



UNIVERSIDAD LAICA “ELOY ALFARO” DE MANABÍ
EXTENSIÓN EN EL CARMEN
CARRERA DE INGENIERÍA DE SOFTWARE
Creada Ley No. 10 – Registro Oficial 31 de noviembre 13 de 1985

PROYECTO INTEGRADOR
PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO DE
SOFTWARE

Sistema web con arquitecturas Frontend dinámicas para la planificación
de producción del vivero Lastenia de Uleam El Carmen

Verduga Calderon Heyder Javier

AUTOR

Ing. Mendoza Villamar Rocío Alexandra

TUTOR

El Carmen, agosto 2026



CERTIFICACIÓN DEL DIRECTOR DE TRABAJO DE GRADUACIÓN

	NOMBRE DEL DOCUMENTO: CERTIFICADO DE TUTOR(A).	CÓDIGO: PAT-04-F-004
	PROCEDIMIENTO: TITULACIÓN DE ESTUDIANTES DE GRADO BAJO LA UNIDAD DE INTEGRACIÓN CURRICULAR	REVISIÓN: 1
		Página 1 de 1

CERTIFICACIÓN

En calidad de docente tutor(a) de la Extensión El Carmen de la Universidad Laica "Eloy Alfaro" de Manabí, CERTIFICO:

Haber dirigido, revisado y aprobado preliminarmente el Trabajo de Integración Curricular bajo la autoría del estudiante **Verduga Calderon Heyder Javier**, legalmente matriculado/a en la carrera de Ingeniería en Software, período académico 2025(2)-2026(1), cumpliendo el total de 288 horas, cuyo tema del proyecto **Sistema web con arquitecturas Frontend dinámicas para la planificación de producción del vivero Lastenia de Uleam El Carmen**.

La presente investigación ha sido desarrollada en apego al cumplimiento de los requisitos académicos exigidos por el Reglamento de Régimen Académico y en concordancia con los lineamientos internos de la opción de titulación en mención, reuniendo y cumpliendo con los méritos académicos, científicos y formales, y la originalidad del mismo, requisitos suficientes para ser sometida a la evaluación del tribunal de titulación que designe la autoridad competente.

Particular que certifico para los fines consiguientes, salvo disposición de Ley en contrario.

Lugar, El Carmen 3 de agosto del 2026.

Lo certifico,


Ing. Rocío Mendoza Villamar, Mgtr.
Docente Tutora
Área: Ingeniería de Software

TRIBUNAL DE SUSTENTACIÓN



Universidad Laica Eloy Alfaro de Manabí

Extensión El Carmen

Carrera de Ingeniería en Software

TRIBUNAL DE SUSTENTACIÓN

Título del Trabajo de Titulación:

SISTEMA WEB CON ARQUITECTURAS FRONTEND DINÁMICAS PARA LA
PLANIFICACIÓN DE PRODUCCIÓN DEL VIVERO LASTENIA DE LA ULEAM EL CARMEN

Modalidad:

Proyecto Integrador

Autor:

Verduga Calderon Heyder Javier

Tutor:

Ing. Mendoza Villamar Rocio Alexandra

Tribunal de Sustentación:

- **Presidente:** Ing. Mora Marcillo Alex Bladimir
- **Miembro:** Ing. Reascos Pinchao Raul Saed
- **Miembro:** Ing. Lopez Rodriguez Carlos Vinicio

Fecha de Sustentación:

26 de agosto de 2026

EXTENSIÓN EN EL CARMEN



DECLARACIÓN DE AUTORÍA

La responsabilidad del contenido de este Trabajo de titulación, cuyo tema es: “Sistema web con arquitecturas Frontend dinámicas para la planificación de producción del vivero Lastenia de Uleam El Carmen”, corresponde exclusivamente a: Verduga Calderon Heyder Javier CI. 1351573058 los derechos patrimoniales de la misma corresponden a la Universidad Laica “Eloy Alfaro” de Manabí.

A handwritten signature in blue ink, appearing to be "J. Verduga", written over a horizontal line.

Verduga Calderon Heyder Javier
CI. 1351573058

DEDICATORIA

Como primer lugar dedico este trabajo a Dios, quien me ha prestado fuerza y claridad en este camino. Su guía en cada decisión y en cada paso han iluminado mis años de formación universitaria, me ha dado valentía para sobrepasar momentos difíciles y ha hecho mantener firme mi convicción. Sin su presencia constante, este logro no tendría el mismo peso, pues ha sido El que me ha recordado en cada etapa que con fe, esfuerzo y perseverancia todo es posible.

De manera muy especial, dirijo esta dedicatoria a mi madre, Vanessa Calderón, quien con su amor inquebrantable, sacrificio diario y apoyo incondicional ha sido el pilar más grande en mi vida. Ella ha estado presente en cada uno de mis procesos, alentándome con palabras de aliento, enseñándome con su ejemplo de lucha y dándome la motivación necesaria para no rendirme. A ella le debo gran parte de este logro, pues su esfuerzo silencioso y su entrega constante me han inspirado a convertirme en una mejor persona y profesional. Asimismo, extendiendo mi gratitud a mi padre, Edwin Verduga, por sus consejos y acompañamiento en este trayecto, y a mi novia, Mishelle Moreira, cuya compañía, comprensión y cariño me han dado la fuerza adicional para continuar con firmeza en los momentos en los que más lo necesitaba.

También agradezco a mis docentes y a la Universidad Laica Eloy Alfaro de Manabí, extensión El Carmen, por brindarme las herramientas necesarias para mi formación académica y exigirme siempre lo mejor de mí. A mis colegas y amistades que me acompañaron en este proceso, gracias por compartir experiencias, apoyo y motivación. Este proyecto de titulación no representa únicamente un compromiso personal, sino también el reflejo del respaldo de todas aquellas personas que han creído en mí. Con humildad y profundo agradecimiento, lo presento como un tributo a la dedicación, la perseverancia y al amor de quienes forman parte de mi vida.

Verduga Calderon Heyder Javier

AGRADECIMIENTO

Expreso mi profundo agradecimiento a los docentes de la Carrera de Ingeniería en Software de la Universidad Laica Eloy Alfaro de Manabí (ULEAM), quienes con su dedicación y conocimiento experto guiaron mis pasos académicos a lo largo de esta carrera exigente. En particular, agradezco a los profesores que dirigieron el proceso de titulación grupal, ofreciendo orientaciones técnicas clave durante el desarrollo del Sistema de Gestión de Vivero de Cacao, y a aquellos que en asignaturas fundamentales como arquitectura de software y bases de datos sembraron las bases sólidas que hicieron posible este proyecto integrador. Su paciencia, retroalimentación constructiva y compromiso con la excelencia profesional han sido fundamentales para mi formación como ingeniero.

También extiendo mi gratitud sincera a mi familia, cuyo apoyo incondicional ha sido el pilar de mi perseverancia durante estos años universitarios. A mis padres y mi abuela materna, gracias por su comprensión ante las largas horas de estudio, su motivación constante y el sacrificio compartido que me permitió enfocarme en esta meta. Este logro no solo representa el cierre de mi etapa académica, sino un tributo a su fe en mí y al ejemplo de esfuerzo que siempre me han dado.

ÍNDICE GENERAL

CERTIFICACIÓN DEL DIRECTOR DE TRABAJO DE GRADUACIÓN.....	III
TRIBUNAL DE SUSTENTACIÓN	¡Error! Marcador no definido.
DECLARACIÓN DE AUTORÍA	V
DEDICATORIA	VI
AGRADECIMIENTO	VII
ÍNDICE GENERAL.....	VIII
ÍNDICE DE TABLAS	XI
ÍNDICE DE ILUSTRACIONES.....	XIII
ÍNDICE DE ANEXOS	XV
RESUMEN.....	XVI
ABSTRACT	XVII
CAPÍTULO I	1
1. INTRODUCCIÓN	1
1.1 Introducción	1
1.2 Presentación del tema	2
1.3 Ubicación y contextualización de la problemática	2
1.4 Planteamiento del problema	3
1.4.1 Problematización	3
1.4.2 Génesis del problema	3
1.4.3 Estado actual del problema	4
1.5 Diagrama causa – efecto del problema.	5
1.6 Objetivos	5
1.6.1 Objetivo general	5
1.6.2 Objetivos específicos	6
1.7 Justificación	6
1.8 Impactos esperados	7

1.8.1	Impacto tecnológico	7
1.8.2	Impacto social.....	8
1.8.3	Impacto ecológico	8
CAPÍTULO II.....		10
2	MARCO TEÓRICO	10
2.1	Antecedentes históricos.....	10
2.2	Antecedentes de investigaciones relacionadas al tema presentado	11
2.3	Definiciones conceptuales.....	12
2.3.1	Sistema web con arquitectura frontend dinámica.	12
2.3.2	Planificación de producción.	19
2.4	Metodología Propuesta.....	25
2.4.1	Metodologías Ágiles (Scrum):.....	25
2.5	Conclusiones.....	27
CAPÍTULO III.....		28
3	MARCO INVESTIGATIVO	28
3.1	Introducción.	28
3.2	Tipo de Investigación.	28
3.2.1	Investigación Bibliográfica:.....	28
3.2.2	Investigación de Campo:	29
3.3	Métodos de Investigación	29
3.3.1	Método Cualitativo.....	29
3.3.2	Método Cuantitativo	29
3.4	Fuentes de Información de Datos.	30
3.4.1	Fuentes Primarias.	30
3.4.2	Fuentes Secundarias.....	30
3.5	Estrategia operacional para la recolección de datos.	31
3.5.1	Población	31
3.5.2	Segmentación.	31
3.5.3	Tamaño de la Muestra.....	32
3.5.4	Análisis de las herramientas de recolección de datos a utilizar.	32
3.5.5	Plan de recolección de datos.....	35

