de los cultivos que permite a los agricultores gestionar la variabilidad espacial y temporal, como la reducción de los recursos naturales en la agricultura, la gestión del riego, la gestión de la producción y la gestión de los fertilizantes, los ataques de intrusión y la supervisión de los datos en tiempo real, y tiene como objetivo reducir la carga de trabajo de los agricultores y aumentar la productividad de las explotaciones mediante el uso de varias tecnologías nuevas. La gestión inteligente de la agricultura se está desarrollando día a día con la ayuda de periféricos de *IoT*, sistemas de procesamiento de imágenes, capacidades de *big data*, análisis de datos y tecnologías inalámbricas (Cicioğlu y Çalhan, 2021).

El desarrollo de nuevas tecnologías agrícolas ha sido, y siempre será, esencial. Los gobiernos invierten miles de millones de dólares para la agricultura por las necesidades nutricionales en el país y los ingresos de exportación, por lo que la máxima productividad es el principal objetivo de la agricultura inteligente (Cicioğlu y Çalhan, 2021). La agricultura inteligente es importante para abordar los desafíos de la producción agrícola en términos de productividad, impacto ambiental, seguridad alimentaria y sostenibilidad (Kamilaris et al., 2017).

### 1.2.2. Agricultura de precisión

La agricultura a nivel mundial está influida por los nuevos paradigmas tecnológicos; por lo que es favorecida con los enormes avances en la Informática y las Comunicaciones. Con el desarrollo de nuevas tecnologías orientadas hacia la agro-industria surge el concepto de AP, definida como el uso de un conjunto de tecnologías con el propósito de monitorear las condiciones físicas, químicas y climáticas del cultivo a fin de optimizar el uso de recursos en el proceso productivo y la calidad final de los productos, tanto de las cosechas como sus derivados (Amondaray et al., 2018).

La AP, que existe desde hace más de cuatro décadas, garantiza el uso eficiente de los recursos agrícolas para aumentar la productividad y la sostenibilidad con el uso de la tecnología (Ofori y El-Gayar, 2021). Es un concepto bastante nuevo de gestión agrícola desarrollado a mediados



## 1.2 Gestión de información de cultivos agrícolas

---

de los años 80 y en este trabajo, en el término «tecnología» se incluye el conjunto completo de herramientas disponibles para la gestión de la AP. Se han estudiado muchos aspectos de la AP, centrándose en: las tecnologías relevantes, los efectos ambientales, los resultados económicos, las tasas de adopción y los impulsores de la adopción y la no adopción. Muchos autores han confirmado los beneficios ambientales y económicos derivados de la AP.

Las actividades relacionadas con la AP incluyen: identificación y localización de cultivos, insectos y malezas, monitoreo del rendimiento, maquinaria, dosificación variable de fertilizantes, herbicidas, insecticidas y fungicidas, monitoreo de la siembra y mapeo. Para maximizar el rendimiento tanto cualitativo como cuantitativo, los buenos agricultores no deben dudar nunca en adoptar las tecnologías modernas, porque todos los agricultores están interesados en producir la mayor parte de sus campos, aumentando la cantidad y la calidad de la cosecha. El control del rendimiento supervisa toda la superficie de cultivo desde el principio hasta el final, es decir, desde el cultivo hasta la cosecha. La recogida de datos es una de las tareas básicas y fundamentales a la hora de realizar un análisis de ellos ([Akhter y Sofi, 2021](#)).

La AP y su conjunto de tecnologías de la información permiten a los operadores agrícolas ajustar sus prácticas de producción. El acceso a información detallada dentro del campo puede reducir los costes de los insumos y aumentar el rendimiento. Las tecnologías de AP pueden reducir los costes al evitar que los agricultores apliquen excesivamente los insumos ([Schimmelpfennig, 2016](#)).

### 1.2.3. Agricultura de precisión dentro de los invernaderos

Una de las principales áreas donde se aplica la AP es en las casas de cultivos protegidos. Estas instalaciones tienen como objetivo proteger a las plantas de la incidencia de la intensa radiación solar y de las fuertes lluvias propias de los países tropicales. En ellas se modifican total o parcialmente las condiciones ambientales, para que las plantas se desarrollen en un medio más favorable que el existente al aire libre. El auge de las casas de cultivo en Cuba se inicia a partir de la transferencia de tecnologías de otros países, principalmente Israel y España. Sin embargo, teniendo en cuenta las características del clima, a los diseños originales se les incluyen modificaciones, buscando el llamado efecto sombrilla, de manera que puedan producirse hortalizas durante todo el año, y fundamentalmente en verano, en momentos en que resulta imposible lograrlo a cielo abierto ([Amondaray et al., 2018](#)).

## 1.2 Gestión de información de cultivos agrícolas

---

La eficiencia en el uso de los recursos y el rendimiento de los cultivos son factores importantes en la gestión de los invernaderos agrícolas. Los métodos de modelización adecuados mejoran eficazmente el rendimiento y la eficiencia del sistema de invernaderos y favorecen el diseño de estrategias de ahorro de agua y energía. Al mismo tiempo, se puede prever con antelación el entorno extremo, lo que reduce las plagas y enfermedades y proporciona alimentos de alta calidad. En consecuencia, el interés de la comunidad científica por la modelación y optimización de los invernaderos ha crecido considerablemente ([Guo et al., 2021](#)).

### 1.2.4. Recolección de datos

El mundo actual está dirigiendo todos los negocios y organizaciones para obtener más beneficios utilizando el análisis de datos. Lo mismo ocurre con la agricultura inteligente. Para tratar con los datos, los investigadores se enfrentan a una serie de retos ([Akhter y Sofi, 2021](#)).

Para hacer frente a los crecientes desafíos de la producción agrícola, es necesario comprender mejor los complejos ecosistemas agrícolas. Esto puede ocurrir mediante modernas tecnologías digitales que monitorizan continuamente el entorno físico, produciendo grandes cantidades de datos a un ritmo sin precedentes. El análisis de estos (grandes) datos permitiría a los agricultores y a las empresas extraer valor de ellos, mejorando su productividad. El análisis de *big data* en la agricultura no se ha aplicado ampliamente, aunque está dando lugar a avances en diversas industrias en que se está utilizando como la banca, los seguros, la comprensión del comportamiento del usuario en línea y la personalización, así como en los estudios medioambientales. Es una técnica moderna con creciente popularidad, es la capacidad de buscar, agregar, visualizar y cruzar grandes conjuntos de datos en un tiempo razonable. Se trata de la capacidad de extraer información y conocimientos allí donde antes era económica o técnicamente inviable hacerlo ([Kamilaris et al., 2017](#)).

### 1.3. Metodologías y herramientas

#### 1.3.1. Tipos de aplicaciones

En la actualidad las aplicaciones informáticas se pueden clasificar como aplicación *desktop* (o de escritorio), aplicaciones para móviles (*apps*) o aplicaciones web. Las mismas están diseñadas como herramientas para realizar operaciones o funciones específicas. Generalmente, para facilitar ciertas tareas complejas y hacer más sencilla la experiencia de los usuarios.

A criterio de Müller (2021), debido al éxito y el dominio de la Web, las aplicaciones de escritorio tradicionales son cada vez menos importantes, tanto para los usuarios como para las empresas. La plataforma móvil se ha vuelto cada vez más destacada, aún así, las empresas prefieren ofrecer sus servicios a través de una interfaz web, donde pueden confiar cómodamente en tecnologías probadas. Para los usuarios, esto simplifica el acceso a los servicios desde cualquier lugar, y también elimina la necesidad de instalar una aplicación en sus máquinas. La web, y el modo en que accedemos a ella y la utilizamos está evolucionando. Hoy en día, cada vez más personas acceden a la web mediante dispositivos móviles y aplicaciones en lugar de ordenadores de sobremesa, por lo tanto, sigue siendo una prioridad.

#### 1.3.2. Metodologías de desarrollo

La metodología ágil es uno de los procesos de ingeniería de software que es famoso por su método iterativo de desarrollo. Es especialmente útil en el contexto de una evolución continua de los requisitos y soluciones, creando una base sólida y ayudando al equipo a responder a los cambios de forma natural y más rápida. Algunos de sus tipos más conocidos son *Extreme Programming* (XP) y *Scrum* (Dragos, 2021).

La metodología XP tiene una aplicación flexible, es una metodología con valores muy notables como la comunicación, la retroalimentación, la simplicidad, el respeto y la valentía. La *metodología XP* se centra en garantizar que el proyecto se entregue a tiempo y con el máximo valor posible, buscando siempre que el cliente quede satisfecho con el producto (Souza y Oliveira, 2021).

Scrum es una metodología de gestión más amplia que no se limita al mundo del desarrollo de

software. Se trata de una metodología esencialmente interactiva e incremental, es decir, el programa se desarrolla por etapas y al final de cada una de ellas se inicia una nueva. En *Scrum*, los cambios son aceptables durante el transcurso del proyecto porque estos cambios pueden implementarse en etapas futuras sin influir directamente en la etapa actual del proyecto. Durante cada etapa del proyecto, los desarrolladores deben codificar y probar, y al final entregar un prototipo de trabajo para que el cliente lo pruebe y evalúe (Souza y Oliveira, 2021).

La metodología *Rational Unified Process* (RUP) es una de las metodologías tradicionales. Esta metodología se centra en la creación y el mantenimiento de modelos, más que en producir grandes cantidades de documentos. Es un proceso de desarrollo de software incremental que ofrece versiones con mejoras en cada versión mientras se siguen las mejores prácticas a lo largo del camino. Las cuatro fases de *RUP* proporcionan un enfoque secuencial e iterativo para el desarrollo de un producto. El flujo de trabajo se divide en actividades lógicas, no es solo un proceso de desarrollo de sistemas, sino también una mentalidad que puede aplicarse a cualquier proceso o proyecto. Las personas responsables de *RUP* trabajan en estrecha colaboración con clientes, socios y grupos de productos para garantizar que sea continuamente actualizada y mejorada para estar a la altura de los nuevos tiempos. *RUP* también se apoya en herramientas que ayudan a automatizar gran parte del proceso (Dragos, 2021). Lo más destacable de *RUP* con respecto a las metodologías ágiles es que la cantidad de artefactos que genera brinda una mejor documentación, lo que facilita que en futuros intentos de optimización del software sea comprensible el proyecto para otros desarrolladores.

### 1.3.3. Lenguajes de programación

Cutting y Stephen (2021) plantean que debe elegir el lenguaje de programación según los objetivos planteados. En el caso de que se quiera hacer un juego, un lenguaje como JavaScript, Java, C o C ++ podría funcionar, y si se espera realizar un buen trabajo con la Web, o Inteligencia Artificial, etc., lenguajes como Python, JavaScript o Ruby son buenas opciones. De esta manera, desde el inicio se debe saber que tipo de sistema informático se quiere desarrollar, o elegir un área de interés. A partir de ahí según sus inclinaciones, se debe elegir el lenguaje.

Java, creado por Sun Microsystems y lanzado en 1995, se convirtió rápidamente en un lenguaje de programación de propósito general ampliamente adoptado, especialmente para entornos en línea (es decir, computación en la nube). Hoy en día, el software escrito con Java impulsa

innumerables teléfonos inteligentes, centros de datos y servicios en línea omnipresentes como Google. A partir de 2021, se encuentra entre los lenguajes de programación más populares tanto para principiantes como para programadores experimentados (Ciesla, 2021).

Aunque el lenguaje de programación conocido como JavaScript comparte sus cuatro primeras letras con Java, ambos tienen muy poco en común. JavaScript fue creado por los pioneros de Internet, Netscape, en la década de los noventa para su uso con la entonces nueva tecnología de los navegadores. El lenguaje sigue vivo en Internet, proporcionando una funcionalidad y facilidad de uso adicionales para una miríada de sitios web (Ciesla, 2021).

Python fue lanzado en 1991, y se ha hecho bastante popular en los últimos años. Es un lenguaje de programación que permite trabajar con rapidez e integrar sistemas con mayor eficacia (Python, 2022). Lo más destacable de Python es su sensibilidad a los espacios en blanco/indentación. Es capaz de hacer prácticamente cualquier cosa, desde aplicaciones sencillas para la web hasta software de escritorio más pesado. Python ha encontrado incluso un nicho en la industria de los videojuegos para la creación de prototipos de juegos (Ciesla, 2021).

### 1.3.4. Sistemas de bases de datos

Una base de datos es una colección de información organizada y guardada en un ordenador para que pueda ser examinada mediante un programa informático y obtener información de ella. En teoría, consta de dos componentes principales: los datos y la información, siendo entonces el objetivo final gestionar los datos para que puedan transformarse en la información adecuada y la recuperación, eliminación y edición de datos pueda hacerse de forma cómoda y rápida (eficaz, eficiente y precisa) (Rawat et al., 2021).

MySQL es un sistema gestor de base de datos orientado a servicios, por lo que no requiere una interfaz de usuario en el escritorio o en la barra de tareas para ejecutarse. Puede utilizarse en modo texto, en modo comando o en PHPMyAdmin (aplicación *online* para gestionar y administrar servidores y bases de datos, así como los objetos que residen en ellos) (Rawat et al., 2021).

SQLite es una biblioteca de lenguaje que sólo requiere unas pocas llamadas C estándar que deberían estar disponibles en cualquier sistema operativo. Se puede construir a partir de un único archivo fuente y no se requiere la configuración de un servidor. SQLite puede almacenar sus bases de datos como un archivo normal en el disco. Para utilizar SQLite en una aplicación todo

lo que se debe hacer es una sola inclusión de la biblioteca y todo estará listo. El proceso de configuración es increíblemente fácil y rápido (Braun, 2021).

PostgreSQL es una de las opciones más sólidas cuando de base de datos se habla, y cuenta con una gran comunidad de colaboradores. Según la documentación oficial de PostgreSQL (2022), es un potente sistema de bases de datos objeto-relacional de código abierto con más de 35 años de desarrollo activo que le ha granjeado una sólida reputación por su fiabilidad, robustez de funciones y rendimiento. Es un software totalmente gratuito, multiplataforma y es muy fácil de usar. Se puede administrar utilizando la herramienta pgAdmin y permite manejar grandes volúmenes de datos.

### 1.3.5. Frameworks

Aunque se utilicen muchas tecnologías diferentes, los *frameworks* siguen siendo un aspecto esencial del proceso de desarrollo de sistemas conectados de forma inteligente. Dado que este sector aún está en sus inicios, actualmente no hay un claro vencedor en este campo (Totey y Vishwakarma, 2021). Aún así se puede realizar un estudio de los principales para seleccionar los que se ajusten a los requerimientos de cada sistema.

Django es un marco de trabajo (*framework*) para el desarrollo de aplicaciones web usando Python. Considera algunas funcionalidades listas para usar que facilitan el desarrollo de aplicaciones web. Como resultado, no es necesario escribir todo el código ni usar tiempo para buscar errores de código en el *framework* (Vidal-Silva et al., 2021). Django (Foundation, 2022) se basa en el estilo Modelo-Vista-Plantilla, y proporciona una división entre las reglas de negocio, los datos y la interfaz (Thoutam, 2021).

Node.js es un entorno de ejecución de código abierto y multiplataforma para ejecutar código JavaScript. La documentación oficial de Node.js (2022) la define de la siguiente manera: «Node.js es un entorno de ejecución para JavaScript construido con el motor JavaScript V8 de Chrome». Significa que para transformar el texto JavaScript en código ejecutable para que la computadora lo procese, utiliza V8, el motor JavaScript de código abierto creado por Google para alimentar los navegadores Chromium y Chrome. Node.js tiene un ecosistema de paquetes en NPM (*Node Package Manager*), que es el ecosistema abierto más grande del mundo, con más de 350 mil paquetes. Le permite usar el mismo lenguaje de programación para el *frontend* y el *backend*

([Calcina Vilcapaza, 2022](#)).