3.6	Análisis y presentación de resultados	35
3.6.1	Tabulación y análisis de los datos.....	35
3.6.2	Presentación y descripción de los resultados obtenidos.....	40
3.6.3	Informe final de análisis de datos.....	42
CAPÍTULO IV:.....		43
4	MARCO PROPOSITIVO.....	43
4.1	Introducción	43
4.2	Descripción de la propuesta	43
4.3	Determinación de recursos.....	45
4.3.1	Humanos:	45
4.3.2	Tecnológicos:	45
4.3.3	Económicos:.....	47
4.4	Desarrollo según metodología Scrum	47
4.4.1	Descripción del Producto.....	47
4.4.2	Historias de Usuario	51
4.4.3	Diseño del Sistema / Descripción Técnica	54
4.4.4	Descripción Técnica / Arquitectura del Sistema	63
4.4.5	Roles y Responsabilidades.....	66
4.4.6	Planificación del Sprint	68
4.4.7	Backlog del Producto	74
4.4.8	Interfaz de Usuario (UI) / Prototipos	81
4.4.9	Definición de Hecho (DoD)	85
4.4.10	Eventos Scrum	87
4.4.11	Proceso de Pruebas	90
4.4.12	Incremento y Entregables	93
CAPÍTULO V.....		97
5.	EVALUACIÓN DE RESULTADOS	97
5.1	Introducción	97
5.2	Presentación y monitoreo de resultados.....	97
5.2.1	Planificación de la Evaluación.....	97
5.2.2	Ejecución del Monitoreo	98
5.3	Interpretación Objetiva.....	101

CAPITULO VI.....	103
6 CONCLUSIONES Y RECOMENDACIONES.....	103
6.1 Conclusiones.....	103
6.2 Recomendaciones.....	104
BIBLIOGRAFÍA	105
ANEXOS	114
GLOSARIO	120

ÍNDICE DE TABLAS

Tabla 1: Plan de recolección de datos.....	35
Tabla 2: Tabulación y análisis de los datos	36
Tabla 3: Tabulación y resultado de la Entrevista.....	38
Tabla 4: Recursos Humanos	45
Tabla 5: Recursos Tecnológicos de Hardware.....	45
Tabla 6: Recursos tecnológicos de Software	46
Tabla 7: Recursos Económicos	47
Tabla 8: Usuarios Objetivo	50
Tabla 9: Historia de usuario 1	51
Tabla 10: Historia de usuario 2	51
Tabla 11: Historia de usuario 3	52
Tabla 12: Historia de usuario 4.....	53

Tabla 13: Historia de usuario 5	53
Tabla 14: Historia de usuario 6	54
Tabla 15: Caso de uso - Gestion de Metas y Cronograma de Producción.....	55
Tabla 16: Caso de uso - Monitoreo de Lotes	56
Tabla 17: Caso de uso - Reprogramación de Cronograma	57
Tabla 18: Caso de uso - Gestión de Alertas y Despachos.....	58
Tabla 19: Roles en el marco Scrum	66
Tabla 20: Roles de Usuario en el Sistema	67
Tabla 21: Sprint 1 - Diseño de base de datos.....	68
Tabla 22: Sprint 2 - Motor de Planificación y Algoritmos	69
Tabla 23: Sprint 3 - Dashboard Reactivo e Interfaz de Usuario	70
Tabla 24: Sprint 4 - Motor de Reprogramación y Resiliencia	71
Tabla 25: Sprint 5 - Alertas y Notificaciones	72
Tabla 26: Sprint 6 - Pruebas de Integración y Registro de Despachos.....	73
Tabla 27: Backlog Inicial.....	74
Tabla 28: Backlog - Fin del Sprint 1	75
Tabla 29: Backlog - Fin del Sprint 2.....	76
Tabla 30: Backlog - Fin del Sprint 3.....	77
Tabla 31: Backlog - Fin del Sprint 4.....	78
Tabla 32: Backlog - Fin del Sprint 5.....	79
Tabla 33: Backlog - Fin del Sprint 6.....	80

Tabla 34: Formulario de Registro de Meta de Producción	91
Tabla 35: Formulario de Registro de Lote	91
Tabla 36: Formulario de Reprogramación de Cronograma	91
Tabla 37: Prueba de caja blanca - Formulario de Registro de Meta de Producción	92
Tabla 38: Prueba de caja blanca - Formulario de Registro de Lote	92
Tabla 39: Prueba de caja blanca - Formulario de Reprogramación de Cronograma ...	93
Tabla 40: Prueba de caja blanca - Validación de Progreso de Meta de Producción	93
Tabla 41: Planificación de la Evaluación	97
Tabla 42: Planificación Automática de Cronogramas	98
Tabla 43: Toma de datos Planificación de Cronogramas	99
Tabla 44: Toma de datos Reprogramación del Cronograma	100
Tabla 45: Toma de datos Validación de Despachos	100
Tabla 46: Tabla comparativa sin sistema y con sistema	101

ÍNDICE DE ILUSTRACIONES

Ilustración 1: Diagrama de causa-efecto	5
Ilustración 2: Caso de Uso - Gestión de Metas y Cronograma de Producción	55
Ilustración 3: Caso de uso - Monitoreo de Lotes	56
Ilustración 4: Caso de uso - Reprogramación de Cronograma	57
Ilustración 5: Caso de uso - Gestión de Alertas y Despachos	58

Ilustración 6: Diagrama de secuencia - Calculo de Cronograma Automático.....	59
Ilustración 7: Diagrama de secuencia - Monitoreo de Lotes	59
Ilustración 8: Diagrama de secuencia - Reprogramación de Cronograma	60
Ilustración 9: Diagrama de secuencia - Registro de Despacho.....	60
Ilustración 10: Diagrama de estado - Meta de Producción	61
Ilustración 11: Diagrama de estado - Lote	61
Ilustración 12: Diagrama de estado - Ciclo de Producción del Lote	62
Ilustración 13: Diagrama de Base de datos.....	62
Ilustración 14: Mapa de Navegación del Sistema.....	81
Ilustración 15: Pantalla de Inicio de Sesión	82
Ilustración 16: Pantalla creación de Viveros	82
Ilustración 17: Pantalla Monitoreo de Lotes	83
Ilustración 18: Pantalla de Fases de Producción.....	84
Ilustración 19: Pantalla de Resumen.....	85
Ilustración 20: Ingreso de Usuario en el login	98
Ilustración 21: Definir meta de producción del Vivero	99
Ilustración 22: Ejecución de algoritmo de calculo de fases	99

ÍNDICE DE ANEXOS

Anexo 1: Correo de Asignacion de Tutor	114
Anexo 2: Evidencia de Encuesta parte 1.....	115
Anexo 3: Evidencia de Encuesta parte 2.....	116
Anexo 4:Evidencia de Entrevista Semi-Estructurada	117
Anexo 5: Pruebas tecnicas en el Viveo Lastenia	118
Anexo 6: Visita al vivero Lastenia.....	118
Anexo 7: Certificado de detección de IA Compilatio.....	119

RESUMEN

El Vivero Lastenia, perteneciente a la Universidad Laica Eloy Alfaro de Manabí (ULEAM), Extensión El Carmen, gestionaba de forma manual y fragmentada la planificación de la producción de plántulas de cacao, lo que generaba retrasos, errores en el cálculo de cronogramas y una falta de trazabilidad sobre el estado real de los lotes en producción.

Ante esta problemática, el presente trabajo de titulación propone el desarrollo de ERP Lastenia, un sistema web con arquitectura frontend dinámica, desarrollado bajo la metodología ágil Scrum, cuyo módulo de Planificación y Seguimiento automatiza el cálculo del cronograma de las 5 fases técnicas del ciclo productivo del cacao (preparación, siembra, germinación, injertación, desarrollo y despacho) a partir de la fecha de inicio de cada lote.

Enfocados en la experiencia del usuario, el frontend se construyó con React y TailwindCSS bajo el patrón MVVM, y el backend con Laravel y MySQL. En donde se desarrollaron funcionalidades, entre las cuales están el registro de una meta de producción por vivero y el registro de lotes con cálculo automático de capacidad, la visualización de lotes en un calendario interactivo, con reprogramación manual de fases con recálculo de las mismas al adelantar o retrasar el cambio de fases, todo resumido en un dashboard de resumen operativo y finaliza con el registro visual de despachos reales.

Basado en las pruebas realizadas del sistema se demostró una reducción del tiempo de cálculo de cronogramas de 49 minutos a 3 segundos, y del tiempo de reprogramación de 20 minutos a 6 segundos en promedio, lo cual confirma la eficiencia técnica que tiene la solución de software frente al proceso manual que antes se usaba antes en el vivero Lastenia. En este módulo se resuelve la problemática de la discontinuidad estructural detectada en el diagnóstico inicial.