Nuxt.js es un *framework* basado en Vue.js para crear aplicaciones web modernas. Se basa en las bibliotecas oficiales de Vue.js (vue, vue-router y vuex). Es un *framework* de código abierto que simplifica y potencia el desarrollo web [Nuxt \(2022\)](#). Está diseñado para darle una arquitectura sólida siguiendo las pautas oficiales de Vue, se puede utilizar para crear desde páginas de destino estáticas hasta complejas aplicaciones web listas para empresas. Es un *framework* completo que posee varios módulos incluidos como la autenticación básica, oauth2 y con redes sociales, el cual nos provee una estructura de proyecto bien definida. También es muy flexible para incluir cualquier biblioteca de componentes para la interfaz; en general está diseñado para construir aplicaciones de gran escala, y agilizar y automatizar algunos procesos en el desarrollo *frontend*. Además, se logra crear componentes reutilizables, esto permitirá que el tiempo del desarrollo de los siguientes módulos sean más cortos, de manera que no se dupliquen innecesariamente ([Calcina Vilcapaza, 2022](#)).

Vuetify es un *framework* de componentes, su objetivo es proporcionar componentes limpios, semánticos y re-utilizables que faciliten la creación de su aplicación. Vuetify utiliza el patrón de diseño Material Design de Google, tomando señales de otros marcos populares como Materialize.css, Material Design Lite, UI semántica y Bootstrap 4 (Vuetify, 2020).([Calcina Vilcapaza, 2022](#))

## 1.4. Conclusiones parciales

Luego del estudio realizado en el presente capítulo, se arriban a las siguientes conclusiones parciales:

- El estudio del marco teórico permitió realizar un análisis sobre los diferentes tipos de aplicaciones, determinando las aplicaciones web como la más adecuada para el desarrollo de la investigación.
- Se seleccionó la metodología *Rational Unified Process (RUP)* como la adecuada para el proceso de desarrollo por ser esta la más destacada de las metodologías tradicionales.
- Se decidió utilizar *PostgreSQL* como gestor de base de datos por ser un gestor sólido con una amplia comunidad de desarrollo y de software libre.

## 1.4 Conclusiones parciales

---

- Se determinó utilizar para el desarrollo de la aplicación el framework Django en el *backend*, al ser un framework establecido y ampliamente utilizado en este tipo de aplicaciones. Además, para el *frontend* se decidió utilizar *Nuxt.js* que destaca por sus facilidades para el desarrollo rápido y basado en componentes.



# CAPÍTULO 2

## **Análisis y diseño de una aplicación web para la gestión de los datos asociados a los cultivos de la Empresa La Quinta**

En este capítulo se caracteriza el objeto de estudio del presente proyecto, y se describe el desarrollo del software a través de la metodología seleccionada, en este caso RUP. Se realiza un estudio del modelo del negocio identificando las reglas del negocio, los actores y trabajadores que intervienen en el mismo, el diagrama de caso de uso del negocio, diagrama de actividades y el modelo de objetos, lo que permite una mejor comprensión de la problemática a tratar. Además se presentan las necesidades y cualidades del sistema, mostrando los requerimientos funcionales y no funcionales, el modelo de casos de uso del sistema y la descripción de cada uno de ellos, así como los actores del mismo. Finalmente se muestran los diagramas correspondientes a las clases del diseño y colaboración así como también los relacionados a la base de datos.

### **2.1. Modelación del negocio**

El modelado de negocios ayuda al equipo de desarrollo a comprender los elementos que intervienen en los procesos de negocios y las necesidades actuales de los usuarios en las empresas. *RUP* ofrece actividades para modelar el negocio, y *UML* ofrece diagramas que ayudan a modelar el

negocio (Edith, 2019).

### 2.1.1. Identificación de los procesos del negocio

La identificación de los procesos del negocio consiste en identificar entidades y trabajadores que participan en la realización de los casos de uso del negocio (Edith, 2019).

Se identifica el siguiente proceso de negocio:

- Proceso de registro de información de datos asociados a cultivos.

Dicho proceso ocurre en la Empresa La Quinta. En un primer momento, el técnico agrónomo decide la acción a realizar, ya sea la siembra o cosecha de un cultivo, o la realización de labores de cultivos o fito-sanitarias a un cultivo ya existente. A continuación el mismo le informa al agricultor la acción que debe realizar y este posteriormente lo ejecuta. Una vez ha terminado, el agricultor se dirige al técnico agrónomo y le informa sobre los detalles de la acción realizada. El técnico registra la información en el modelo de registro de datos de cultivos y lo archiva finalizando así el proceso.

### 2.1.2. Reglas del negocio

Las reglas del negocio representan una serie de restricciones de la organización para realizar las actividades o asociadas a la información. Las mismas determinan las políticas y estructura de la información, regulan el funcionamiento de la empresa, describen restricciones y comportamientos, y aunque no son requisitos influyen en ellos (Edith, 2019).

A continuación, se exponen las que fueron identificadas en este caso:

- El agricultor no puede realizar una acción a menos que el técnico le informe que debe realizarla.

### 2.1.3. Actores y trabajadores del negocio

Un actor puede ser una persona o un grupo de personas homogéneas, otro sistema, o una máquina. Lo que se modela como actor es el rol que se juega cuando interactúa con el negocio para

Tabla 2.1: Actores del negocio

Actor	Descripción
Técnico	Interesado en la recopilación de datos referentes a los cultivos.

Tabla 2.2: Trabajadores del negocio

Trabajador	Descripción
Técnico	Es el responsable de orientar la realización de un proceso de cultivo, cosecha o la realización de acciones a cultivos existentes. Es el encargado de recopilar los datos de las acciones realizadas por el agricultor y archivarlas.
Agricultor	Es quien realiza las acciones agrícolas orientadas por el técnico.

beneficiarse de sus resultados(González, 2005).

A continuación, se presentan los actores, caso de uso del negocio y diagrama de actividad perteneciente al caso de uso del negocio.

La Tabla 2.1 muestra el actor de negocio y su descripción, del proceso registrar datos asociados a cultivos.

Los trabajadores del negocio son el punto de partida para derivar un primer conjunto de actores y casos de uso del sistema (Edith, 2019). Un trabajador del negocio no es un actor del negocio. Es un representante del negocio con el cual un actor del negocio interactúa y que se transformará en un actor en los casos de uso del sistema, donde utiliza el sistema como herramienta para prestarle un servicio a los actores del negocio (Metzner y Niño, 2016).

Los trabajadores identificados en el negocio se muestran en la Tabla 2.2:

### 2.1.4. Diagrama de casos de uso del negocio

El modelo de casos de uso del negocio describe los proceso de negocio de una empresa en términos de casos de uso del negocio y actores del negocio que corresponden con los procesos del negocio y con los clientes. Un modelo de casos de uso presenta un sistema (el negocio) desde la perspectiva de su uso y esquematiza cómo proporciona valor a sus usuarios. Se describe mediante diagramas de casos de uso (Edith, 2019).

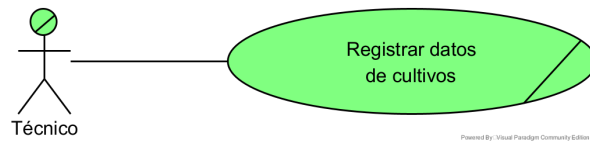


Figura 2.1: Casos de uso del negocio

Los diagramas de casos de uso del negocio identifican los casos de uso del negocio, los actores del negocio y sus relaciones. Un caso de uso del negocio modela los servicios que el negocio ofrece a los actores del negocio, y es independiente de las tecnologías. Una estrategia que permite evitar la consideración de funcionalidades de sistema en los casos de uso del negocio consiste en considerar que el negocio implementa sus servicios de manera no computarizada (Metzner y Niño, 2016).

En la Figura 2.1 se muestra el trabajador del negocio Técnico que es quien inicia y se beneficia del caso de uso del negocio «Registrar datos de cultivos».

### 2.1.5. Diagrama de actividades de los casos de uso del negocio

Los Diagramas de actividad de los casos de uso del negocio describen las acciones y los resultados asociados a un flujo de eventos de un caso de uso del negocio. Se usan principalmente para modelar el flujo de trabajo y es útil para analizar los casos de uso describiendo las acciones que necesitan realizarse, cuándo se realizan y quién es el responsable de realizarlas sin dar el detalle de cómo cooperan o colaboran los objetos (Metzner y Niño, 2016).

El diagrama de actividades se muestra en la figura 2.2, mostrando cómo ocurre el proceso del negocio.

### 2.1.6. Diagrama de clases del modelo de objetos

El Diagrama de clases del modelo de objetos describe cómo cada caso de uso del negocio es llevado a cabo por parte de un conjunto de trabajadores que utilizan un conjunto de entidades

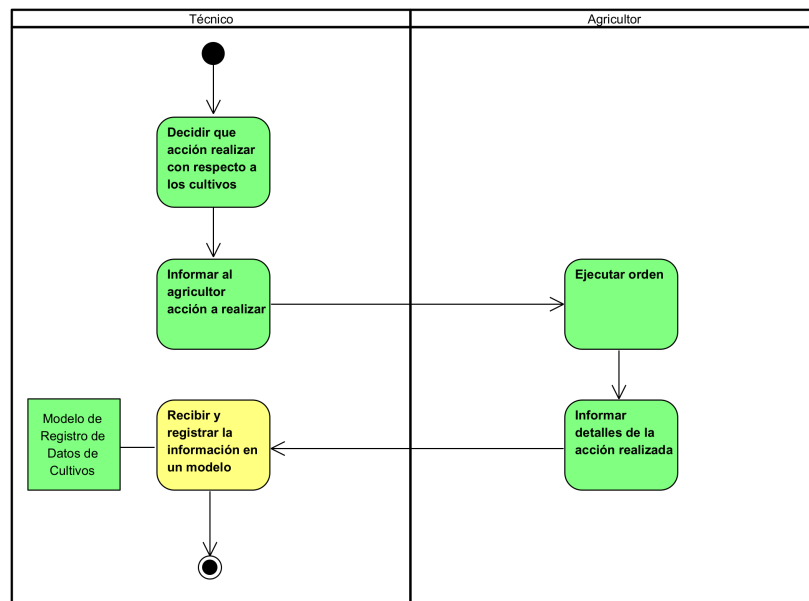


Figura 2.2: Diagrama de actividades

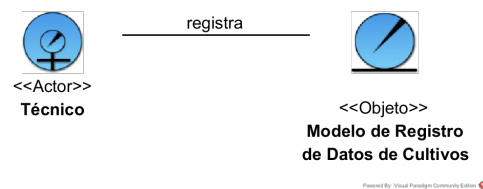


Figura 2.3: Modelo de objetos

del negocio y de unidades de trabajo. El mismo presenta un sistema (el negocio) desde la perspectiva de su uso y esquematiza cómo proporciona valor a sus usuarios (Edith, 2019). El mismo identifica las clases del negocio y las relaciones entre ellas. Se corresponde con la estructura de la organización y de la información. Es de destacar que el modelado orientado a objetos específicamente modela el comportamiento. Los objetos existen en un sistema computacional y están sujetos a restricciones y acciones del sistema, poseen un ciclo de vida: se crean y se destruyen, pueden ser asignados a otros objetos y tienen un comportamiento (Metzner y Niño, 2016).

En la Figura 2.3 se muestra el modelo de objetos que pertenece al presente proyecto.

### 2.2. Necesidades y cualidades del sistema

Las necesidades y cualidades del sistema establecen qué tiene que hacer exactamente el sistema que construyamos. Son el contrato que se debe cumplir, de modo que los usuarios finales tienen que comprender y aceptar los requisitos que se especifiquen. Los establecen los usuarios finales y participantes en los procesos ([Edith, 2019](#)).

Para definir correctamente las necesidades del sistema se debe:

- Establecer un acuerdo entre clientes y otros *stakeholders* sobre la funcionalidad del sistema.
- Proveer a los desarrolladores un mejor entendimiento de los requisitos del sistema.
- Definir el ámbito del sistema.
- Proveer una base para la planeación de los contenidos técnicos de las iteraciones.
- Proveer una base para estimar costos y tiempo de desarrollo del sistema.
- Definir una interfaz de usuarios para el sistema, enfocada a las necesidades y metas del usuario.

Clasificación general de los requerimientos del sistema:

- Funcionales
- No Funcionales

#### 2.2.1. Requerimientos funcionales

Los requerimientos funcionales representan la funcionalidad del sistema, y se modelan mediante casos de uso([Edith, 2019](#)). Para la identificación de los requerimientos funcionales del sistema se consideran las especificaciones de los casos de uso del negocio ([Metzner y Niño, 2016](#)).

Después de realizar las entrevistas con los usuarios del sistema quedaron definidos los principales requisitos funcionales que se muestran en la Tabla 2.3(ver Anexo A para consultar todos los requerimientos funcionales).

## 2.2 Necesidades y cualidades del sistema

Tabla 2.3: Requisitos funcionales

Nº	Requerimientos
R 1	Autenticar usuarios
R 2	Cambiar contraseña
R 3	Gestionar usuarios
R 4	Gestionar modelos de invernaderos
R 5	Gestionar invernaderos
R 6	Gestionar áreas de cultivos
R 7	Gestionar tipos de suelo
R 8	Gestionar tipos de cultivos
R 9	Gestionar especies
R 10	Gestionar variedades
R 11	Gestionar cultivos
R 12	Gestionar actividades fito-técnicas
R 13	Gestionar fertilizaciones orgánicas
R 14	Gestionar fertilizaciones químicas
R 15	Gestionar fertilizaciones biológicas
R 16	Gestionar controles de maleza manual
R 17	Gestionar controles de maleza mecánicos
R 19	Gestionar riegos
R 20	Gestionar podas
R 21	Gestionar aporques
R 22	Gestionar drenajes
R 23	Gestionar desbotones
R 24	Gestionar controles mecánicos
R 25	Gestionar controles químicos
R 26	Gestionar controles biológicos
R 27	Gestionar cosechas
R 28	Mostrar listado de actividades fito-técnicas realizadas a un cultivo
R 29	Mostrar listado de actividades fito-sanitarias realizadas a un cultivo
R 30	Cerrar sesión

### 2.2.2. Requerimientos no funcionales

Los requerimientos no funcionales son los atributos que debe exhibir el sistema como facilidad de uso, fiabilidad, eficiencia, portabilidad, etc (Edith, 2019). Debe pensarse en estas propiedades como las características que hacen al producto atractivo, usable, rápido y confiable.

#### Apariencia o interfaz externa

- La interfaz estará diseñada de modo tal que el usuario pueda tener en todo momento el control de la aplicación, lo que le permitirá ir de un punto a otro dentro de ella con gran facilidad. Se cuidará porque la aplicación sea lo más interactiva posible.

#### Usabilidad

- La aplicación web podrá ser usada por aquellos usuarios que no tengan experiencia en el uso de la computadora, sólo necesitarían un ligero entrenamiento sobre el funcionamiento de los principales elementos de una interfaz estándar en el ambiente de los Sistemas Operativos Windows (uso del mouse, manejo de menús, botones, cuadros de texto, entre otros.)

#### Rendimiento

- No se requiere de una capacidad de procesamiento alta, pues la aplicación no ejecutará algoritmos complejos.

#### Soporte

- Se requiere un servidor de bases de datos con soporte de volúmenes medianos de información. Se documentará la aplicación para garantizar su soporte. Se realizará mantenimiento a fin de aumentar las funcionalidades del mismo a través de versiones posteriores y según las nuevas necesidades de los clientes.

#### Portabilidad

- El producto podrá ser utilizado sobre plataforma Windows, Linux u otro sistema operativo. La estandarización del protocolo de TCP/IP y HTTP permite la interacción del lado del cliente para los sistemas operativos más difundidos como los Sistemas GNU/Linux (Debian, Ubuntu, Nova, etc.), Windows o MacOS.

#### Seguridad



## 2.2 Necesidades y cualidades del sistema

---

- Debe garantizar la conectividad e integridad de los datos almacenados a través de la red usando el protocolo de comunicación HTTPS y el SGBD respectivamente. Debe garantizar la confidencialidad para proteger la información de acceso no autorizado. Esto estará garantizado por el Sistema Gestor de Base de Datos. El sistema impondrá un estricto control de acceso que permitirá a cada usuario tener disponible solamente las opciones relacionadas con su actividad. La información deberá estar disponible a los usuarios en todo momento, limitada solamente por las restricciones que estos tengan de acuerdo con la política de seguridad del sistema.

### **Integridad**

- La información manejada está protegida contra la corrupción y los estados inconsistentes pues los mecanismos de validación y el administrador del sistema se encargarán de que los datos entrados sean confiables, de calidad y salvado para los casos de errores.

### **Disponibilidad**

- Los usuarios tienen garantizado el acceso a la información sin ningún inconveniente y al mismo tiempo.

### **Requisitos legales**

- La herramienta propuesta responderá a los intereses de la Empresa La Quinta.

### **Confiabilidad**

- La aplicación en caso de fallos garantiza que las pérdidas de información sean mínimas y los datos almacenados no se pierden ni se modifiquen ya que los mismos solo son modificados cuando se confirma la acción requerida.

### **Software**

- Del lado del cliente se espera un sistema que funcione en un navegador que interprete las funciones básicas de JavaScript, css3 y html5 como Google Chrome.

### **Hardware**

- Las computadoras situadas en los puestos de trabajo de los usuarios requerirán como mínimo un procesador Pentium IV, 512 Mb de memoria RAM. Estas máquinas deben estar conectadas en red con el servidor.

## 2.2 Necesidades y cualidades del sistema

Tabla 2.4: Actores del sistema

Actores del sistema	Descripción
Técnico	Encargado de gestionar todos los módulos de gestión disponibles en la aplicación a excepción de los usuarios.
Administrador	Encargado de la gestión de usuarios y el resto de módulos de gestión de la aplicación.

- Como servidor se requerirá un computador con un procesador Pentium IV, 2 Gb de memoria RAM y al menos 100 Gb de disco duro.

### 2.2.3. Modelo de casos de uso del sistema

Un actor del puede corresponderse con un actor del negocio, en caso de que el actor del negocio acceda al sistema (Metzner y Niño, 2016).

En la tabla 2.4 se muestran los actores del sistema, identificados luego de analizar todos los usuarios que tendrán responsabilidades a realizar dentro del mismo.

Un caso de uso del sistema describe lo que debe hacer el sistema para automatizar uno o más pasos de la realización del caso de uso de negocio. Se representan los casos de uso del sistema, los actores del sistema y las relaciones entre los casos de uso y sus actores (Metzner y Niño, 2016).

La figura 2.4 muestra los casos de usos del sistema que son importante para la arquitectura del mismo.

Para una mejor comprensión de los requerimientos solicitados por el usuario, se describe en la Tabla 2.5 el caso de usos más importante de la solución propuesta como parte de esta investigación.

Tabla 2.5: Descripción de los casos de uso del sistema

Nombre del caso de uso	Gestionar cultivos
------------------------	--------------------

## 2.2 Necesidades y cualidades del sistema

---

Tabla 2.5: Descripción de los casos de uso del sistema

Actores	Técnico, Administrador
Propósito	Visualizar todos los cultivos actuales, añadir y eliminar cultivos, así como actualizar los datos de los ya insertados en caso que sea necesario.
Resumen	El caso de uso se inicia cuando el administrador o el técnico necesita adicionar un nuevo cultivo. Una vez adicionado será posible visualizarlo en el listado de cultivos disponibles así como editar los datos o eliminarlo.
Referencias	R11., R11.1, R11.2, R11.3, R11.4
Precondiciones	Debe existir un área de cultivo insertada anteriormente. Debe existir una variedad insertada anteriormente.
Pos-condiciones	
Curso normal de eventos	
Acción del actor	Respuesta del sistema
1. El técnico desea adicionar, modificar o eliminar un cultivo	2. El sistema ejecuta alguna de las siguientes acciones: 2.1. Si el técnico escoge la opción de adicionar un nuevo cultivo en el sistema se ejecuta el CA1 2.2. Si escoge la opción de modificar los datos de un cultivo se ejecuta el CA2 2.3. Si escoge la opción de eliminar un cultivo existente se ejecuta el CA3
Flujo alternativo	CA1. Insertar

## 2.2 Necesidades y cualidades del sistema

---

Tabla 2.5: Descripción de los casos de uso del sistema

2. El técnico escoge la opción adicionar cultivo.	1. El sistema muestra la opción adicionar cultivo.
4. El técnico llena los datos del cultivo y escoge la opción adicionar.	3. El sistema solicita los datos del cultivo. 5. El sistema adiciona los datos del cultivo en la base de datos y muestra mensaje de información “Cultivo adicionado correctamente”.
Flujo alternativo	CA2. Modificar
2. El técnico escoge la opción modificar. 4. El técnico confirma la opción de modificar.	1. El sistema muestra los cultivos existentes. 3. El sistema activa los campos editables del cultivo. 5. El sistema actualiza los datos modificados del cultivo en la base de datos y muestra mensaje de información “Cultivo modificado correctamente”.
Flujo alternativo	CA3. Eliminar
2.El técnico selecciona el cultivo que desea eliminar . 4.El técnico confirma la opción de eliminar(E1) .	1. El sistema muestra el listado de cultivos existentes. 3. El sistema solicita la confirmación para eliminar el cultivo seleccionado. 5. El sistema elimina el cultivo de la base de datos y muestra mensaje de información “Cultivo eliminado correctamente”.

---

Tabla 2.5: Descripción de los casos de uso del sistema

<b>E1</b>	Excepción 1: Si el cultivo no puede ser eliminado, se muestra un mensaje de error "El cultivo no pudo ser eliminado"
<b>Prioridad</b>	Crítico

### 2.3. Análisis y diseño del sistema

Esta disciplina define la arquitectura del sistema y tiene como objetivos trasladar requisitos en especificaciones de implementación, al decir análisis se refiere a transformar casos de uso en clases, y al decir diseño se refiere a refinar el análisis para poder implementar los diagramas de clases de análisis de cada caso de uso, los diagramas de colaboración, el de clases de diseño, el de secuencia de diseño, el de estados de las clases, y el modelo de despliegue de la arquitectura (Chacón, 2006).

El diseño de software es quizás la etapa más importante y definitoria del proceso de desarrollo de software para que el producto que se obtenga sea de calidad. El propósito del diseño se puede resumir en transformar los requisitos en un diseño que describa cómo el sistema debe ser, desarrollar una robusta arquitectura del sistema y adaptar el diseño para que se corresponda con el entorno de implementación, diseñando sus funcionalidades. Debe estar bien especificado para que el sistema pueda ser implementado sin imprecisiones (Descripción et al., 2014).

En el modelo de análisis se analizan y refinan los requerimientos del modelo de casos de uso para obtener una visión detallada de los requerimientos del sistema. El modelo de análisis se describe en un lenguaje para desarrolladores y proporciona una visión general y conceptual del sistema respecto a lo que se tiene que hacer y no cómo se va a hacer. Por este motivo es que es un modelo útil y conveniente ya que facilita comprender el sistema sin mostrar detalles de alternativas de diseño que pueden variar y están atadas al entorno de implementación. El modelo de análisis se considera como una versión inicial del modelo de diseño, y el modelo de diseño especifica el diseño detallado de las clases (Metzner y Niño, 2016).

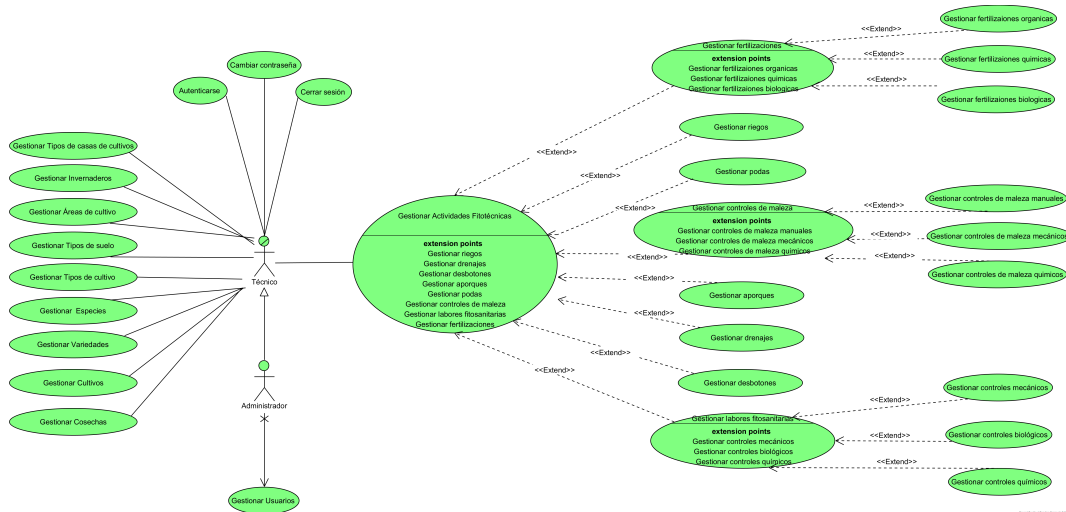


Figura 2.4: Modelo de casos de uso del sistema

### 2.3.1. Diagramas de Colaboración

Un diagrama de colaboración muestra la interacción entre los objetos resaltando la organización estructural de los objetos en lugar del orden de los mensajes intercambiados (Chacón, 2006).

Es una forma alternativa de representar los mensajes intercambiados entre los objetos, aquí se enfatiza en los objetos en si y su interrelación, aunque siguiendo la secuencia de los mensajes también se puede entender los pasos del caso de uso.

En la Figura 2.5 se puede ver el diagrama de colaboración del caso de uso "Gestionar cultivos" en el escenario "Insertar cultivo".

En las en el Anexo B, específicamente en las Figuras B.1, B.2 y B.3, se pueden ver el resto de los escenarios para el mismo caso de uso "Gestionar Cultivos".

### 2.3.2. Diagramas de Clases del Diseño

Los diagramas de clases del diseño posibilitan un mejor entendimiento de la forma en que se encuentra estructurado el sistema, cada clase del diagrama cuenta con atributos y métodos bien definidos lo que facilita al programador el desarrollo de la aplicación (Descripción et al., 2014).

## 2.3 Análisis y diseño del sistema

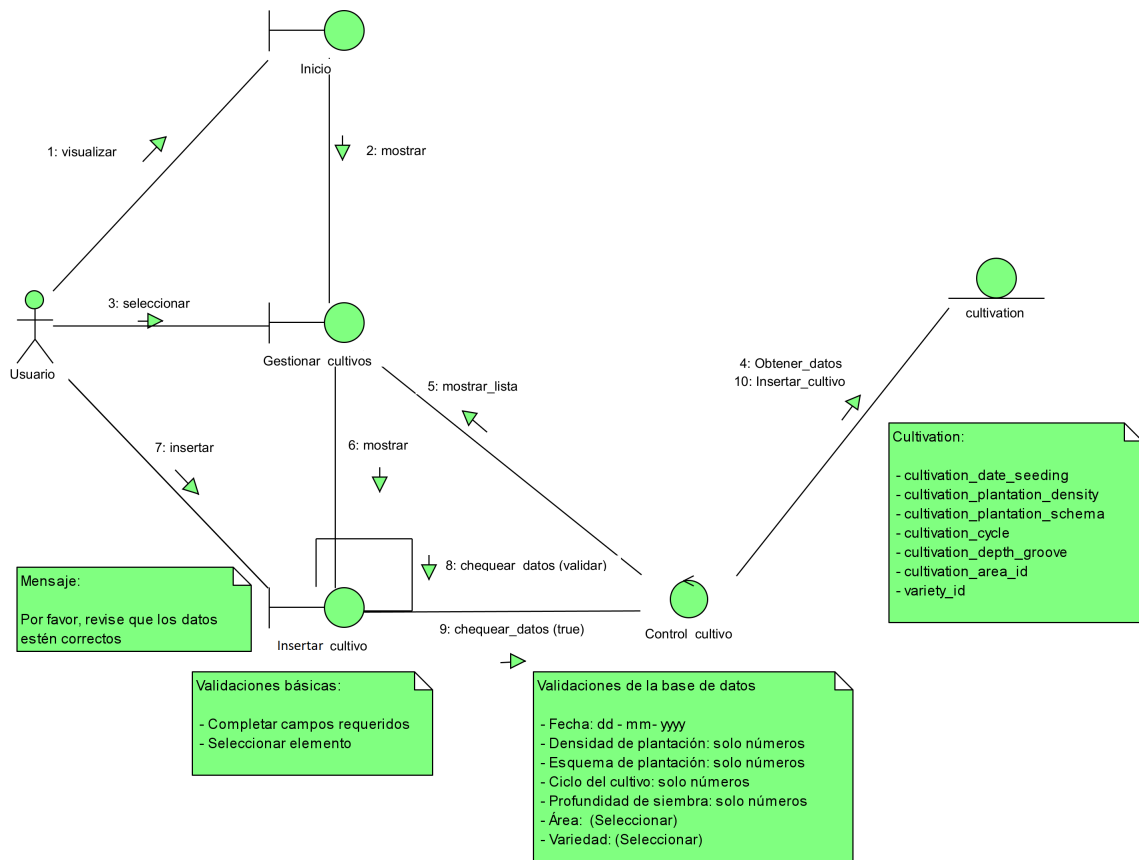


Figura 2.5: Diagrama de colaboración «Gestionar cultivos». Escenario «Insertar cultivo»

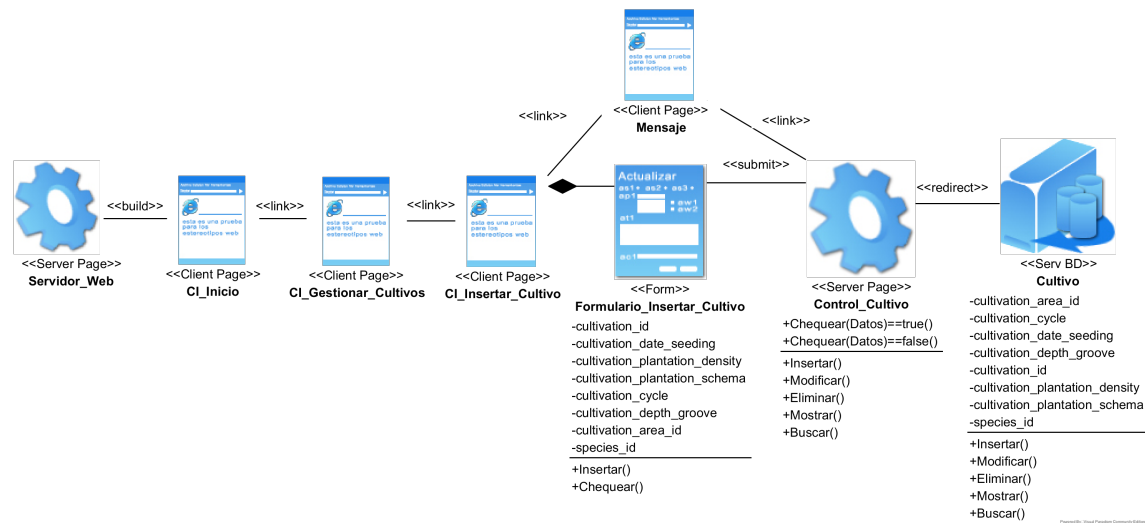


Figura 2.6: Diagrama de diseño «Insertar cultivo»

En los diagramas de clase se especifican propiedades de las clases entidad, interfaz y control: tipo de datos y visibilidad de los atributos y operaciones especificados en las clases del modelo de análisis. Una operación especificada en una clase del modelo de análisis se convierte en uno o más métodos en el modelo de diseño. Se analiza si es posible generalizar operaciones de las clases de análisis. Atributos especificados en una clase de análisis se pueden convertir en clases de diseño (Metzner y Niño, 2016).