Palabras clave: planificación de producción, arquitectura frontend dinámica, Scrum, MVVM, vivero de cacao.

ABSTRACT

The Lastenia Nursery, belonging to the Eloy Alfaro Lay University of Manabí (ULEAM), El Carmen Extension, managed the production planning of cacao seedlings manually and in a fragmented manner, resulting in delays, errors in schedule calculations, and a lack of traceability regarding the actual status of production batches.

To address this problem, this thesis proposes the development of ERP Lastenia, a web-based system with a dynamic frontend architecture, developed using the Scrum agile methodology. Its Planning and Monitoring module automates the calculation of the schedule for the five technical phases of the cacao production cycle (preparation, planting, germination, grafting, development, and shipment) based on the start date of each batch. With a focus on user experience, the frontend was built using React and Tailwind CSS under the MVVM pattern, and the backend using Laravel and MySQL. The system developed functionalities including the registration of a production target per nursery and the registration of batches with automatic capacity calculation, the visualization of batches on an interactive calendar, and manual rescheduling of phases with recalculation when phase changes are advanced or delayed. All of this is summarized in an operational dashboard and concludes with the visual recording of actual shipments.

Based on system testing, a reduction in schedule calculation time from 49 minutes to 3 seconds and in rescheduling time from 20 minutes to 6 seconds on average was demonstrated. This confirms the technical efficiency of the software solution compared to the manual process previously used at the Lastenia nursery. This module resolves the structural discontinuity identified in the initial diagnostic assessment. **Keywords: production planning, dynamic frontend architecture, Scrum, MVVM, cacao nursery.**

CAPÍTULO I

1. INTRODUCCIÓN

1.1 Introducción

Para la economía ecuatoriana, producir cacao es un pilar fundamental, sobre todo en regiones costeras como Manabí, donde la Uleam extensión El Carmen mantiene el vivero de cacao en el campus Lastenia como punto de investigación y producción de plántulas. Sin embargo, los procesos manuales de gestión dentro del vivero generan deficiencias en la planificación de la producción, monitoreo, coordinación del personal en disponibilidad, lo que afecta directamente la productividad y sostenibilidad administrativa. El siguiente proyecto de titulación propone desarrollar un Sistema de Gestión del Vivero de Cacao, un proyecto el cual está compuesto por seis módulos interconectados que facilitan la administración de los recursos del vivero.

El sistema aborda los desafíos específicos que enfrenta el vivero de la Uleam El Carmen con una solución tecnológica de última generación que permite operaciones de campo más eficientes, con escalabilidad y seguimiento en todo tiempo. Este sistema es una plataforma unificada por los módulos de todos los integrantes del grupo, garantizando esa consistencia técnica y funcional que se espera de un sistema de esta magnitud. Dicho sistema no sólo aborda las limitaciones locales, sino que también está alineado con la tendencia global de agricultura inteligente aplicada a los viveros.

En este documento se ha detallado el proceso de análisis, selección bibliográfica analítica y justificación con criterios suficientes, siguiendo metodologías académicas estandarizadas para asegurar la rigurosidad y fiabilidad de la información brindada. Los resultados validan la solución propuesta, teniendo en cuenta que se tiene que adaptar al contexto rural. Por esto, este proyecto sostiene un avance tecnológico, práctico y sostenible para el correcto desarrollo del vivero de cacao Lastenia.

1.2 Presentación del tema

Se presenta el tema “Sistema web con arquitecturas frontend dinámicas para la planificación de producción del vivero Lastenia de Uleam El Carmen”, por el cual se busca innovar y ayudar administrativamente a la producción y el fortalecimiento educativo dentro del vivero Lastenia de Uleam El Carmen. Se da esta propuesta ya que la organización y monitoreo manual de dicho vivero ha puesto en disminución la efectividad administrativa y la producción de plántulas. Con la implementación de este sistema, En este proyecto se integraron todas las etapas de producción, optimización de los recursos, facilitar el monitorear en todo tiempo y proporcionar herramientas para la tomar decisiones, poniendo como base una gestión moderna y competitiva.

1.3 Ubicación y contextualización de la problemática

El vivero está ubicado en el campus Lastenia de la Uleam extensión El Carmen, en el cantón El Carmen dirección Av. Chone kilómetro 32, donde antes realizaba sus actividades el colegio Lastenia Vera. El sector es estratégico ya que allí es donde se dan esfuerzos académicos y de productividad para fomentar el desarrollo de los conocimientos agronómicos de la extensión. En este contexto se puede decir que la Universidad no solo cumple un rol formativo, sino también da pasos en innovación, vinculación en la sociedad y da su apoyo en lo que es la producción agrícola medio de proyectos que se enfocan en el mejoramiento de cultivos como es el del cacao. Con estas iniciativas se impulsa el crecimiento y esto aporta en la economía local, sumando a estudiantes universitarios, productores de cacao y especialistas en el área con tecnologías que se pueden aplicar en la agricultura.

En la actualidad, en el contexto donde el vivero lleva una gestión manual y fragmentada, existen limitaciones en el control, seguimiento y registro de actividades. Las tareas, desde cuando se preparan las fundas con tierra fértil y la respectiva siembra, hasta cuando se hace el monitoreo para la germinación y posterior entrega de las plántulas, carecen de un sistema que haga más fácil las actividades administrativas, optimizar los recursos y tomar decisiones operativas. Cuando hay altos picos de demanda se agrava la situación, es decir, que la velocidad y precisión administrativa tienen más relevancia a la hora de evitar perder recursos y e intentar no generar retrasos que signifiquen el retroceso de la producción.

1.4 Planteamiento del problema

1.4.1 Problemática

Como problemática en el vivero Lastenia se puede considerar que la gestión de la producción se basa en procesos manuales, con registros que se encuentran dispersos y no centralizados. Esta situación dificulta disponer de información organizada y actualizada que permita realizar un seguimiento adecuado de la producción de plántulas de cacao. Al no contar con información integrada sobre los diferentes lotes en producción, se dificulta conocer oportunamente su estado y determinar el avance de las actividades que se encuentran en curso. Estas limitaciones pueden generar retrasos e inconsistencias en la información recabada de cada lote, dificultando que los responsables administrativos puedan anticiparse a situaciones como variaciones en la demanda de plántulas, problemas en la germinación de las mismas o requerimientos urgentes de insumos. De igual manera, la falta de información consolidada limita la posibilidad de valorar el comportamiento de la producción mediante métricas e indicadores de efectividad, dificultando identificar oportunamente aspectos que requieran ser mejorados.

Analizando por separado la situación, el manejo manual de los registros puede propiciar errores, omisiones o pérdida de información, dificultando mantener datos precisos sobre las actividades realizadas en cada lote. Esta condición también puede repercutir en la planificación de las compras de insumos, debido a la dificultad para determinar oportunamente las necesidades que demanda la producción. Como consecuencia, la planificación puede verse condicionada por la aparición de imprevistos, generando una gestión que responde a las necesidades conforme estas se presentan en lugar de anticiparse a ellas. Por otra parte, la falta de información organizada sobre el avance de cada lote dificulta determinar el estado real de los diferentes procesos en ejecución dentro del vivero. Esta situación puede generar poca agilidad administrativa y aumentar la carga operativa, debido al tiempo que se requiere para recopilar, verificar y relacionar información necesaria para el seguimiento de la producción.

1.4.2 Génesis del problema

Cuando se habla del origen del problema en el vivero Lastenia, se debe considerar que, desde que el vivero entró en funcionamiento, la administración de los procesos del vivero se ha mantenido manualmente y con registros físicos, lo cual es una práctica común en muchos

viveros de cacao dentro de la región, incluso del país, que por lo general tienen un enfoque tradicional en la gestión agrícola. Esto se ha visto agravado porque se decide dar limitaciones a los recursos dados a la capacitación del personal y que se incorporen innovaciones tecnológicas, generando así que el flujo de información que pueda aportar los viveros sea incompleta e ineficiente.

Entrando en más detalles, el crecimiento de la demanda y que existan diversas actividades dentro del vivero que tienen que ser planificadas, monitoreadas y registradas han puesto en evidencia que hay una ineficiencia en los métodos manuales para afrontar este tipo de retos. La gestión principalmente fragmentada da lugar a que haya errores frecuentes en la preparación, al darle un seguimiento a los lotes y al controlar inventarios, lo que deriva en pérdidas rendimiento que afecta proporcionalmente en la calidad del producto final. De igual manera, la carencia de automatización de procesos administrativos dificulta el generar reportes que sean fiables, dar análisis históricos de ciclos anteriores y ejecutar acciones que puedan corregir falencias de la manera más oportuna. Como consecuencias tenemos que el vivero enfrenta limitantes claras que no permiten escalar sus procesos y consolidar desarrollo como referente académico y de innovación agrícola como apoyo a la gran comunidad productora de cacao que existe en la región.

1.4.3 Estado actual del problema

El vivero, perteneciente a la Uleam extensión El Carmen, es un proyecto agrícola y de vinculación con la comunidad que es promovida por la misma universidad, teniendo como propósito la investigación y la formación práctica de los estudiantes de las áreas de agronomía de la extensión, así mismo también se busca contribuir al desarrollo a productores dentro de la región mediante entrega de plántulas calidad. En todo este tiempo ha sido un espacio de formación, investigación y experimentación, aunque su gestión productiva se ha visto envuelta en deficiencias debido a que se ha llevado siempre una organización empírica y manual lo que da paso a la falta un registro de actividades que se pueda monitorear y verificar digitalmente, y a la desorganización a la hora de impartir tareas e incluso insumos que se necesiten.