En este caso, el diagrama de clases de diseño para el caso de uso "Gestionar cultivos" en el escenario "Insertar cultivo" se puede ver en la figura 2.6.

En las en el Anexo B, específicamente en las Figuras B.4, B.5 y B.6, se pueden ver el resto de los escenarios para el mismo caso de uso "Gestionar Cultivos".

## 2.4. Diseño de la base de datos

En la Figura 2.7 se presenta el diagrama entidad-relación del sistema propuesto.

Todos los datos de la aplicación serán almacenados en una misma base de datos estructurada por tablas. En la figura 2.8 se muestra el diagrama físico de la base de datos.



## 2.4 Diseño de la base de datos

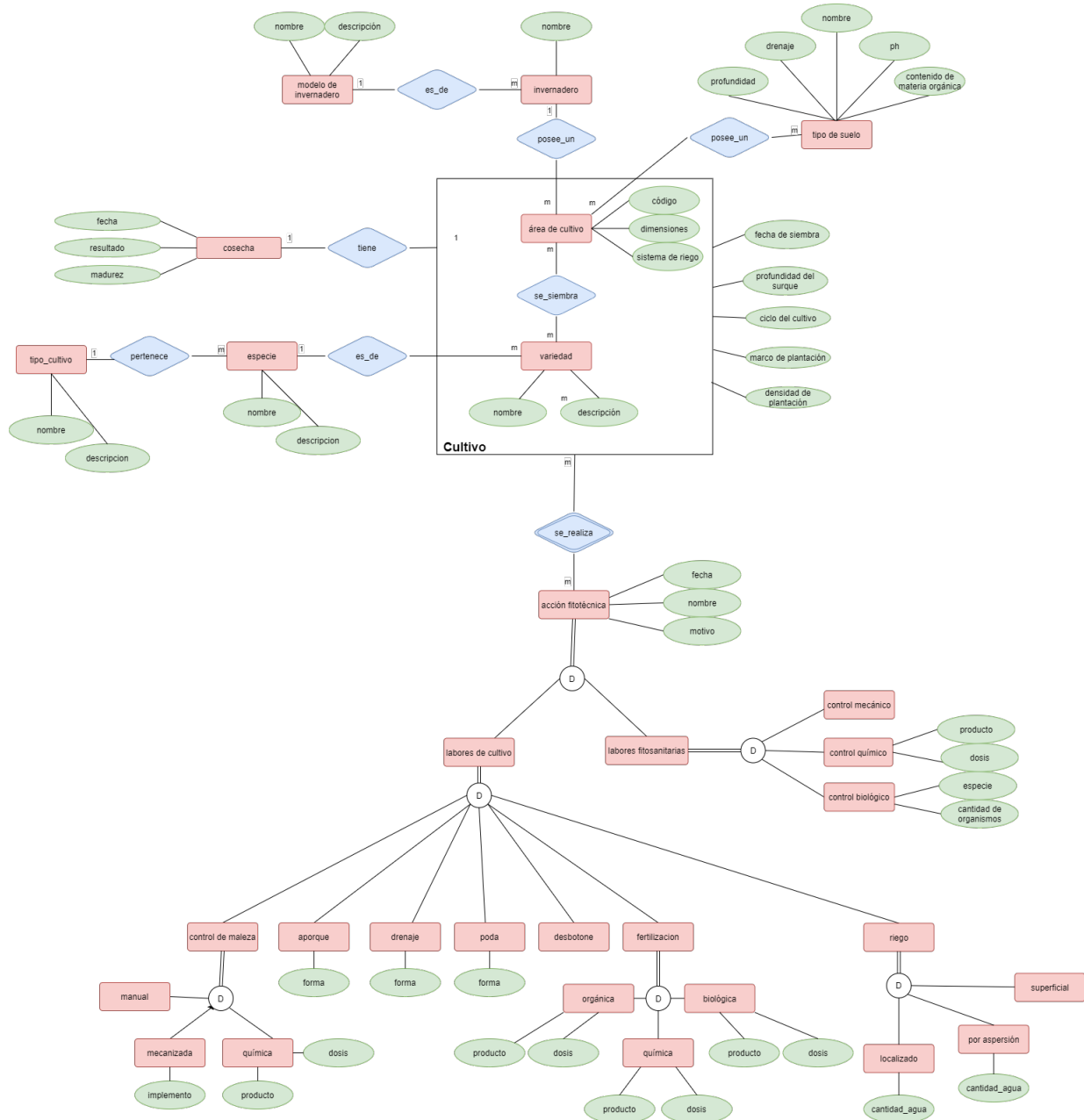


Figura 2.7: Modelo de entidad-relación

## 2.5 Conclusiones parciales

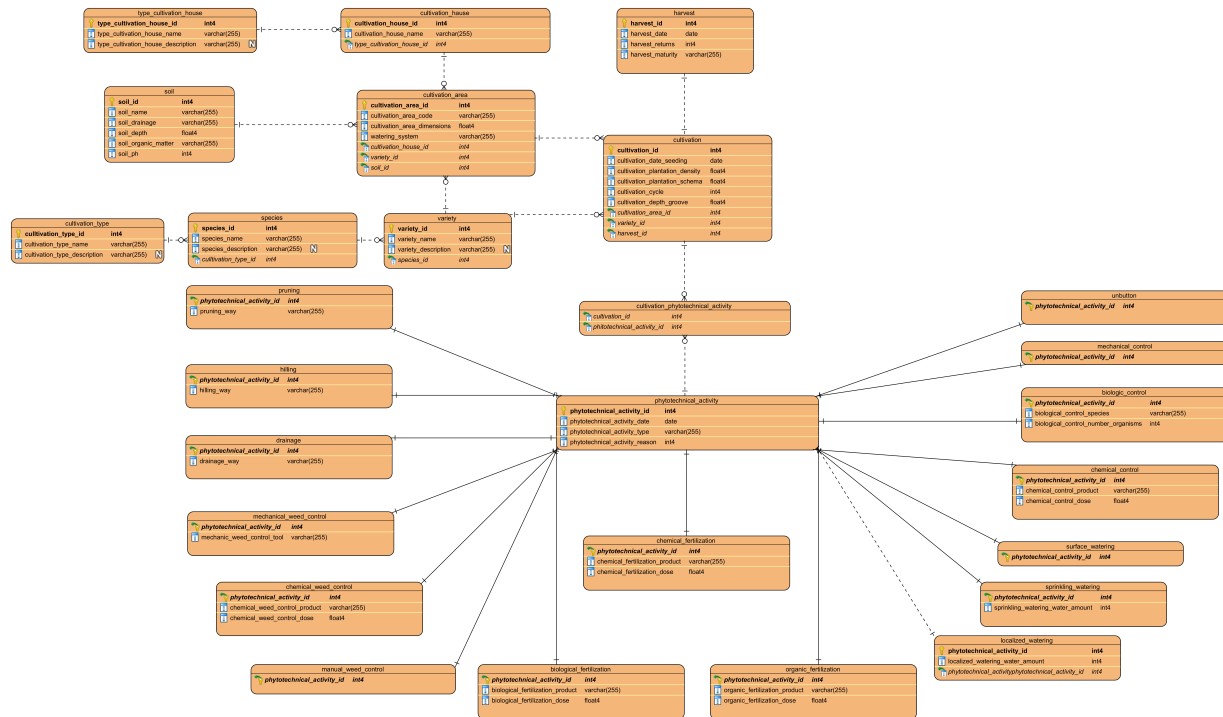


Figura 2.8: Modelo físico de la base de datos

## 2.5. Conclusiones parciales

La utilización de la metodología RUP permitió comprender cómo funcionan el negocio y el sistema a través de sus artefactos, . Además, permitió identificar los requerimientos funcionales, las reglas y los procesos del negocio. La construcción de todos estos artefactos propició que se esclareciera cómo funciona el flujo de eventos que se realiza en cada uno de los procesos del negocio y se establecieron las relaciones de cada uno de los actores del sistema con las diferentes funcionalidades a implementar. A partir de ello se logró diseñar y desarrollar una aplicación web para contribuir a la gestión de datos asociados a los cultivos de la Empresa La Quinta.

# CAPÍTULO 3

## **Validación del correcto funcionamiento de una aplicación web para la gestión de los datos asociados a los cultivos de la empresa La Quinta**

En este capítulo se especifica el proceso de desarrollo de la aplicación web mediante la utilización de la metodología RUP. Además, se especifican los aspectos de seguridad que se tuvieron en cuenta, diseño de interfaz y tratamiento de errores. A continuación se muestran las principales interfaces y funcionalidades del prototipo inicial, así como el diagrama de despliegue. Para finalizar, se realiza la validación de la solución propuesta a través de diferentes pruebas realizadas al software.

### **3.1. Ayuda, tratamiento de errores y seguridad**

Al realizar una aplicación web para determinado cliente, se debe entregar un producto con la calidad requerida, y para ello es necesario tener en cuenta varios aspectos. Los principales son la ayuda del sistema, el tratamiento de los errores y la seguridad. A continuación se evidencia el manejo de cada uno de esos elementos en la presente aplicación web.

### 3.1.1. Ayuda

La aplicación web cuenta con una sección de ayuda disponible para todos los usuarios del sistema. En la misma se explica como acceder y manejar cada una de las funcionalidades del sistema. Esta opción se encuentra disponible en todo momento.

### 3.1.2. Tratamiento de errores

Uno de los problemas más frecuentes en la ejecución de un sistema surge con las numerosas situaciones de error que pueden existir producto a variables que no pueden ser controladas por el sistema. Frente a situaciones como estas es necesario tomar precauciones para evitar cualquier error emergente que surja y tratarlo de la forma correcta ([Descripción et al., 2014](#)).

Durante la fase de implementación de la aplicación web, se tomó en consideración evitar la mayor cantidad de excepciones o errores posibles. Se aprovecharon para ello los potenciales del *framework* Django, y se validó que la información introducida y gestionada por los usuarios tuviese los formatos correctos procurando evitar la generación de excepciones.

En caso de que lo anterior no pudiera realizarse, se proporcionaron mensajes de error que siguen las siguientes normas:

- Escribir los mensajes de error con un lenguaje comprensible para el usuario.
- Ser específico en los mensajes de error acerca de cual es el error y como solucionarlo.
- Utilizar en todos los mensajes el mismo formato para lograr una congruencia entre los diferentes errores y así facilitar la detección intuitiva de la causa de los mismos.
- No culpar al usuario del problema ocurrido.

### 3.1.3. Seguridad

La seguridad del sistema se maneja a través de la autenticación de usuarios. Primeramente el usuario debe registrarse proporcionando los datos requeridos de manera correcta, a continuación podrá iniciar sesión accediendo a las opciones de la aplicación, y finalmente podrá cerrar sesión una vez terminado su trabajo.

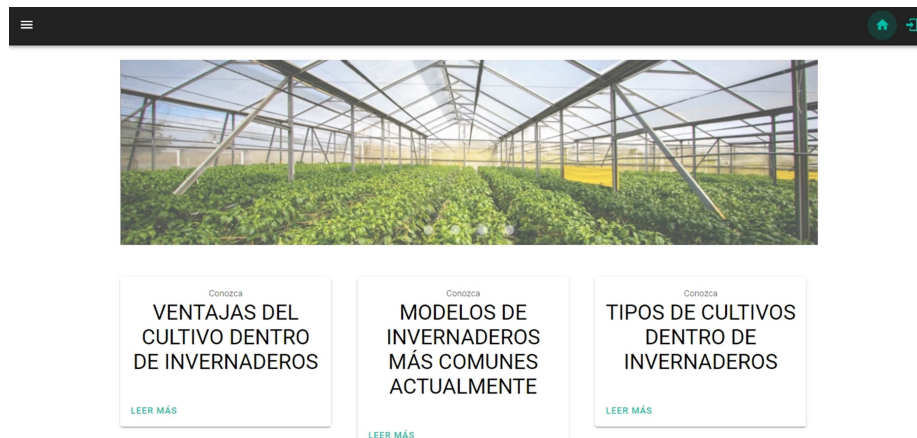


Figura 3.1: Página de inicio

El usuario Administrador será el único capaz de acceder a la gestión de usuarios de la aplicación. Cada usuario podrá acceder a las opciones que le sean permitidas dentro de la aplicación en correspondencia con el rol que desempeñe su usuario.

## 3.2. Prototipos de interfaz de usuarios

Al iniciar la aplicación web el usuario podrá visualizar la interfaz correspondiente a la página inicial, la cual se muestra en la Figura 3.1. Una vez se ha iniciado sesión estará disponible una página inicial diferente, con algunos datos referentes al usuario en si. A partir de ahí le será posible a acceder a las opciones de gestión del sistema, algunos a partir de las opciones del *dashboard*, aunque todas se encuentran disponibles en la barra lateral.

Las interfaces principales siguen el mismo formato para una experiencia más cómoda del usuario. Para iniciar el manejo de los cultivos y cosechas el usuario deberá insertar datos de las plantas que serán cultivadas, así como de los espacios en que serán sembradas, ejemplo de ello los invernaderos con los que cuenta. La Figura 3.2 muestra el formulario que aparece al hacer clic en la opción de "Añadir" dentro de la interfaz "Gestionar invernaderos", y luego de añadirlo, en caso de haber cometido un error podrá eliminarse, acción que deberá ser confirmada como se muestra en la Figura 3.3.