En la actualidad, en el vivero se pueden presentar deficiencias relacionadas con la planificación operativa y la organización de su producción, lo que ha generado que no se coordinen de manera oportuna las diferentes etapas como lo son la siembra, germinación,

trasplante y entrega, ocasionando retrasos, carga de tareas y desconocimiento del estado real de los lotes que están en producción en ese momento. Esto impide que se aproveche de manera más optimizada la capacidad de producción que puede ofrecer el vivero, disminuye la eficiencia y el rendimiento en la utilización de recursos, y por ende termina afectando la calidad de las plántulas producidas por despachar.

1.5 Diagrama causa – efecto del problema.



Ilustración 1: Diagrama de causa-efecto

1.6 Objetivos

1.6.1 Objetivo general

Desarrollar un sistema web con arquitectura frontend dinámica para la planificación y gestión automatizada de la producción de plántulas de cacao en el vivero del Campus Lastenia de la Uleam extensión El Carmen, el cual se hará mediante la integración de un módulo encargado de registrar, monitorear y controlar de los procesos de gestión administrativa de vivero Lastenia en tiempo real.

1.6.2 Objetivos específicos

Realizar una recopilación bibliográfica sobre las teorías, las metodologías y los estudios que se han hecho previamente con relación a sistemas web y una planificación productiva en viveros, lo cual servirá como base teórica y técnica para desarrollar el proyecto.

Realizar un estudio de campo el cual se hará mediante entrevistas y encuestas a entidades administrativas y de autoridad en el vivero Lastenia para reconocer necesidades, conseguir requerimientos técnicos y funcionales para el diseño de las interfaces reactivas del sistema frontend que ayuden con la planificación y seguimiento de producción.

Implementar un módulo digital especializado para gestionar y planificar automatizadamente los flujos de trabajo, el control de los recursos y la visualización en todo tiempo de producción de las plántulas.

Validar la eficiencia del sistema que se ha propuesto por medio pruebas en situaciones reales y retroalimentación con los usuarios finales del sistema.

1.7 Justificación

El proyecto "Sistema web con arquitecturas frontend dinámicas para la planificación de producción del vivero Lastenia de Uleam extensión El Carmen" tiene como propósito lograr una significativa transformación y mejora de los procesos de gestión mediante la implementación del módulo automatizado. Este sistema busca que se facilite la administración, que esta sea eficiente, pueda optimizar tiempos de trabajo gestionando automáticamente las etapas productivas, ayude a mejorar la precisión y acierto en la toma de decisiones. Dichos beneficios permitirán que exista un mejor control del inventario y que se planifique de una manera más ajustada a las necesidades reales presentes en el vivero, lo que significa menos carga administrativa y más tiempo enfocado en la productividad y calidad de las plántulas producidas.

Además, el sistema se proyecta a ser parte del crecimiento institucional de la Uleam, el cual será favorecido por la implementación de soluciones innovadoras y tecnológicas en ámbitos académicos y productivos. Esta implementación promueve una mejora continua y asegura un aprendizaje más práctico en estudiantes y docentes profesionales en el área, dando

facilidad al análisis de resultados a través de reportes y monitoreo interactivo. Y mediante esta forma, el vivero

Como último punto, se espera también que se encuentren desarrollos sostenibles en la producción de cacao en la región, al fomentar esta mejora en eficiencia y control de los tiempos en un vivero. Este desarrollo, en la forma de afrontar la gestión productiva del vivero, implicará un impacto positivo en el ámbito de producción de cacao, con beneficios socioeconómicos para la comunidad cacaotera que se asienta en la región. El proyecto se justifica por su aporte que brinda en la innovación tecnológica, la educación y el desarrollo agronómico.

1.8 Impactos esperados

1.8.1 Impacto tecnológico

De acuerdo con Alvarado (2025), *“la implementación de sistemas web en la gestión agrícola de pequeños productores de cacao representa un avance técnico significativo que reduce el tiempo de registro de actividades en un 74,5% y disminuye un 72% los errores en inventarios, validando su utilidad para optimizar procesos productivos”* (p. 3). Aquí se puede evidenciar que la adopción de plataformas digitales en el mundo agrícola no solo trata de modernización las operaciones administrativas, sino también, incita a una gestión más eficiente y confiable. Las arquitecturas frontend dinámicas brindan una interfaz adaptable, amigable e interactiva que ayuda a la experiencia del usuario en estos entornos rurales y productivos, de este modo se promoverá una transición hacia la digitalización sustentable de una forma óptima.

El impacto tecnológico del proyecto se refleja incorporar un sistema con arquitecturas frontend dinámicas para que se optimice la gestión y control de la producción en viveros de cacao. Esta mejora tecnológica ayuda en el procesamiento y almacenamiento de la información que se pueda recolectar de resultados productivos del vivero, permitiendo hacer actualizaciones de la planificación en tiempo real y minimizar márgenes de error en producción. Además, que gracias a que potencia la toma de decisiones dejando que el usuario visualice interactivamente los datos, ayudando a la autonomía del productor en el manejo y gestión del vivero. En consecuencia, esto impulsa a que se promuevan soluciones eficientes y sostenibles con herramientas tecnológicas en procesos productivos agrícola.

1.8.2 Impacto social

De acuerdo con Santero Montalvo y Meneses Cabrera (2023), el cultivo de cacao y las iniciativas asociativas vinculadas a su producción han actuado como verdaderos agentes de transformación social en territorios rurales, al mejorar los ingresos familiares, fortalecer la organización comunitaria y promover la participación de mujeres y pequeños productores en la toma de decisiones. Al respecto, se demuestra que las intervenciones centradas a la estructuración y el acceso avanzado a programas de gestión pueden tener impactos positivos en la unión social de la comunidad, la disminución de la pobreza y la edificación de proyectos de desarrollo local en comunidades cacaoteras.

Asimismo, se evidencia que el proyecto propone contribuir con un impacto social positivo, permitiendo la integración de herramientas tecnológicas accesibles, funcionales e intuitivas en viveros de cacaos. La plataforma con arquitectura Frontend Dinámica favorece el desarrollo de las capacidades locales para la capacitación tecnológica, fortaleciendo el sentido de pertenecer a un modelo productivo moderno y colaborativo. De igual manera, la transformación digital impulsa a mejorar las condiciones laborales, garantizando oportunidades de empoderamiento a pequeños productores, asegurando un desarrollo social sostenible al avance tecnológico.

1.8.3 Impacto ecológico

La Alianza de Bioversity International y el CIAT (2023) señalan que la sostenibilidad de la cadena de cacao requiere acciones concretas para contrarrestar la deforestación, reducir las emisiones asociadas al uso del suelo y promover la conservación de bosques y la restauración de paisajes en las zonas productoras. Este tipo de estrategias evidencia que el manejo responsable de los sistemas de producción de cacao tiene un impacto ecológico directo, al orientar a las comunidades hacia prácticas agrícolas que disminuyen la degradación ambiental y fortalecen la resiliencia frente al cambio climático.

Desde esta perspectiva, el impacto ecológico del proyecto se relaciona con el uso de un sistema web con arquitectura Frontend Dinámica para sostener el manejo ambiental más responsable en los viveros de cacao, ya que, la plataforma ayuda con el manejo, registro y monitoreo de insumos y control de las plántulas que se traducen en prácticas que ayudan con

la reducción de residuos. El sistema impulsa a tomar decisiones que fortalecen el compromiso de los productores con la conservación del ecosistema junto a la producción sostenible del cacao a largo plazo.

CAPÍTULO II

2 MARCO TEÓRICO

2.1 Antecedentes históricos

Para Vasquez García (2017), la evolución de los sistemas web ha marcado un hito en cuanto al desarrollo de apps interactivas y eficientes; también ha apoyado el crecimiento de sectores como la agricultura y la gestión productiva. En un inicio, el desarrollo web se basaba especialmente en el renderizado del lado del servidor, donde las interacciones que el usuario usuarios hacían en la web implicaban que se recargue completamente la página, añadido a esto, había una experiencia limitada en cuanto a velocidad y dinamismo. Gracias al avance de técnicas de desarrollo web como AJAX y la aparición de nuevas tecnologías como frameworks modernos (React, Angular, Vue.js), se consolidó el paradigma de aplicaciones de una sola página (SPA), al permitir que directamente desde el navegador se pueda manipular elementos visuales dinámicamente haciendo una interacción fluida y atractiva para el usuario.

Según Rojas et al. (2020), una arquitectura frontend dinámica es la que facilita el análisis y visualización de datos en el área de agrícola, ya que optimiza la correcta toma de decisiones y permite el acceso a información primordial en tiempo real. Luzuriaga (2023) resalta que integrando arquitecturas web modernas en la agroindustria permite que los usuarios puedan interactuar con sistemas inteligentes, procesar volúmenes de datos considerables y sobre todo visualizar resultados de la producción por medio de interfaces frontend avanzadas, esto incrementa la precisión y la eficacia en cuanto a gestión agrícola se trata.

Es importante aplicar estos sistemas web desarrollados con arquitecturas frontend dinámicas por su gran capacidad para transformar la experiencia visual y operativa de los usuarios, al centralizar el control y mejorar la trazabilidad de los procedimientos operativos. El desarrollo frontend moderno tiene más enfoque en crear interfaces visuales que sean interactivas, adaptables y responsivas, que no tan solo faciliten navegar y usar una aplicación web, sino que también se lleve una gestión más ágil y que sea más precisa para obtener y dar información. Cuando se incorpora paneles de control, gráficos interactivos y visualización de procesos en tiempo real, que son propios de un desarrollo web frontend dinámico, ayuda a que el monitoreo y gestión de producción sean eficientes.

2.2 Antecedentes de investigaciones relacionadas al tema presentado

"Aplicativo web para la gestión y control de producción de un vivero de cacao" – Universidad Agraria del Ecuador

El proyecto desarrollado por Torres Chillogalli María Angélica en 2023, se focaliza en la creación de un sistema web que optimiza, registra, controla y gestiona la producción en un vivero de cacao. Dicho sistema permite que se organice la información de factores de la producción, también gestión económica como activos y gastos. Añadido a esto, incorpora herramientas visuales como son los gráficos de barras y semáforos cromáticos que visualmente da una evaluación del rendimiento y ventas.

La implementación del aplicativo web es un avance significativo en camino a la digitalización de procesos agrícolas, superando las limitaciones que traen los registros manuales y centralizando la información que nos relevante para la producción. El sistema de Torres Chillogalli usa visualización cromática, es decir, que ofrece una referencia clara sobre el estado productivo según colores que representen buen estado, advertencias y peligro según los datos indiquen que se presente alguno de estos estados, lo que contribuye a que la gestión sea más informada y por ende más estratégica. Como señala la autora. "Expandirse, dejando a un lado la recolección de datos de forma manual, adentrándose al mundo computacional, dando uso de algún sistema de información, que fortalezca la toma de decisiones a futuro" (Torres Chillogalli, 2023).

"Desarrollar sistema web para la gestión administrativa de plantas de cacao en el Vivero Díaz en Zhucay" – Universidad Agraria del Ecuador

Mendoza Fernández Marco Vinicio aborda un problema puntual: la ausencia de una herramienta tecnológica para gestionar y dar seguimiento a las plantas en el Vivero Díaz, en Zhucay. El sistema web que desarrolló permite registrar gastos, llevar el control de la cantidad de plantas producidas y gestionar las ventas, usando PHP y MySQL como base tecnológica.

En este proyecto se puede evidenciar cómo la digitalización mediante un sistema web transforma la gestión administrativa y productiva, y hace que el acceso a información precisa y actualizada sea mucho más fácil. Automatizar procesos y que se usen tecnologías libres ayudan a que esta solución digital sea sostenible y escalable. Como se menciona Mendoza en su tesis, "la propuesta permitió analizar la situación actual del negocio y mediante técnicas como entrevista y observación se logró recolectar la información necesaria para conocer las necesidades y requerimientos de la organización" (Mendoza Fernández, 2022).

"Sistema web de gestión de las actividades de producción en el cultivo de cacao para la finca cacaotera Álvarez El Triunfo" – Escuela Superior Politécnica del Litoral

El proyecto de Villa López Bryan José plantea una solución tecnológica para la gestión de las actividades productivas que se realizan en una finca cacaotera. Este sistema web permite planificar, registrar y monitorear todas las etapas que se llevan a cabo en un proceso productivo, desde que se siembra hasta que se cosecha el cacao, el utiliza en su proyecto herramientas como PHP, JavaScript, HTML y MySQL. El proyecto de Villa resalta que es importante la retroalimentación continua con usuarios reales para así asegurar una correcta funcionalidad y que estos usuarios tengan un sentido de pertenencia con el programa.

La experiencia que se obtuvo de este proyecto muestra que al implementar sistemas web en la producción de cacao no solo mejora la organización y el control, sino que también ayuda con la participación de los usuarios en el perfeccionamiento de la herramienta, como indica el autor del proyecto, "manteniendo retroalimentación continua con la propietaria para el cumplimiento de sus requerimientos" (Villa López, 2023). Esto nos garantiza que el programa nos responda a situaciones reales en el entorno productivo agrícola.

2.3 Definiciones conceptuales

2.3.1 Sistema web con arquitectura frontend dinámica.

2.3.1.1 Historia y evolución de la programación web.

La programación se ha transformado con Internet dejó de ser un canal solo para consulta. Durante la era Web 1.0, las páginas se construían con páginas estáticas con poca interactividad con el usuario. La llegada de la Web 2.0 cambió en lo dinámico, incorporando

el contenido de los usuarios, la interacción bidireccional y servicios interactivos. La Web 3.0, en cambio, propone una gestión más semántica de los datos basada en tecnologías semánticas. Ibrahim (2021) explica esta evolución como un proceso constante en el que la web expande sus funciones y en la diversidad de interacciones que toma una plataforma, capaz de brindar experiencias personalizadas y conectadas.

Anwar (2022) comenta que el progreso hacia una web semántica también refleja la necesidad de comprender la información y vincularla de una manera que sea más útil para las personas y los sistemas informáticos. No basta con almacenar documentos o mostrar contenidos si las aplicaciones actuales necesitan capturar contextos, integrar datos de diferentes fuentes e interactuar con los dispositivos conectados. En la escena, la interacción entre Web 3.0 e Internet de las cosas, abre posibilidades para construir servicios distribuidos que reaccionen a situaciones específicas, pero existen desafíos relacionados con la privacidad, la interoperabilidad y la seguridad de estos sistemas. Anwar también sostiene, que el crecimiento de estas tecnologías dependerá de cómo se superen estos desafíos sin perder accesibilidad ni confianza.

2.3.1.2 Arquitectura cliente-servidor en sistemas web.

La arquitectura cliente-servidor no es reciente (ha estado operando décadas), pero como funciona cambió. Las necesidades de las aplicaciones actuales exigen que, en lugar de recibir un documento HTML por cada click que el usuario da, el navegador reciba solo lo que necesita para renderizar la pantalla. El estándar REST ha dominado el panorama por ello es flexible y permite unir interfaces y tecnologías distintas, según el examen que hizo Karlsson y sus colegas (2024) sobre las aplicaciones de API REST en producciones industriales. Ya, eso no significa automáticamente que una API que lleva ese nombre sea eficiente, cómo se estructuran los recursos, cómo se procesan las solicitudes que la emite, y cuán claras estaban las respuestas, son decisiones que pueden diferir el éxito o el fallo de la conexión cliente-servidor.

Separar el frontend y backend es un beneficio, ya que cada uno funciona a su velocidad, un equipo mejora formularios y pantallas, otras reglas de negocio y el almacenamiento sin pisarse. Pero esto también tiene un coste: esa separación da escenarios donde una conexión puede fallar, la API le da información incompleta, le da con una nueva sin avisarle, o manejes mal una excepción, el cliente deja de funcionar, aunque este ni se toca. Ehsan y sus

colaboradores (2022) son transparentes: “Las pruebas no basta con probar las respuestas exitosas, sino también los datos que da, fallos de autenticación y solicitudes que no vienen al caso”. Cada arquitectura acoplada simplifica el mantenimiento, sí, pero solo si se mantiene la comunicación del contrato acordado.

2.3.1.3 HTML5, CSS3 y JavaScript en el desarrollo web moderno.

Antes de pensar en animaciones o funciones vistosas, una aplicación necesita algo más básico: una estructura que se entienda. Para eso está HTML5, que organiza el contenido con etiquetas que dicen qué función cumple cada sección, mientras CSS3 se encarga de la distribución visual y de que todo se vea bien en distintos dispositivos. Jain (2024) y su equipo, al analizar los fundamentos del desarrollo web moderno, destaca estas dos tecnologías como pilares de la profesión, ya que, en conjunto, permiten y mantienen la capacidad de crear páginas responsivas sin perder estructura ni legibilidad..

Para Can y Göktaş (2023) JavaScript opera en otro plano: convierte esa estructura en una interfaz interactiva, por lo que acciones como validar un formulario, consultar un servicio sin recargar la página o actualizar una tabla instantáneamente dependen, en la mayoría de los casos, de este lenguaje. La evolución de JavaScript también explica por qué hoy existen ecosistemas completos para la construcción de aplicaciones del lado del cliente, y como resultado, frameworks como Angular, React y Vue.js, entre otros, son derivados de JavaScript, pero tienen como común denominador que, organizan la lógica, los componentes y las actualizaciones de la vista de maneras muy diferentes. Esta variedad permite elegir la herramienta adecuada para cada proyecto, pero no exime de dominar los estándares básicos, porque un framework organiza el trabajo, pero si el HTML, los estilos o la lógica están mal contruidos, ese orden es solo aparente.

2.3.1.4 Introducción a frameworks y librerías frontend reactivas.

Si de frameworks se trata, Karić y Durmić (Karić & Durmić, 2024) llegan a una conclusión en su comparación de tecnologías, que es que la conveniencia de un framework depende del tamaño del proyecto, de lo que pide, y de la experiencia que tenga el equipo que lo va a usar. Elegir entre Angular, React y Vue no debería depender de cuál aparece más en las búsquedas de empleo o cuál genera más ruido en internet. Según la investigación de Karić y

Durmić, existen tres librerías principales con diferentes propiedades para tener en cuenta al desarrollar software web: Angular es un framework completo con un conjunto de herramientas integradas, React es una biblioteca basada en componentes que deja la elección de una herramienta al desarrollador, por último, Vue puede integrarse por partes en un proyecto sin necesidad de sobrescribirlo por completo. Las diferencias entre estas tecnologías influyen directamente en la curva de aprendizaje, en cómo se organiza el código y, por consiguiente, en el mantenimiento de la aplicación web.