Una vez insertados todos los datos requeridos, finalmente será posible insertar cultivos, rellenando

## 3.2 Prototipos de interfaz de usuarios

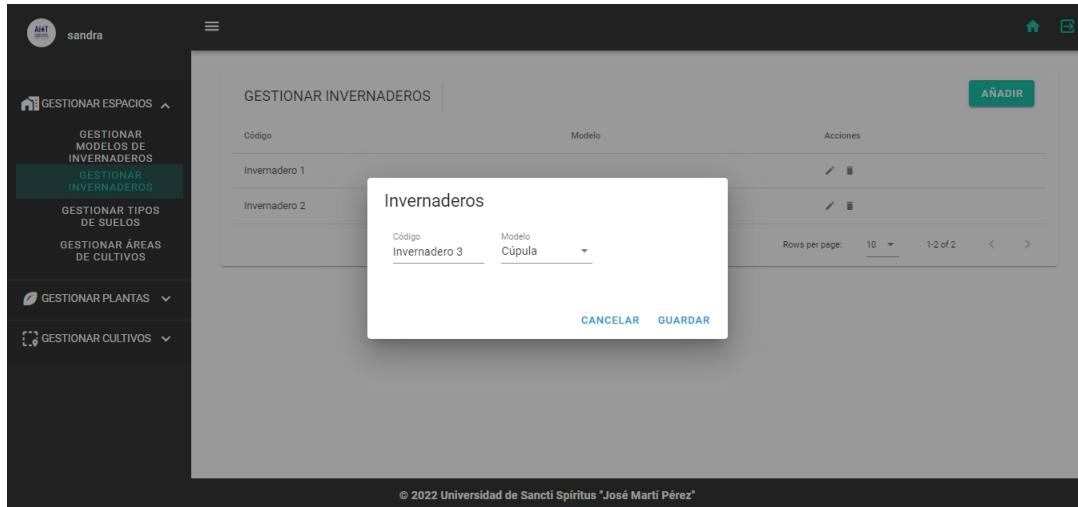


Figura 3.2: Interfaz "Insertar invernadero"

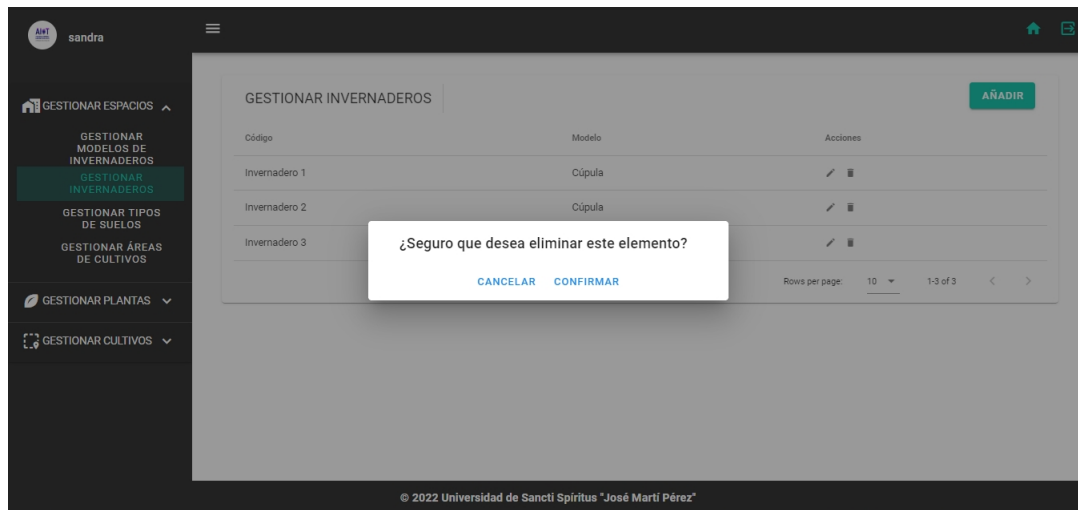


Figura 3.3: Interfaz "Eliminar invernadero"

## 3.2 Prototipos de interfaz de usuarios

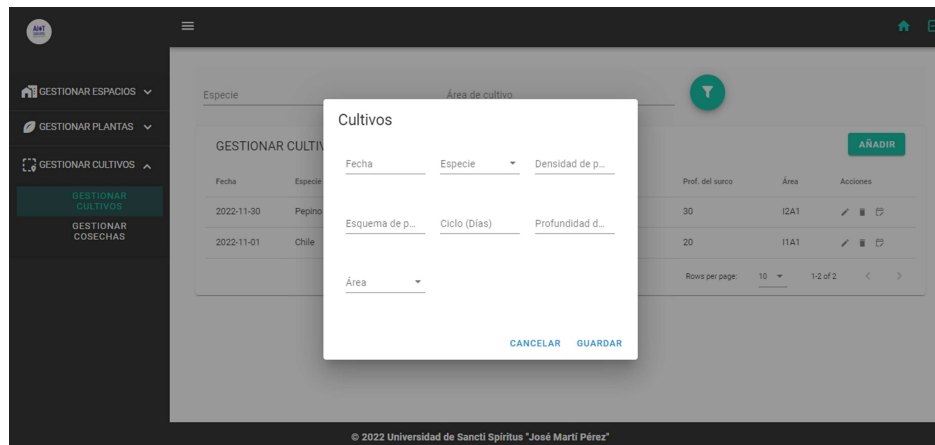


Figura 3.4: Interfaz "Insertar cultivos"

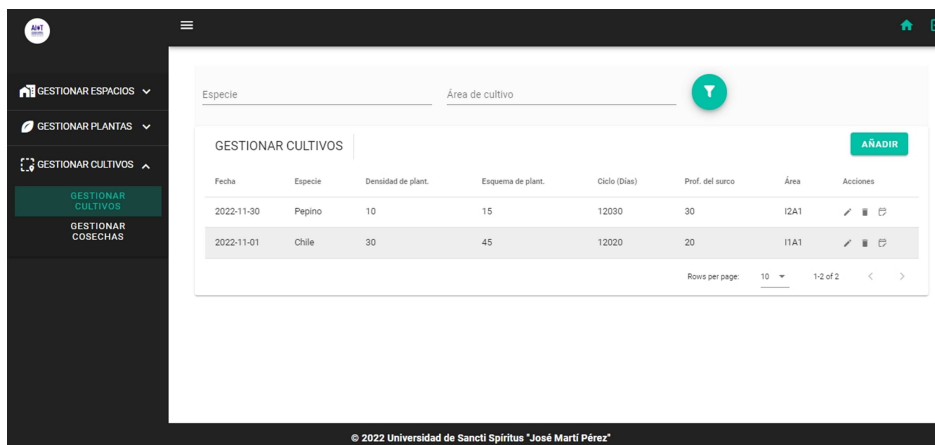


Figura 3.5: Interfaz "Gestionar cultivos"

do los campos que sean requeridos, y con los datos correctos como se aprecia en la Figura 3.4. Una vez insertado el cultivo podrá ser gestionado desde la interfaz que se muestra en la Figura 3.5 para acceder a las opciones de editar y eliminar.

Otra de las opciones disponibles para cada cultivo es insertar, editar y o eliminar las labores de cultivo o acciones fito-sanitarias que se le puedan realizar a los mismos. En la Figura 3.6 se muestra la interfaz que se hace presente cuando el usuario accede la opción "Acciones fitotécnicas" de un cultivo en específico. En esta interfaz se puede ver la opción para añadir las nuevas acciones realizadas al cultivo, en la Figura 3.7 se muestra el formulario correspondiente a añadir la nueva labor de cultivo "drenaje". Una vez insertado será visible en la tabla que aparece a la izquierda que contiene las acciones realizadas al cultivo en cuestión, y que solo muestra el

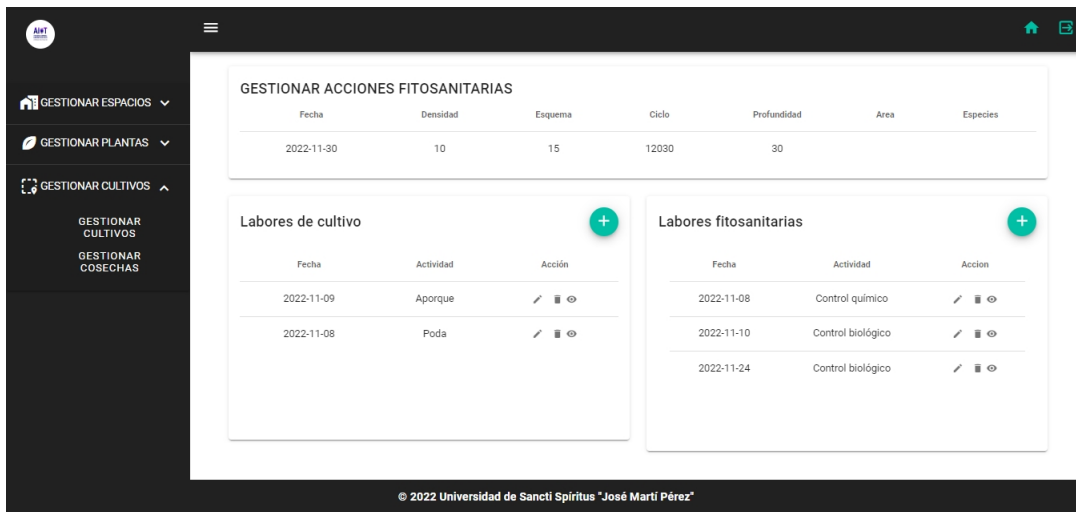


Figura 3.6: Interfaz "Gestionar acciones fitosanitarias"

nombre de la acción realizada junto a la fecha correspondiente, y al lado las opciones de editar y eliminar, además se puede ver una nueva opción para ver todos los datos de la acción en el icono que simula un ojo. Al hacer clic en la misma, como se ve en la Figura 3.8 se podrá visualizar una tarjeta con el resto de los datos de la acción.

### 3.3. Modelo de implementación

Según explica Chacón (2006), la implementación tiene como objetivos implementar las clases de diseño como componentes, asignar los componentes a los nodos, probar los componentes individualmente e integrar los componentes en un sistema ejecutable. Los artefactos para la implementación capturan y presentan la realización de la solución presentada en el análisis y diseño del sistema.

Se utiliza el modelo de implementación, conjuntamente los diagramas de componentes para comprender cómo se organizan los componentes y dependen unos de otros. El modelo de implementación es una colección de componentes, y de subsistemas de aplicación que contienen estos componentes.

El Diagrama de despliegue es un diagrama estructurado que muestra la arquitectura del sistema desde el punto de vista del despliegue (distribución) de los artefactos del software en los destinos



### 3.3 Modelo de implementación

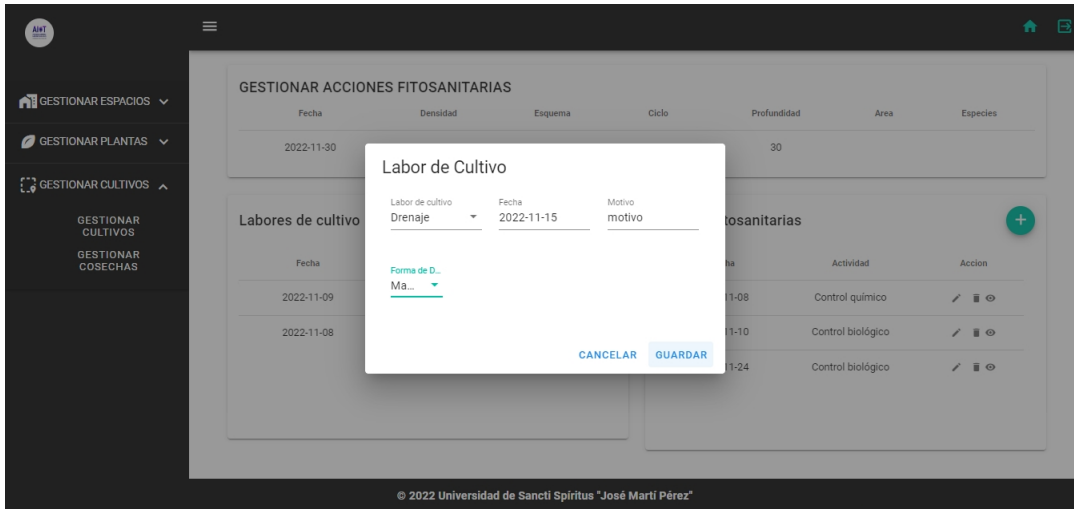


Figura 3.7: Interfaz "Insertar labor de cultivo (Drenaje)"

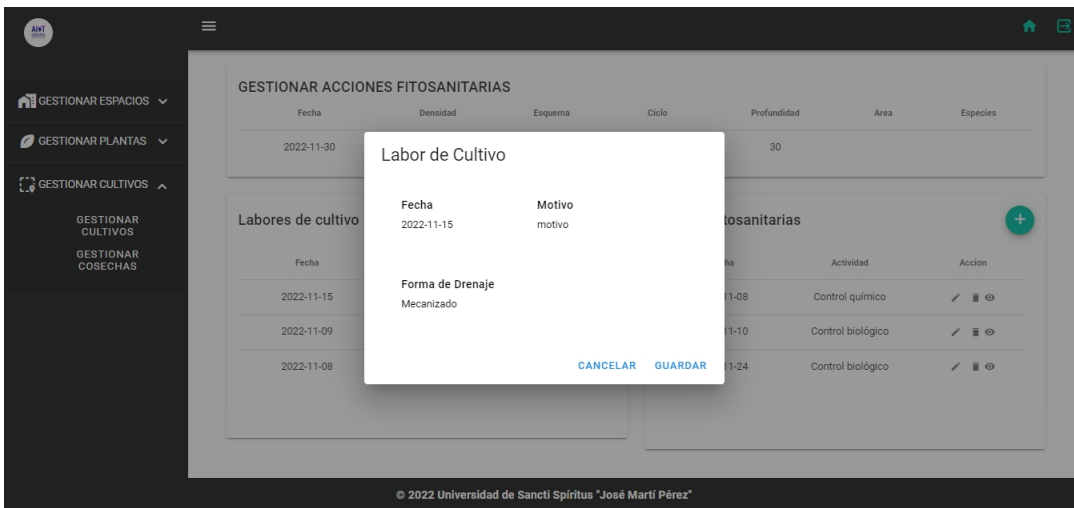


Figura 3.8: Interfaz "Mostrar labor de cultivo (Drenaje)"

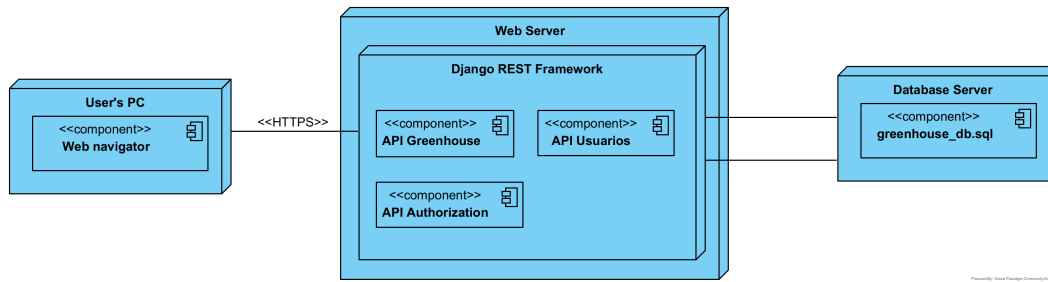


Figura 3.9: Diagrama de despliegue

de despliegue. Los artefactos representan elementos concretos en el mundo físico que son el resultado de un proceso de desarrollo. Ejemplos de artefactos son archivos ejecutables, bibliotecas, archivos, esquemas de bases de datos, archivos de configuración entre otros (Descripción et al., 2014).

La Figura 3.9 muestra el diagrama de despliegue correspondiente a la aplicación web. En el mismo se muestran los diferentes módulos, los cuales son el servidor de la base de datos, el *backend* y el *frontend*. Además se puede ver la computadora del usuario, donde será finalmente ejecutada la aplicación desde un navegador web.

## 3.4. Pruebas

### 3.4.1. Pruebas al *backend* utilizando Postman

Postman es una plataforma de API para crear y utilizar API. Postman simplifica cada paso del ciclo de vida de las API y agiliza la colaboración para que pueda crear mejores API, más rápido (Postman, 2022). A través de la misma se realizarán pruebas al software a nivel de *backend* para comprobar el correcto funcionamiento de las principales peticiones al caso de uso "Gestionar cultivos", así como al proceso de inicio de sesión.

La Figura 3.10 muestra el resultado devuelto luego de mostrar los cultivos insertados en la base de datos.

La Figura 3.11 muestra el resultado devuelto luego de insertar un cultivo con los parámetros correctos.

## 3.4 Pruebas

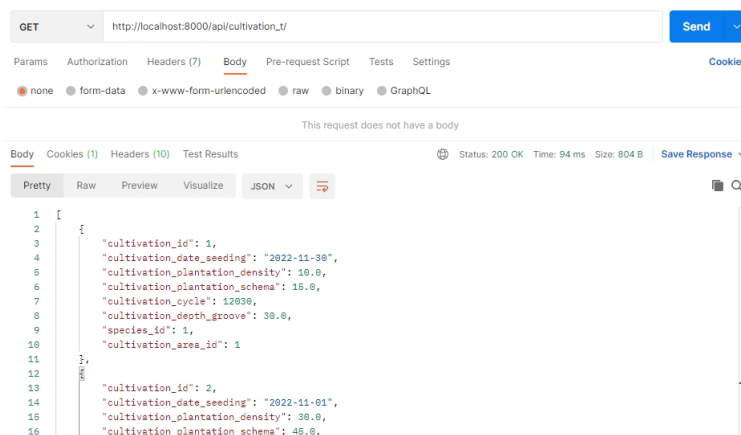


Figura 3.10: Respuesta a la petición "Mostrar cultivos"

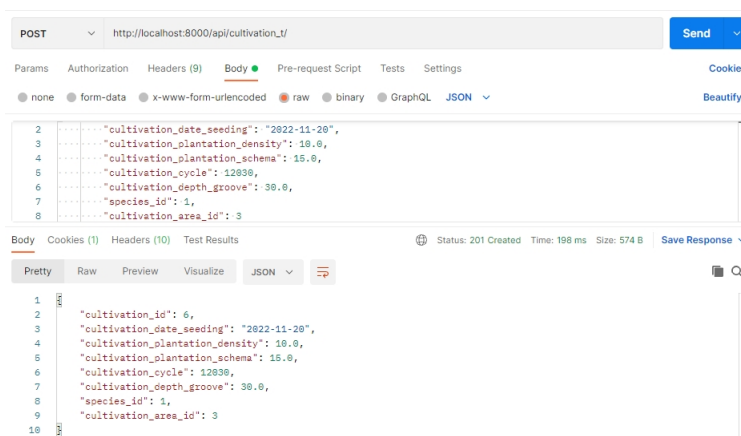


Figura 3.11: Respuesta a la petición "Insertar cultivo" proporcionando parámetros correctos

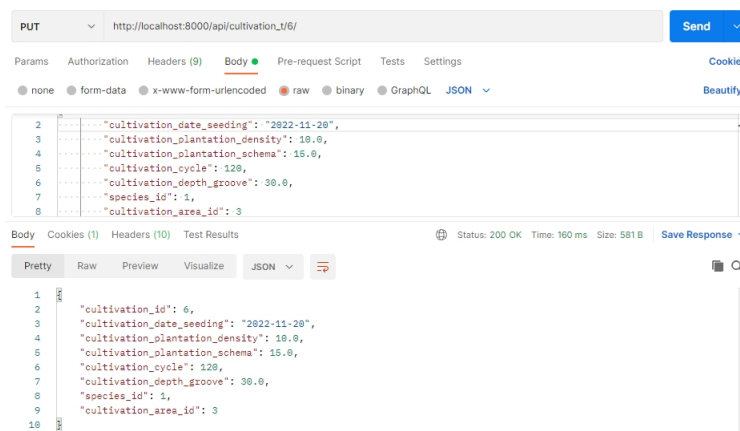


Figura 3.12: Respuesta a la petición "Modificar cultivo" proporcionando parámetros correctos

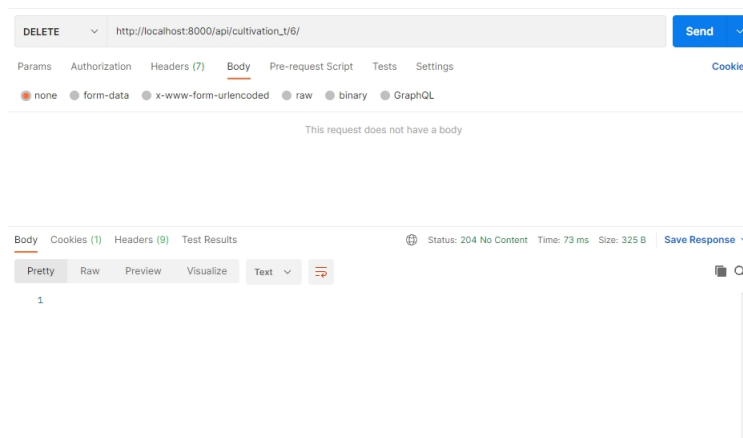


Figura 3.13: Respuesta a la petición "Eliminar cultivo"

La Figura 3.12 muestra el resultado devuelto luego de actualizar los datos de un cultivo con parámetros aceptados.

La Figura 3.13 muestra el resultado devuelto luego de eliminar un cultivo.

La Figura 3.14 muestra el resultado devuelto luego de un inicio de sesión exitoso.

### 3.4.2. Pruebas de rendimiento utilizando Firebug

Firebug es una extensión muy útil para desarrolladores web que permite examinar minuciosamente cada uno de los elementos de las páginas en busca de errores del código o fallos de

```

1  [OK]
2  {
3    "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ0b2Z1b198eXB1IjoIYWJkZm9uZSIsInR5cCI6IkpXVCJ9.eyJ0b2Z1b198eXB1IjoIYWJkZm9uZSIsInR5cCI6IkpXVCJ9",
4    "refresh-token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ0b2Z1b198eXB1IjoIYWJkZm9uZSIsInR5cCI6IkpXVCJ9",
5    "user": {
6      "username": "sandra",
7      "rol": "admin",
8      "email": "sandra@gmail.com",
9      "name": "sandra",
10     "last_name": "sandza"
11   },
12   "message": "Inicio de Sesión Exitoso"
}

```

Figura 3.14: Respuesta a la petición "Autenticar usuario"

```

GET http://localhost:8000/api/cultivation_t/ 200 OK 75ms
Headers Response
[{"cultivation_id":1,"cultivation_date_seeding":"2022-11-30","cultivation_plantation_density":10.0,"cultivation_plantation_schema":15.0,"cultivation_cycle":12030,"cultivation_depth_groove":30.0,"species_id":1,"cultivation_area_id":1}, {"cultivation_id":2,"cultivation_date_seeding":"2022-11-01","cultivation_plantation_density":30.0,"cultivation_plantation_schema":45.0,"cultivation_cycle":12020,"cultivation_depth_groove":20.0,"species_id":2,"cultivation_area_id":2}]
Object { data=[2], status=200, statusText=OK, ... }

```

Figura 3.15: Respuesta al caso de uso "Mostrar cultivos"

presentación. Por medio de Firebug es posible controlar cualquier cosa que ocurre dentro del navegador cuando carga cualquier página web. Permite examinar todos los elementos HTML y además los estilos CSS y los códigos de *scripting* en Javascript que se estén ejecutando.

Esta extensión se ha utilizado para realizar pruebas de rendimiento al software, mostrando el tiempo de respuesta de cada petición como se puede apreciar en la parte superior de cada figura.

La Figura 3.15 muestra la respuesta obtenida luego de mostrar los cultivos insertados en la base de datos.

La Figura 3.16 muestra los parámetros insertados para el caso de uso "Insertar cultivo".

La Figura 3.17 muestra la respuesta ante los parámetros insertados para el caso de uso "Insertar cultivo".

La Figura 3.18 muestra los parámetros insertados para el caso de uso "Modificar cultivo".

```

POST http://localhost:8000/api/cultivation_t/ 201 Created 123ms
Headers Post Response JSON
JSON
cultivation_area_id      "3"
cultivation_cycle       "120"
cultivation_date_seeding "2022-12-06"
cultivation_depth_groove "35"
cultivation_plantation_density "300"
cultivation_plantation_schema "200"
species_id              "2"
Source
{"cultivation_date_seeding":"2022-12-06","species_id":"2","cultivation_plantation_density":"300","cultivation_plantation_schema":"200","cultivation_cycle":"120","cultivation_depth_groove":"35","cultivation_area_id":"3"}

```

Figura 3.16: Parámetros insertados para el caso de uso "Insertar cultivo"

```

POST http://localhost:8000/api/cultivation_t/ 201 Created 123ms
Headers Post Response JSON
{"cultivation_id":7,"cultivation_date_seeding":"2022-12-06","cultivation_plantation_density":300.0,"cultivation_plantation_schema":200.0,"cultivation_cycle":120,"cultivation_depth_groove":35.0,"species_id":2,"cultivation_area_id":3}

```

Figura 3.17: Respuesta ante los parámetros insertados para el caso de uso "Insertar cultivo"

```

PUT http://localhost:8000/api/cultivation_t/7/ 200 OK 91ms
Headers Put Response JSON
JSON
cultivation_area_id      "3"
cultivation_cycle        120
cultivation_date_seeding "2022-12-06"
cultivation_depth_groove 30
cultivation_plantation_density 300
cultivation_plantation_schema 200
species_id               "1"
Source
{"cultivation_date_seeding":"2022-12-06","species_id":"1","cultivation_plantation_density":300,"cultivation_plantation_schema":200,"cultivation_cycle":120,"cultivation_depth_groove":30,"cultivation_area_id":"3"}

```

Figura 3.18: Parámetros insertados para el caso de uso "Modificar cultivo"

La Figura 3.19 muestra la respuesta ante los parámetros insertados para el caso de uso "Modificar cultivo".

La Figura 3.20 muestra la respuesta ante credenciales de inicio de sesión inválidos.

La Figura 3.21 muestra la respuesta ante credenciales de inicio de sesión correctos.

### 3.4.3. Pruebas de aceptación

La técnica de caja negra, se enfoca en probar el sistema sin tomar en cuenta la estructura interna del mismo, su objetivo es validar que las salidas sean las esperadas. Se centra en encontrar las circunstancias en las que el sistema no se comporta conforme a las especificaciones establecidas (CHIU, 2015).

Las pruebas de aceptación son un tipo de pruebas pertenecientes al modelo de caja negra, y son las pruebas de sistema por parte del cliente. El objetivo en este nivel de prueba es obtener el visto bueno del cliente, no se deberían encontrar defectos funcionales graves en el sistema. Es por ello que las pruebas de aceptación son realizadas por el usuario (CHIU, 2015).

En la Tabla 3.1 se muestran las clases válidas y no válidas para cada código de entrada en el caso de uso "Gestionar cultivos", así como la regla requerida por cada uno de ellos.

```

PUT http://localhost:8000/api/cultivation_t/7/ 200 OK 91ms
Headers Put Response JSON
{"cultivation_id":7,"cultivation_date_seeding":"2022-12-06","cultivation_plantation_density":300.0,"cultivation_plantation_schema":200.0,"cultivation_cycle":120,"cultivation_depth_groove":30.0,"species_id":1,"cultivation_area_id":3}
Object { data=Object, status=200, statusText="OK", ... }

```

Figura 3.19: Respuesta ante los parámetros insertados para el caso de uso "Modificar cultivo"

## 3.5 Conclusiones parciales



Figura 3.20: Respuesta ante credenciales inválidas para el caso de uso "Autenticar usuarios"



Figura 3.21: Respuesta ante credenciales correctas para el caso de uso "Autenticar usuarios"

La Figura 3.22 muestra el resultado obtenido luego de la ejecución utilizando las entradas propuestas.

En la Figura 3.23 se muestran algunos ejemplos de los mensajes de error o éxito de la aplicación. La Figura 3.23a muestra el resultado devuelto luego de insertar un cultivo utilizando datos erróneos. Además, en la Figura 3.23b se muestra el resultado devuelto luego de insertar un drenaje utilizando entradas correctas. Finalmente, la Figura 3.23c muestra el resultado devuelto luego de intentar iniciar sesión con credenciales incorrectas.

## 3.5. Conclusiones parciales

En este capítulo se definieron los principios de diseño seguidos en el sistema y el modelo de implementación. Fue creado el diseño de la interfaz de la aplicación y cómo se manejará la ayuda, el tratamiento de los errores, y seguridad en el mismo. Se representó el diagrama de componentes integrado al de despliegue. Finalmente se llevó a cabo las pruebas de caja negra

Tabla 3.1: Caso de prueba para el caso de uso "Gestionar cultivos". Escenario "Insertar cultivo"

Código de entrada	Tipo	Clases válidas	Clases no válidas	Regla
Fecha	Valor	aaaa-mm-dd(1)	Otro (2)	DateField
Densidad de plantación	Valor	0-9 (3)	Otro (4)	IntegerField
Esquema de plantación	Valor	0-9 (5)	Otro (6)	IntegerField
Ciclo del cultivo	Valor	0-9 (7)	Otro (8)	IntegerField
Profundidad del surco	Valor	0-9 (9)	Otro (9)	IntegerField
Especie	Valor	A-Z, a-z (11)	Otro (12)	CharField
Área de cultivo	Valor	A-Z, a-z (13) 0-9 (14)	Otro (15)	CharField

### 3.5 Conclusiones parciales

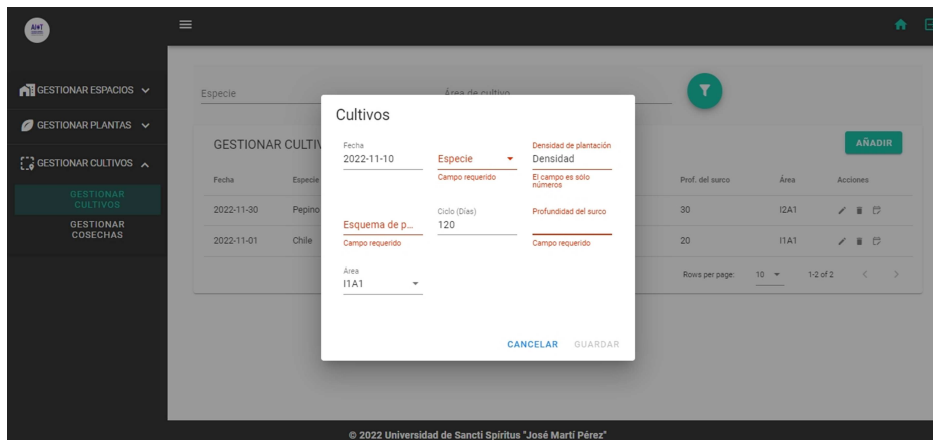
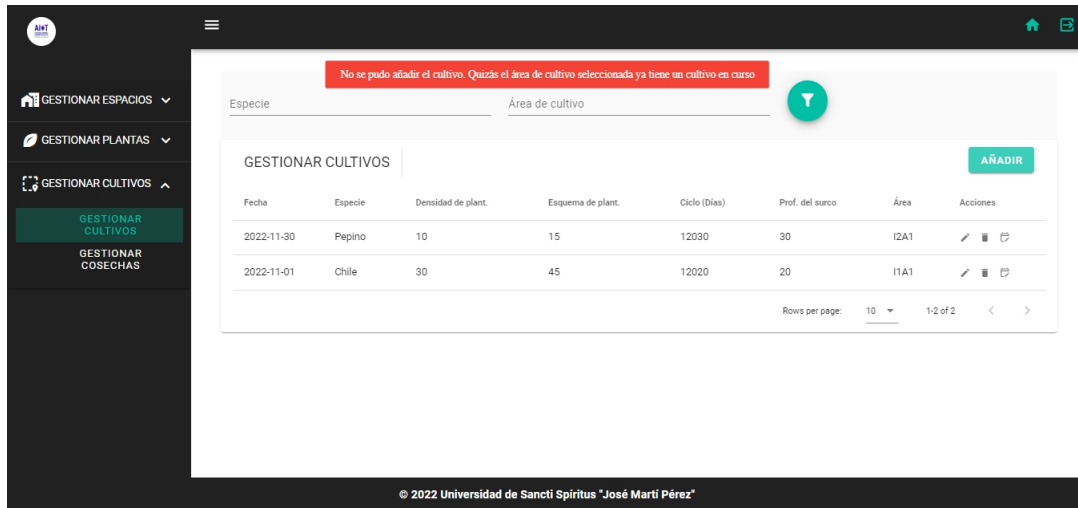


Figura 3.22: Resultado de la ejecución para las entradas propuestas

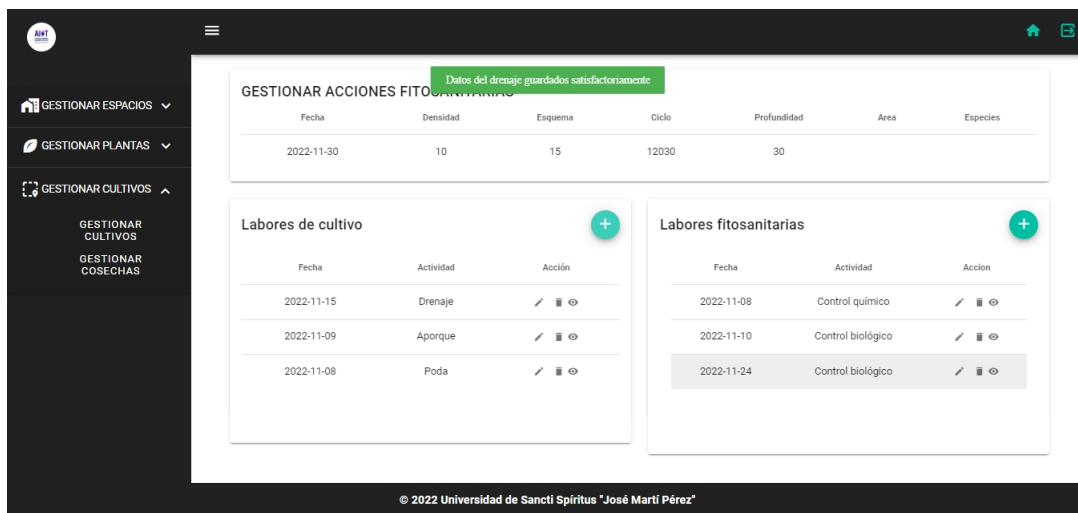
realizadas al caso de uso más significativo.