Para Piastou (2023) dice que la reactividad son los cambios se ven sin interrupciones raras, interrupciones como, si un dato cambia, la interfaz actualiza lo que tiene que actualizar y deja el resto de la vista intacta. Detrás de esa simpleza hay un proceso de detección, comparación y renderizado que, mal administrado, sí puede afectar el rendimiento.

2.3.1.5 Modelos de arquitectura MVVM y SPA (Single Page Application).

Jayaraman y Kumar (2024) dieron por concluido que en webs de una sola página no hay necesidad de descargar el documento HTML completo para cada acción, afirmando que solo se cargaría la estructura principal de la página en el DOM, mientras que el contenido se actualizaría a medida que el usuario navega por las diferentes secciones del aplicativo web, esto es usado ampliamente para plataformas de uso diario como paneles de administración, herramientas de gestión, apps de monitoreo y seguimiento, entre otros. Para Jayaraman y Kumar la velocidad de la aplicación no refleja la realidad total en cuanto a su rendimiento; si bien el tiempo de carga inicial puede ser prolongado, una implementación bien pensada de rutas y gestión de estado puede hacer que la aplicación parezca rápida, aunque los problemas solo se harán evidentes más adelante. Se plantea qué tanto pueden mejorar el rendimiento y el orden de una SPA las novedades que fueron sumando las nuevas tecnologías de desarrollo web. Adoptar este modelo, entonces, pide bastante más que sacarse de encima las recargas tradicionales.

“No todas las tecnologías aplican esta separación con el mismo rigor. Al comparar las arquitecturas de Angular, React y Vue, aparecen diferencias claras en cómo cada una vincula el estado con los elementos visuales” (Can & Göktaş, 2023). Además, Can y Göktaş en 2023 afirmaron que la sincronización automática evita el código duplicado, pero que esto debe controlarse rigurosamente ya que un enlace bidireccional mal implementado dificulta

enormemente la comprensión de dónde se produjo realmente el cambio. MVVM es beneficioso no solo porque actualiza la pantalla de forma más eficiente, sino también porque esta evita que la lógica de presentación se mezcle forma innecesaria con las reglas principales del sistema. Asimismo, MVVM fomenta la organización de la lógica antes de que la interfaz se convierta en un caos incomprensible de eventos, validaciones y cambios.

2.3.1.6 Componentización y reutilización de vistas con frameworks modernos

Hay un criterio al que apunta Kothapalli (2021), es que, la componentización y reutilización de vistas con frameworks modernos como un formulario que se usa en cinco módulos distintos y no debería escribirse cinco veces, basados en esa idea, parte la arquitectura basada en componentes, es decir, cada elemento tiene una responsabilidad puntual la cual se puede probar por separado y que después se puede combinar con otros componentes para armar vistas mas grandes. Este concepto es el que ha acompañado el crecimiento de las aplicaciones frontend ya que permitió manejar interfaces cada vez más complejas.

Esta lógica de construcción modular es para escalar con aplicaciones micro-frontends, que en vez de dividirse solamente en botones, tablas o formularios, las apps web se dividen en áreas funcionales completas que cada equipo de trabajo puede desarrollar y desplegar con más autonomía, esta lógica de modularidad es una necesidad que Savani (2023) conecta directamente con la escala que están alcanzando los proyectos web actuales. Sin embargo, esta independencia no es infinita, ya que los módulos deben compartir mecanismos de navegación, comunicación y presentación para que el usuario no tenga la sensación de estar navegando por una aplicación defectuosa. Además, es necesario gestionar las versiones y las dependencias compartidas. En un sistema pequeño, la infraestructura a veces resulta redundante o demasiado compleja, pero en uno grande puede evitar que un solo cambio requiera reconstruir y volver a desplegar toda la interfaz.

2.3.1.7 Principios de diseño de interfaces reactivas y usabilidad.

Para Weichbroth (2024) activar un botón que guarde/registre algo y quedarse esperando sin saber que paso lo que tenía que pasar duran unos segundos, pero esos pocos segundos alcanzan para generar desconfianza en un sistema, por eso se indica que una interfaz reactiva tiene que mostrar lo que está pasando en el sistema, detalles como indicadores de procesos,

confirmaciones o mensaje de errores claros son claves para una buena experiencia de usuario. El autor de la investigación describe las pruebas de usabilidad como el proceso que permite observar precisamente estas interacciones y detectar obstáculos desde el punto de vista de quien usa la aplicación, no de quien la construyó.

Tampoco sirve de mucho una interfaz que solo funciona bien en el monitor donde se diseñó. En un teléfono o una tablet cambia el espacio disponible, cambia cómo se navega, y hasta cambia el contexto en el que alguien está mirando esa pantalla. Parlakkiliç (2021) hizo una evaluación de sitios académicos y encontró una relación significativa entre el diseño responsivo y la usabilidad percibida, concluyó que adaptar una aplicación es más complicado que achicar la tipografía: puede implicar reorganizar columnas, simplificar menús, o cambiar qué acciones se ven primero. Las unidades relativas, las cuadrículas flexibles y las media queries ayudan con eso, pero al final las decisiones hay que probarlas directamente en pantallas distintas. Un diseño técnicamente adaptable puede seguir siendo incómodo si no toma en cuenta que la forma de interactuar también cambia.

2.3.1.8 Accesibilidad y compatibilidad de interfaces web.

En palabras de Acosta-Vargas y su equipo (2021), pensar en accesibilidad desde el arranque cambia decisiones que, de otra forma, parecerían puramente visuales, cosas como el contraste entre el texto y el fondo, cómo están armados los encabezados, la descripción de una imagen: todo eso decide quién puede entender el contenido y en qué condiciones. De las pruebas hechas por los autores para su investigación, indican que se siguen presentando las mismas barreras de acceso para usuarios con discapacidad, por eso mismo aclaran que la accesibilidad se debe pensar y desarrollar desde un principio y no como un añadido después de que el aplicativo sea lanzado, declarando que deberían tratarse como un extra pensado para un grupo reducido.

Ara y sus colegas (2024) son tajantes en comentar que se le puede avisar a las tecnologías de asistencia el nombre, la función o el estado de un control dinámico, pero no arregla sola una estructura HTML mal armada. Si existe un elemento nativo que sirve para eso, casi siempre conviene usarlo en vez de recrear su comportamiento a fuerza de atributos extra. en su revisión sobre evaluación de accesibilidad web: ninguna técnica por sí sola detecta todas las barreras. Las herramientas automáticas atrapan errores puntuales; una revisión manual

chequea la navegación por teclado, el orden del foco, qué tan claros son los mensajes. Y la participación de usuarios reales todavía encuentra problemas que los dos métodos anteriores dejaron pasar. La compatibilidad de verdad sale de combinar estas tres cosas, no de aprobar un solo validador automático.

2.3.1.9 Gestión automatizada del flujo de trabajo y recursos.

“Automatizar esto baja la carga administrativa y evita que el seguimiento dependa de que alguien se acuerde de mandar un recordatorio, incorporar algoritmos inteligentes puede llevar esa automatización más lejos todavía, con análisis y decisiones que se ajustan a la información disponible” (Nama, 2022) .Basado en esto es conveniente aclarar que no se debe confundir eficiencia con autonomía total: hay excepciones, datos incompletos y situaciones ambiguas que necesitan que alguien revise a mano.

Mylsamy y Mehrotra (2025) dicen que mientras la automatización que vimos recién ordena tareas del negocio, CI/CD trabaja sobre otra cosa: el ciclo de construir y publicar el software. Cada cambio que llega al repositorio puede disparar una cadena de compilaciones, pruebas y chequeos; si algo falla, el despliegue se frena antes de llegar a los usuarios, por lo que se puede decir que, si bien el objetivo final de estos flujos de trabajar es realizar despliegues con mas frecuencia, su valor va más allá de entregar actualizaciones con una mayor velocidad, ya que se puede rastrear que versión se implementó, que comprobaciones se superaron y donde se produce el error, y para que todo esto funcione el flujo de trabajo debe reflejar los riesgos reales del proyecto, por lo que trabajar con simples comprobaciones irrelevantes genera una falsa sensación de seguridad, ya que consume recursos y no aporta nada a la confianza real del producto.

2.3.1.10 Pruebas automatizadas y aseguramiento de calidad (QA).

En palabras de Bhimanapati y su equipo (2024) , una función puede andar perfecto sola y fallar apenas se conecta con el resto de la aplicación. Por eso las pruebas unitarias y las de extremo a extremo contestan preguntas distintas: las primeras chequean pedazos chicos de lógica y ayudan a encontrar el error rápido; las segundas simulan el recorrido completo, desde que el usuario hace clic hasta que ve el resultado en la pantalla o en el servidor.