### 3.5 Conclusiones parciales



(a) Resultado erróneo "Insertar cultivo"



(b) Resultado satisfactorio "Insertar drenaje"



(c) Resultado erróneo "Iniciar sesión"

Figura 3.23: Ejemplos de diferentes mensajes de la aplicación

# CONCLUSIONES

A partir del desarrollo del presente proyecto, y de la investigación realizada para el desarrollo del mismo, se ha podido arribar a las siguientes conclusiones:

1. El estudio de los fundamentos teóricos y metodológicos para la elaboración del sistema permitió determinar que la metodología *RUP* es la adecuada para desarrollo del sistema. Además, permitió seleccionar el *framework* de desarrollo *Django* para el *backend* y el *framework Nuxt.js* vinculado al *framework* de componentes *Vuetify* para el *frontend*. También el análisis de diferentes sistemas gestores de base de datos permitió determinar a *PostgreSQL* como el adecuado. Se escogieron estas herramientas y tecnologías por ser de código libre y tener la característica de integrarse para funcionar en cualquier plataforma.
2. La utilización de la metodología *RUP*, permitió diseñar y desarrollar una aplicación web para contribuir a la gestión de datos asociados a los cultivos de la Empresa La Quinta, sustentado en las bases del software libre y la programación orientada a objetos, lo cual le brinda a la aplicación una mayor flexibilidad y capacidad de mantenimiento.
3. La utilización de pruebas de aceptación permitió validar el correcto funcionamiento de una aplicación web con funcionalidades que se ajustan a las necesidades del cliente y que brinda al usuario la posibilidad de interactuar de manera dinámica con los datos.

# RECOMENDACIONES

A partir del estudio realizado y de las conclusiones a las que se llegaron a través del presente trabajo se recomienda:

- Continuar con la investigación para la gestión de la información relativa al almacenamiento y venta de los productos luego de las cosechas.
- Extender la aplicación para incluir el monitoreo de variables ambientales y de suelo, que influyen en el desarrollo de los cultivos, mediante la utilización de soluciones de Internet de las Cosas (IoT).
- Procesar los datos almacenados con el uso de Inteligencia Artificial, lo que permitirá la extracción de conocimiento para el apoyo a la toma de decisiones en aras de optimizar la producción.

# REFERENCIAS

- Akhter, R. y Sofi, S. A. (2021). Precision agriculture using IoT data analytics and machine learning, *Journal of King Saud University-Computer and Information Sciences* . Publisher: Elsevier.
- Amondaray, L. R., Figueredo, F. M. P. y Castilla, M. M. G. (2018). Red de sensores inalámbricos para las casas de cultivos protegidos “San José”, p. 11.
- Braun, J. (2021). *Lateral Join for SQLite*, Master’s thesis, Universität Tübingen.
- Calcina Vilcapaza, M. (2022). Desarrollo de un sistema web utilizando el Framework LoopBack 4 y NuxtJS para la gestión y control de celulares corporativos de la UPeU - Campus Juliaca. Accepted: 2022-06-01T15:43:38Z Publisher: Universidad Peruana Unión.  
**URL:** <https://repositorio.upeu.edu.pe/handle/20.500.12840/5511>
- Chacón, J. C. R. (2006). Aplicación de la metodología RUP para el desarrollo rápido de aplicaciones basado en el estándar J2EE, *Guatemala:(tesis de grado) para obtener el título de ingeniería en ciencias y sistemas-Universidad de San Carlos de Guatemala* .
- CHIU, C. C. (2015). LAS PRUEBAS EN EL DESARROLLO DE SOFTWARE, *UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO. México: sn* .
- Cicioğlu, M. y Çalhan, A. (2021). Smart agriculture with internet of things in cornfields, *Computers & Electrical Engineering* **90**: 106982. Publisher: Elsevier.
- Ciesla, R. (2021). Java, C#, and Python 101, *Programming Basics* pp. 13–27. Publisher: Apress, Berkeley, CA.  
**URL:** [https://link.springer.com/chapter/10.1007/978-1-4842-7286-2\\_2](https://link.springer.com/chapter/10.1007/978-1-4842-7286-2_2)

- Cutting, V. y Stephen, N. (2021). Comparative Review of JAVA and Python, **21**(1): 6.
- da Silveira, F., Lermen, F. H. y Amaral, F. G. (2021). An overview of agriculture 4.0 development: Systematic review of descriptions, technologies, barriers, advantages, and disadvantages, *Computers and Electronics in Agriculture* **189**: 106405. Publisher: Elsevier.
- Descripción, C., Visor, N. y Impresora, N. (2014). 3.6. Diagrama de Despliegue., “*La manera cómo se presentan las cosas no es la manera como son; y si las cosas fueran como se presentan la ciencia entera sobraría.*” *Karl Marx* p. 44. Publisher: Universidad de las Ciencias Informáticas.
- Dragos, P. (2021). Overview of the Agile Rational Unified Process (Rup) in the Context of Software Development Projects, *Journal of Business and Economics* p. 681.
- Edith, A. T. S. (2019). Modelado de Negocios.
- Erazo Neira, J. L. y Sánchez Escudero, M. A. (2022). Desarrollo de una aplicación en ambiente Android para el control automatizado de las variables de producción en condiciones de invernadero del tomate cherry en los hogares del cantón Daule. Accepted: 2022-05-17T04:32:50Z Publisher: Universidad de Guayaquil. Facultad de Ciencias Matemáticas y Físicas. Carrera de Ingeniería en Sistemas Computacionales.  
**URL:** <http://repositorio.ug.edu.ec/handle/redug/59912>
- FAO (2015). News detail.  
**URL:** <http://www.fao.org/soils-2015/news/news-detail/en/c/277682/>
- FAO (2022). Soil classification | FAO SOILS PORTAL | Food and Agriculture Organization of the United Nations.  
**URL:** <https://www.fao.org/soils-portal/data-hub/soil-classification/en/>
- Foundation, D. S. (2022). Django documentation.  
**URL:** <https://docs.djangoproject.com/en/4.1/>
- Gondchawar, N. y Kawitkar, R. (2016). IoT based smart agriculture, *International Journal of advanced research in Computer and Communication Engineering* **5**(6): 838–842.
- González, A. H. (2005). Identificación de procesos de negocio, *Ingeniería Industrial* **26**(1): 4. Publisher: Departamento de Ediciones (Editorial Cujae).

- Guo, Y., Zhao, H., Zhang, S., Wang, Y. y Chow, D. (2021). Modeling and optimization of environment in agricultural greenhouses for improving cleaner and sustainable crop production, *Journal of Cleaner Production* **285**: 124843. Publisher: Elsevier.
- Harris, D. R. y Fuller, D. Q. (2014). Agriculture: definition and overview, *Encyclopedia of global archaeology* pp. 104–113. Publisher: Springer New York.
- Kamilaris, A., Kartakoullis, A. y Prenafeta-Boldú, F. X. (2017). A review on the practice of big data analysis in agriculture, *Computers and Electronics in Agriculture* **143**: 23–37. Publisher: Elsevier.
- Madruga Peláez, A., Estevez Pérez, A. A., López, R. S., Santana Ching, I. y García Algora, C. M. (2019). Red de Sensores Inalámbricos para la Adquisición de Datos en Casas de Cultivo, *Ingeniería* **24**(3): 224–234.  
**URL:** <https://revistas.udistrital.edu.co/index.php/reving/article/view/14437>
- Martínez Pérez, E. M. y López de Jiménez, R. E. (2021). *Diseño de sistema multiplataforma innovador para el monitoreo y control de variables microambientales de un invernadero y una estación acuícola, aplicando e-Agricultura : propuesta para la Escuela Nacional de Agricultura*, ENA, ITCA Editores. Accepted: 2021-10-28T16:06:16Z.  
**URL:** <http://www.redicces.org.sv:80/jspui/handle/10972/4334>
- Metzner, C. y Niño, N. (2016). El proceso de desarrollo RUP-GDIS, *Revista Venezolana de Computación* **3**(1): 13–22.
- Müller, E. (2021). Web technologies on the desktop: an early look at Flutter. Accepted: 2021-06-01T15:49:08Z ISBN: 9781759422046.  
**URL:** <http://elib.uni-stuttgart.de/handle/11682/11515>
- Node.js (2022). Node.js.  
**URL:** <https://nodejs.org/es/>
- Nuxt (2022). The Intuitive Vue Framework.  
**URL:** <https://nuxtjs.org/>
- Ofori, M. y El-Gayar, O. (2021). Drivers and challenges of precision agriculture: a social media perspective, *Precision Agriculture* **22**(3): 1019–1044. Publisher: Springer.

PostgreSQL, P. G. D. (2022). PostgreSQL.

**URL:** <https://www.postgresql.org/>

Postman (2022). Postman API Platform.

**URL:** <https://www.postman.com>

Python (2022). Welcome to Python.org.

**URL:** <https://www.python.org/>

Quintero Pulgar, J. A. (2021). Sistema para el manejo de datos climáticos de pequeñas producciones agrícolas bajo invernadero: un acercamiento a la agricultura inteligente. Accepted: 2022-02-23T15:48:49Z Publisher: Universidad Nacional de Colombia.

**URL:** <https://repositorio.unal.edu.co/handle/unal/81046>

Rawat, B., Purnama, S. y Mulyati, M. (2021). MySQL Database Management System (DBMS) On FTP Site LAPAN Bandung, *International Journal of Cyber and IT Service Management* **1**(2): 173–179. Number: 2.

**URL:** <https://iiast-journal.org/ijcitsm/index.php/IJCITSM/article/view/47>

Rodríguez, L. A. G., Bazán, Y. G., Pompa, Y. G., Fernandez, L. F. y Núñez, S. P. (2016). Aplicaciones informáticas para el desarrollo del sector empresarial y social en la provincia Granma.

Schimmelpfennig, D. (2016). Farm profits and adoption of precision agriculture, *Technical report*.

Souza, J. C. A. y Oliveira, M. R. (2021). METODOLOGIAS áGEIS: um comparativo entre extreme programming (XP) e scrum, *Ciência & Tecnologia* **13**(1): 133–141. Number: 1.

**URL:** <https://citec.fatecjaboticabal.edu.br/index.php/citec/article/view/205>

Thoutam, V. (2021). A Study On Python Web Application Framework, *Journal of Electronics, Computer Networking and Applied Mathematics (JECNAM)* ISSN : 2799-1156 **1**(01): 48–55. Number: 01.

**URL:** <http://journal.hmjournals.com/index.php/JECNAM/article/view/112>

Totey, A. y Vishwakarma, D. R. G. (2021). Design A Model To Analyze Open Source Nodejs Iot Frameworks, **18**(6): 14.

## REFERENCIAS

---

Vidal-Silva, C. L., Sánchez-Ortiz, A., Serrano, J., Rubio, J. M., Vidal-Silva, C. L., Sánchez-Ortiz, A., Serrano, J. y Rubio, J. M. (2021). Experiencia académica en desarrollo rápido de sistemas de información web con Python y Django, *Formación universitaria* **14**(5): 85–94. Publisher: Centro de Información Tecnológica.

**URL:** [http://www.scielo.cl/scielo.php?script=sci\\_abstract&pid=S0718-50062021000500085&lng=es&nrm=iso&lng=es](http://www.scielo.cl/scielo.php?script=sci_abstract&pid=S0718-50062021000500085&lng=es&nrm=iso&lng=es)



# ANEXO A

## Requerimientos funcionales detallados

N°	Requerimientos	N°	Requerimientos
<b>R 1</b>	<b>Autenticar usuarios</b>	<b>R 16</b>	<b>Gestionar controles de maleza manual</b>
<b>R 2</b>	<b>Cambiar contraseña</b>	R 16.1	Insertar controles de maleza manual
<b>R 3</b>	<b>Gestionar usuarios</b>	R 16.2	Modificar controles de maleza manual
R 3.1	Insertar usuarios	R 16.3	Eliminar controles de maleza manual
R 3.2	Modificar usuarios	R 16.4	Mostrar controles de maleza manual
R 3.3	Eliminar usuarios	<b>R 17</b>	<b>Gestionar controles de maleza mecánicos</b>
R 3.4	Mostrar usuarios	R 17.1	Insertar controles de maleza mecánicos
<b>R 4</b>	<b>Gestionar modelos de invernaderos</b>	R 17.2	Modificar controles de maleza mecánicos

<b>N°</b>	<b>Requerimientos</b>	<b>N°</b>	<b>Requerimientos</b>
R 4.1	Insertar modelos de invernaderos	R 17.3	Eliminar controles de maleza mecánicos
R 4.2	Modificar modelos de invernaderos	R 17.4	Mostrar controles de maleza mecánicos
R 4.3	Eliminar modelos de invernaderos	<b>R 18</b>	<b>Gestionar controles de maleza químicos</b>
R 4.4	Mostrar modelos de invernaderos	R 18.1	Insertar controles de maleza químicos
<b>R 5</b>	<b>Gestionar invernaderos</b>	R 18.2	Modificar controles de maleza químicos
R 5.1	Insertar invernaderos	R 18.3	Eliminar controles de maleza químicos
R 5.2	Modificar invernaderos	R 18.4	Mostrar controles de maleza químicos
R 5.3	Eliminar invernaderos	<b>R 19</b>	<b>Gestionar riegos</b>
R 5.4	Mostrar invernaderos	R 19.1	Insertar riegos
<b>R 6</b>	<b>Gestionar áreas de cultivos</b>	R 19.2	Modificar riegos
R 6.1	Insertar áreas de cultivos	R 19.3	Eliminar riegos
R 6.2	Modificar áreas de cultivos	R 19.4	Mostrar riegos
R 6.3	Eliminar áreas de cultivos	<b>R 20</b>	<b>Gestionar podas</b>