Para Moń y Pańczyk, (2025), Cypress, Selenium y Playwright tiene un mismo propósito que es automatizar las interacciones con el navegador, sin embargo, las implementaciones internas son las que marcan diferencia y no son equivalentes en cuando a flujo de trabajo. Debido a esto la elección de la herramienta debe basarse en el tipo de aplicación que se esté probando y en la capacidad técnicas del equipo, dado que el mantenimiento es fundamental porque una prueba que falla sin motivo aparente en interfaces que cambian rápidamente o que utiliza selectores frágiles, deja de ser una simple prueba y comienza a ser una carga para el equipo de desarrollo. Por lo tanto, lo que se suponía que protegía el proceso de desarrollo ahora lo ralentiza.

2.3.2 Planificación de producción.

2.3.2.1 Evolución de la gestión productiva en viveros agrícolas.

Urbina-Céspedes, Campos-Cerdas y Núñez-Parrales (2026) sostienen que la planificación de la producción agrícola ha dejado de depender únicamente de la experiencia del productor para apoyarse en modelos cuantitativos que combinan pronóstico y optimización. Su método combina el análisis de series temporales y la programación lineal para pronosticar la tasa de producción a lo largo del tiempo y determinar la asignación óptima de tierras cultivables a cada cultivo para obtener el máximo rendimiento esperado dentro de las limitaciones de recursos. Los autores concluyen que este tipo de modelo puede utilizarse para generar escenarios predictivos a partir de datos históricos oficiales, que pueden emplearse tanto para la planificación estratégica como operativa.

Alcanzar niveles de producción crecientes y sostenidos exige asumir la planificación y el control de la gestión productiva como una disciplina científica, apoyada en principios como la selección adecuada del recurso humano, la asignación de responsabilidades, el establecimiento de estándares de producción y las políticas de incentivos laborales (Palomeque Solano, 2022).

2.3.2.2 Importancia de los sistemas web en la producción de plántulas de cacao

Kitole, Mkuna y Sesabo (2024) , a partir de un estudio empírico sobre el sector agrícola de Tanzania, muestran que la digitalización agrícola facilita el acceso a datos sobre clima,

precios y servicios de extensión, con una incidencia positiva sobre el bienestar económico de los pequeños productores. El estudio sugiere que el acceso al crédito, la educación, el apoyo gubernamental y los servicios de extensión son los factores más críticos para esta integración digital, y también afirma que es imposible aprovechar todo el potencial del sector agroindustrial y su cadena de valor sin un proceso de digitalización eficaz. Con base en los resultados, los investigadores concluyen que la transformación digital de la agricultura es, en esencia, un cambio radical en el acceso a la información y en las decisiones de producción, y por lo tanto va más allá de una simple innovación tecnológica. Este argumento también se aplica a los viveros; por consiguiente, su transformación digital es parte integral de este cambio estructural más amplio.

De León López y Victorino Ramírez (2024) surge una perspectiva es más conservadora: si bien la agricultura digital ofrece formas más eficientes y sostenibles de producir alimentos, la adopción de esta tecnología también está determinada por factores económicos, sociales y culturales, que pueden generar resultados positivos, pero también muy negativos para los productores tradicionales, especialmente en áreas donde la agricultura convencional sigue siendo la más extendida.

2.3.2.3 Modelos digitales de cronogramas y gestión de etapas productivas.

Un sistema web orientado a la gestión del mantenimiento en cultivos de cacao redujo en 74,5 % el tiempo dedicado al registro de actividades (de 47 a 12 minutos) y disminuyó en 72 % los errores en el manejo de inventarios (Alvarado J. C., 2025) . Con esos números, los autores confirman la utilidad de este tipo de plataformas para optimizar procesos que antes se realizaban de forma manual. No obstante, el mismo estudio identificó que persistían prácticas manuales en una parte importante de las labores, lo que llevó a sus responsables a sugerir que la efectividad de estas herramientas digitales depende, más que de su sofisticación técnica, de qué tan bien se adapten a las condiciones y capacidades locales de los productores.

En Orellana, Ecuador, Vivanco, Iza, Sailema y Encarnación (2024) desarrollaron una aplicación web progresiva para la gestión de pronósticos agrícolas en plantaciones de cacao, con el propósito declarado de acercar herramientas digitales de planificación a productores que tradicionalmente dependen de la experiencia empírica para anticipar variables productivas. Los autores también señalan que su propuesta permite el acceso a la información y predicciones en

situaciones donde la conectividad es limitada o nula, estos casos generalmente dados en zonas rurales productoras de cacao, por esto mismo destacan la necesidad de desarrollar sistemas web que se adapten a infraestructuras tecnológicas que se puedan aplicar en ese entorno real, considerando que darle solución a ese problema real de la conectividad es ideal para que el sistema tenga una función vital a la hora de que el usuario lo emplee en actividades de campo afines al rubro donde no se tenga las comodidades de la conectividad.

2.3.2.4 Programación y registro de actividades productivas

Custodio, Billones, Concepcion y Vicerra (2024) diseñaron un modelo de programación lineal para identificar las temporadas más rentables de cosecha en cultivos de arroz y maíz en Filipinas. Esto demuestra que un problema de planificación de actividades agrícolas puede formalizarse como un modelo matemático, cuyo objetivo es maximizar el valor económico de la producción en respuesta a las variaciones de precios y costes. En el caso de estudio de los autores, se registró un aumento del 13,8 % en la rentabilidad regional, lo que ellos consideraron como prueba de que los programas de producción, si se basan en modelos de optimización digital, pueden convertirse en una herramienta estratégica para la toma de decisiones, en lugar de una simple lista de fechas.

Esteso, Alemany, Ortiz e Iannacone (2022) proponen, por su parte, un marco conceptual y un modelo de optimización multiobjetivo para planificar la siembra y la cosecha de plantas cuyo valor depende de la concentración de determinadas moléculas y de un tiempo mínimo entre cosechas sucesivas. El modelo que plantean gestiona de manera integrada las distintas etapas del ciclo productivo (siembra, cosecha, poscosecha, distribución y almacenamiento), evitando la descoordinación que surge cuando cada productor planifica de forma individual y equilibrando los costos de la cadena de suministro con la calidad del producto final.

2.3.2.5 Inventarios automatizados de insumos y materiales en viveros.

Sánchez-Calle y Castillo-Armas (2022) , en una revisión bibliográfica sobre el efecto de distintos algoritmos en la agricultura, identifican que la inteligencia artificial se ha convertido en un componente esencial de la evolución tecnológica de la agroindustria, al brindar al productor información precisa para tomar decisiones sobre sus procesos. Sin

embargo, los propios autores señalan que en contextos como el agrícola peruano todavía son pocas las empresas que han incorporado este tipo de tecnologías, y que persisten deficiencias en procesos como la selección y clasificación de productos, lo que genera pérdidas evitables.

En un caso aplicado directamente a un vivero, Mayorga Vásquez, Riccardi Martillo, Bermeo Almeida y Guevara Arias (2022) diseñaron un sistema web para automatizar los procesos administrativos y de producción de los viveros del cantón Milagro, en Ecuador. El módulo que describen administra la información sobre las características de las plantas, el lote de producción y las etapas de crecimiento, y registra las tareas ejecutadas junto con la asignación de herramientas e insumos a cada trabajador.

2.3.2.6 Trazabilidad y seguimiento de recursos en tiempo real

Ese mismo sistema web, el que Mayorga Vásquez et al. (2022) diseñaron para los viveros del cantón Milagro, incluye también un módulo de inventario que merece atención aparte. Este módulo registra las plantas según su etapa de crecimiento (tipo, cantidad, ubicación, entradas y salidas) además de otros insumos como semillas, tierra de siembra y fertilizantes, y genera una alerta automática cuando las existencias descienden por debajo de un nivel mínimo previamente configurado, de acuerdo con lo que describen los autores.

Los problemas de sobreinventario, agotamiento de existencias, pérdida de productos y registros conflictivos, que generaban tiempos excesivos de atención al cliente en una empresa comercializadora de productos de riego, motivaron el desarrollo de una metodología de control de inventario que combina un sistema de clasificación de productos, la reorganización física del almacén, controles internos y un sistema computarizado de seguimiento (Corella-Parra, 2023) .

2.3.2.7 Registro y análisis de indicadores productivos clave.

Gómez Gómez, Parra Urueta y Guzmán Pineda (2025) desarrollaron un sistema de trazabilidad para el proceso agroindustrial del cacao (desde la semilla hasta la comercialización) que combina una aplicación web con una API REST para la gestión comercial y un dispositivo IoT con sensores que monitorea el proceso de fermentación del grano conforme a la normativa técnica vigente. El proyecto, que los autores implementaron en

Tierralta (Córdoba, Colombia), incluyó además una línea de tiempo interactiva para visualizar cada etapa del proceso. Según reportan, esto permitió sistematizar y estandarizar procedimientos que antes dependían del registro manual, con mejoras en la calidad del producto y la generación de datos útiles para la toma de decisiones futuras.

En un plano más amplio, el de la cadena de suministro en general, Alzate y Giraldo (2023) realizaron un mapeo bibliométrico de las investigaciones sobre blockchain publicadas entre 2000 y 2022, e identificaron tres tendencias principales de aplicación: transparencia, trazabilidad y seguridad de las transacciones entre los actores involucrados. Los autores argumentan que la tecnología proporciona acceso libre e inmutable al historial de transacciones de los bienes, lo cual es especialmente relevante para el seguimiento en tiempo real de los recursos, y al reducir la posibilidad de manipular los registros, permite a todos los actores de la cadena, incluidos los productores agrícolas, como los viveros, verificar el origen y el estado de los insumos y productos, sin la mediación de un tercero.