<b>N°</b>	<b>Requerimientos</b>	<b>N°</b>	<b>Requerimientos</b>
R 6.4	Mostrar áreas de cultivos	R 20.1	Insertar podas
<b>R 7</b>	<b>Gestionar tipos de suelo</b>	R 20.2	Modificar podas
R 7.1	Insertar tipos de suelo	R 20.3	Eliminar podas
R 7.2	Modificar tipos de suelo	R 20.4	Mostrar podas
R 7.3	Eliminar tipos de suelo	<b>R 21</b>	<b>Gestionar aporques</b>
R 7.4	Mostrar tipos de suelo	R 21.1	Insertar aporques
<b>R 8</b>	<b>Gestionar tipos de cultivos</b>	R 21.2	Modificar aporques
R 8.1	Insertar tipos de cultivos	R 21.3	Eliminar aporques
R 8.2	Modificar tipos de cultivos	R 21.4	Mostrar aporques
R 8.3	Eliminar tipos de cultivos	<b>R 22</b>	<b>Gestionar drenajes</b>
R 8.4	Mostrar tipos de cultivos	R 22.1	Insertar drenajes
<b>R 9</b>	<b>Gestionar especies</b>	R 22.2	Modificar drenajes
R 9.1	Insertar especies	R 22.3	Eliminar drenajes

<b>N°</b>	<b>Requerimientos</b>	<b>N°</b>	<b>Requerimientos</b>
R 9.2	Modificar especies	R 22.4	Mostrar drenajes
R 9.3	Eliminar especies	<b>R 23</b>	<b>Gestionar desbotones</b>
R 9.4	Mostrar especies	R 23.1	Insertar desbotones
<b>R 10</b>	<b>Gestionar variedades</b>	R 23.2	Modificar desbotones
R 10.1	Insertar variedades	R 23.3	Eliminar desbotones
R 10.2	Modificar variedades	R 23.4	Mostrar desbotones
R 10.3	Eliminar variedades	<b>R 24</b>	<b>Gestionar controles mecánicos</b>
R 10.4	Mostrar variedades	R 24.1	Insertar controles mecánicos
<b>R 11</b>	<b>Gestionar cultivos</b>	R 24.2	Modificar controles mecánicos
R 11.1	Insertar cultivos	R 24.3	Eliminar controles mecánicos
R 11.2	Modificar cultivos	R 24.4	Mostrar controles mecánicos
R 11.3	Eliminar cultivos	<b>R 25</b>	<b>Gestionar controles químicos</b>
R 11.4	Mostrar cultivos	R 25.1	Insertar controles químicos

<b>N°</b>	<b>Requerimientos</b>	<b>N°</b>	<b>Requerimientos</b>
<b>R 12</b>	<b>Gestionar actividades fito-técnicas</b>	R 25.2	Modificar controles químicos
R 12.1	Insertar actividades fito-técnicas	R 25.3	Eliminar controles químicos
R 12.2	Modificar actividades fito-técnicas	R 25.4	Mostrar controles químicos
R 12.3	Eliminar actividades fito-técnicas	<b>R 26</b>	<b>Gestionar controles biológicos</b>
R 12.4	Mostrar actividades fito-técnicas	R 26.1	Insertar controles biológicos
<b>R 13</b>	<b>Gestionar fertilizaciones orgánicas</b>	R 26.2	Modificar controles biológicos
R 13.1	Insertar fertilizaciones orgánicas	R 26.3	Eliminar controles biológicos
R 13.2	Modificar fertilizaciones orgánicas	R 26.4	Mostrar controles biológicos
R 13.3	Eliminar fertilizaciones orgánicas	<b>R 27</b>	<b>Gestionar cosechas</b>
R 13.4	Mostrar fertilizaciones orgánicas	R 27.1	Insertar cosechas
<b>R 14</b>	<b>Gestionar fertilizaciones químicas</b>	R 27.2	Modificar cosechas
R 14.1	Insertar fertilizaciones químicas	R 27.3	Eliminar cosechas
R 14.2	Modificar fertilizaciones químicas	R 27.4	Mostrar cosechas

<b>N°</b>	<b>Requerimientos</b>	<b>N°</b>	<b>Requerimientos</b>
R 14.3	Eliminar fertilizaciones químicas	<b>R 28</b>	<b>Mostrar listado de actividades fito-técnicas realizadas a un cultivo</b>
R 14.4	Mostrar fertilizaciones químicas	<b>R 29</b>	<b>Mostrar listado de actividades fito-sanitarias realizadas a un cultivo</b>
<b>R 15</b>	<b>Gestionar fertilizaciones biológicas</b>	<b>R 30</b>	<b>Cerrar sesión</b>
R 15.1	Insertar fertilizaciones biológicas		
R 15.2	Modificar fertilizaciones biológicas		
R 15.3	Eliminar fertilizaciones biológicas		
R 15.4	Mostrar fertilizaciones biológicas		

# **ANEXO B**

**B.1. Diagramas de colaboración (Caso de uso "Gestionar cultivos")**

**B.2. Diagramas de clases de diseño (Caso de uso "Gestionar cultivos")**

## B.2 Diagramas de clases de diseño (Caso de uso "Gestionar cultivos")

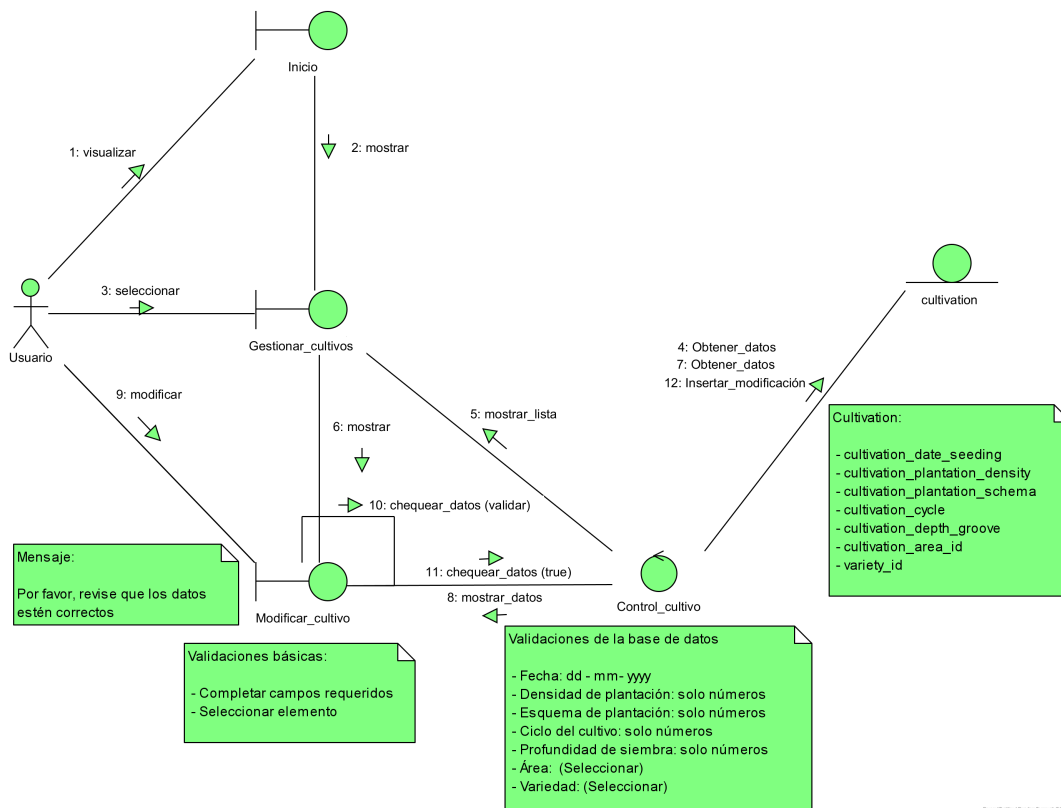


Figura B.1: Diagrama de colaboración «Gestionar cultivos». Escenario «Modificar cultivo»



## B.2 Diagramas de clases de diseño (Caso de uso "Gestionar cultivos")

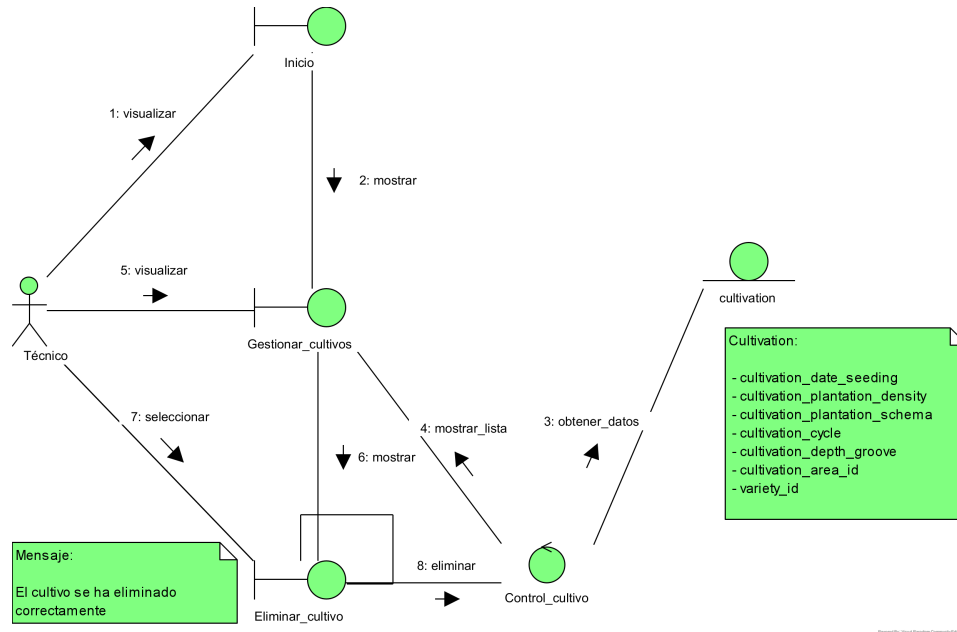


Figura B.2: Diagrama de colaboración «Gestionar cultivos». Escenario «Eliminar cultivo»

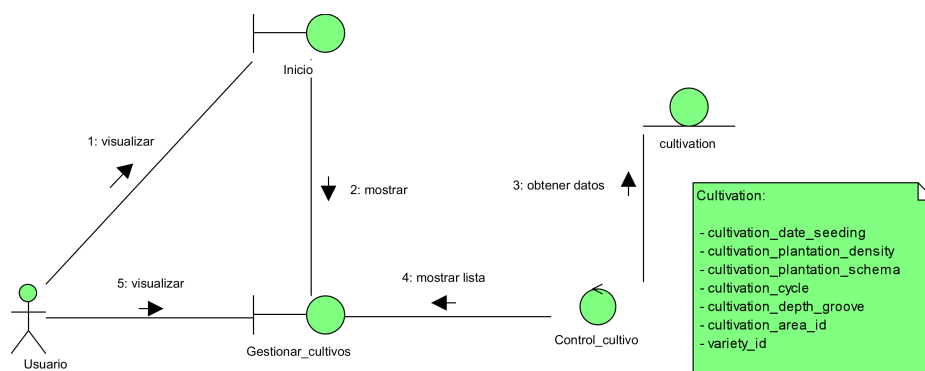


Figura B.3: Diagrama de colaboración «Gestionar cultivos». Escenario «Mostrar cultivo»

## B.2 Diagramas de clases de diseño (Caso de uso "Gestionar cultivos")

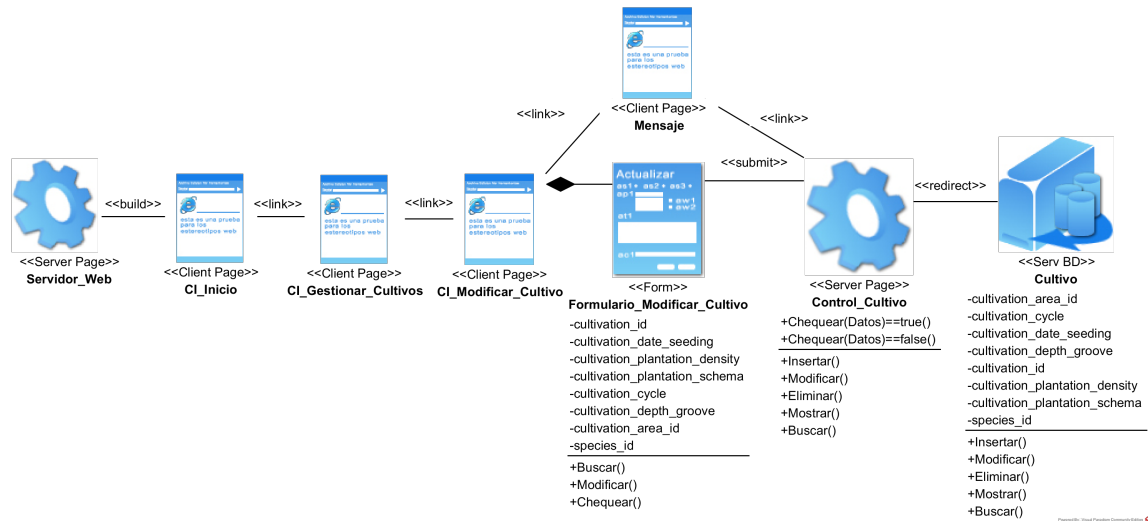


Figura B.4: Diagrama de diseño «Modificar cultivo»

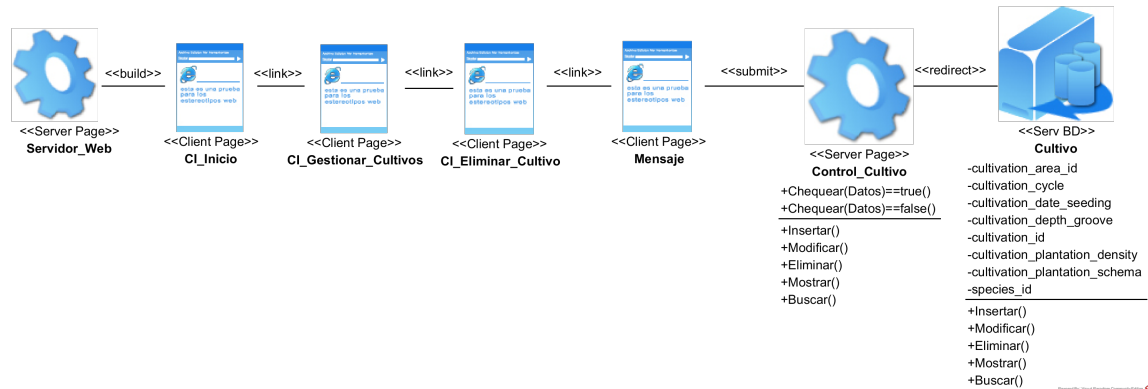


Figura B.5: Diagrama de diseño «Eliminar cultivo»

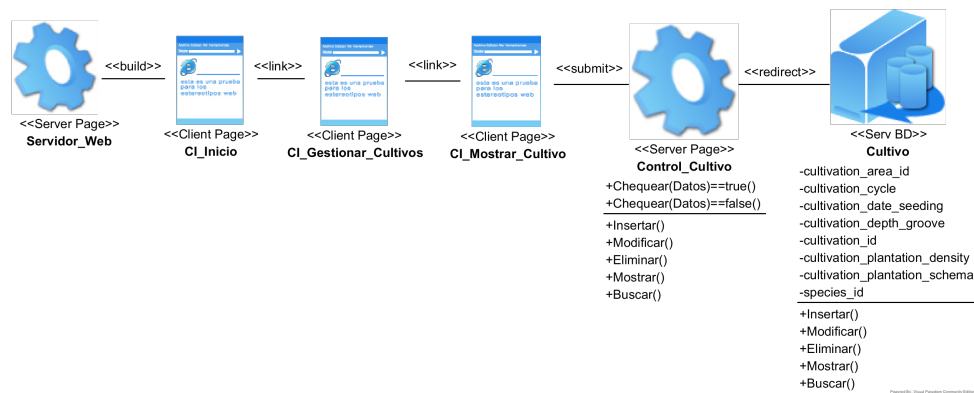


Figura B.6: Diagrama de diseño «Mostrar cultivo»