2.3.2.8 Paneles interactivos y reportes automatizados para la toma de decisiones.

En palabras de Estrella Quispe (2023) una aplicación tecnológica de indicadores para sistemas de gestión productiva, basada en tecnologías de sistemas embebidos, distingue entre indicadores de gestión de alto nivel (orientados al desempeño general de la organización) e indicadores de bajo nivel, enfocados en procesos o áreas específicas como producción, ventas o atención al cliente, por lo mismo la autora afirma que el diseño de los indicadores de rendimiento (KPI) es fundamental al hablar del seguimiento de cualquier proceso productivo, pero su implementación en tiempo real no es tan sencilla, ya que para diseñar un aplicativo de gestión es crucial elegir los indicadores adecuados desde un principio. Este mismo principio se aplica directamente al seguimiento de indicadores en la producción de plántulas.

En un contexto agrícola institucional, Valecillos Barreto y Sarmiento Gil (2026) propusieron un sistema de indicadores de gestión (fundamentado en la Administración por Objetivos y en los estándares SMART) para fortalecer la planificación operativa de doce unidades de producción social agrícola en el estado Trujillo, Venezuela.

2.3.2.9 Tecnologías emergentes para la automatización y monitoreo en viveros de cacao.

Falcón-Suárez, Betancourt-Perera, Liriano-González y Pérez-Hernández (2023) desarrollaron una aplicación web, el Sistema de Análisis de Datos Agrícolas (SADA), que unifica los servicios de socialización de información y automatiza funcionalidades como la adecuación de variedades de cultivo y el diseño de rotaciones, generando reportes a partir de los parámetros edafoclimáticos y fisiogeográficos que introduce el usuario.

En Riobamba, Ecuador, Chiriboga-Zamora, Romero-Flores, Tapia-Hermida y Santander-Samaniego (2024) investigaron el impacto de las herramientas de Business Intelligence en la gestión de las asociaciones de producción agrícola y constatan que el 62% de los miembros desconocían el concepto. El capital limitado y la falta de conocimientos tecnológicos constituyen obstáculos para la adopción de la herramienta, ya que los autores concluyen que herramientas como Tableau permiten integrar diversas fuentes de datos, realizar un seguimiento de los KPI y crear paneles personalizados con capacidades de análisis en tiempo real, lo que simplifica el proceso de toma de decisiones colaborativa y genera informes más profesionales.

2.3.2.10 Estrategias sostenibles y certificación ambiental en la producción de plántulas de cacao.

Un Modelo Integrado de Adopción de Tecnologías en la Agricultura (MIATA), diseñado para analizar los factores que determinan la incorporación de inteligencia artificial e Internet de las Cosas entre productores de cacao de la provincia de El Oro, Ecuador, encontró que el 70,7 % de los agricultores encuestados no utilizaba ningún tipo de tecnología en sus fincas, mientras que solo el 29,3 % había incorporado herramientas básicas como sistemas de riego (Izquierdo, 2025).

Khoddamzadeh y Souza Costa (2023) evaluaron, en un caso experimental centrado específicamente en la etapa de vivero, distintas dosis de fertilizante de liberación lenta en plántulas de cacao. Para ello utilizaron sensores ópticos (el SPAD, el atLEAF y el índice de vegetación NDVI) con los que monitorearon de manera no destructiva el contenido de clorofila

de las plantas durante seis meses. Los resultados que reportan muestran que las dosis más altas de fertilizante incrementaban la concentración de nitrógeno en el agua de lixiviación, con el consiguiente riesgo ambiental, mientras que un manejo más moderado, apoyado en la lectura constante de los sensores, permitía sostener un crecimiento adecuado de las plántulas con menor riesgo de contaminación.

2.4 Metodología Propuesta.

2.4.1 Metodologías Ágiles (Scrum):

Schwaber y Sutherland (2020) , en la guía oficial del framework, definen Scrum como un proceso empírico organizado en iteraciones cortas llamadas Sprints, donde la inspección y la adaptación constantes permiten que el producto evolucione según la retroalimentación directa de quien lo usa. Esta base empírica (decisiones tomadas a partir de la observación y la experiencia real durante el desarrollo, y no de una planificación cerrada desde el inicio) es justamente lo que Žáček et al. (2025) destacan como la razón de la adopción masiva de Scrum en la industria del software actual, aunque los mismos autores advierten que esa misma flexibilidad exige disciplina en cada uno de sus eventos para no derivar en desorden.

Para este proyecto se adoptó el marco de trabajo Scrum, organizado en ciclos iterativos de dos semanas. Durante cada Sprint se priorizaron las funcionalidades del módulo de planificación, que fue validado mediante pruebas con los propios usuarios. Este enfoque permitió ajustar la interfaz dinámica a partir de la experiencia del personal del Vivero Lastenia al usar la plataforma web, garantizando que la herramienta resultara práctica y eficiente en el día a día. A partir de la aplicación de esta metodología se adaptaron las siguientes fases:

2.4.1.1 Sprint Planning (Planificación del Sprint).

Žáček et al. (2025) documentan que la fase de planificación del Sprint es, en la práctica, uno de los puntos más frágiles de Scrum: la selección de tareas suele quedar librada a la intuición del equipo, sin un método que respalde esas decisiones. Los autores retoman el caso de Mahnic (2011) , quien encontró que los equipos completaban en promedio solo el 42 % de las tareas planificadas en su primer Sprint, cifra que subía a cerca del 75 % recién en el segundo. Este hallazgo respalda la importancia de un Sprint Planning bien ejecutado, con objetivos realistas acordados entre el Product Owner y el equipo antes de comprometerse con el ciclo.

Para el proyecto se realizaron sesiones de planificación cada dos semanas, en las que se priorizaron las funcionalidades del módulo de planificación. En esas reuniones se segmentó de manera lógica el desarrollo de la estructura frontend, según el algoritmo necesario para la planificación y gestión de los lotes del vivero.

2.4.1.2 Daily Scrum (Reunión Diaria).

Stray, Sjøberg y Dybå (2016) , tras observar doce equipos de software en empresas distribuidas entre Malasia, Noruega, Polonia y el Reino Unido, encontraron que compartir información y poder discutir problemas en el momento son los factores que más valoran los desarrolladores en este evento de máximo 15 minutos. Los mismos autores advierten que la reunión pierde ese valor cuando se convierte en un reporte de estado para el jefe, en vez de una sincronización real entre el equipo.

En el desarrollo del proyecto se realizaron reuniones breves para monitorear el avance de la codificación. Estas sesiones permitieron identificar a tiempo dificultades técnicas en distintos componentes del módulo, lo que garantizó que el flujo de trabajo no se detuviera y que se cumplieran los plazos establecidos.

2.4.1.3 Sprint Review (Revisión del Sprint).

Una encuesta a más de 15.000 miembros de equipos ágiles y cerca de 1.800 interesados permitió medir la calidad del Sprint Review a partir de dos preguntas concretas: si los interesados probaban lo construido durante el evento, y si esa retroalimentación generaba cambios reales en el Product Backlog (Verwijns, 2024) . Los resultados confirman que un Sprint Review bien ejecutado es el mecanismo central para validar que el software cumple las expectativas del usuario final y los objetivos del negocio.

Al finalizar cada ciclo se presentaron los avances funcionales a los encargados del Vivero Lastenia. Esto permitió validar si la visualización del cronograma en el frontend resultaba intuitiva para los técnicos, e integrar ajustes visuales inmediatos según sus sugerencias antes de pasar a la siguiente fase.

2.4.1.4 Sprint Retrospective (Retrospectiva del Sprint).

Un análisis de 37 retrospectivas de equipo a lo largo de casi tres años, en una organización ágil distribuida, mostró que la mayoría de las discusiones giraban en torno a temas que el equipo podía controlar y mejorar directamente (Lehtinen, 2017) . Sin embargo, según los mismos autores, algunas conversaciones se repiten sprint tras sprint sin resolverse, cuando el problema de fondo escapa a la capacidad de decisión del equipo.

Tras cada entrega se analizaron los cuellos de botella detectados durante el desarrollo del sistema. Se identificó la necesidad de mejorar la documentación del código en el módulo de inventarios, lo que se tradujo en una mayor velocidad de respuesta y una mejor organización técnica en las etapas finales del proyecto.

2.5 Conclusiones.

En este sentido, el estudio de los sistemas web y de las tecnologías orientadas al desarrollo de interfaces dinámicas permite establecer que su utilidad no se limita a la presentación de información, sino que también posibilita organizar, gestionar y facilitar el acceso a los procesos que requieren interacción constante con los usuarios. La incorporación de estructuras dinámicas, componentes reutilizables y mecanismos de automatización favorece la construcción de soluciones capaces de adaptarse a diferentes necesidades, manteniendo condiciones adecuadas de funcionalidad, usabilidad y seguridad. De esta manera, los fundamentos analizados permiten reconocer la importancia de contar con herramientas web que faciliten la gestión de información y contribuyan a responder de manera ordenada a los requerimientos presentes en un entorno productivo.

Por otra parte, la planificación de la producción constituye un elemento fundamental para mantener un adecuado control de las actividades, recursos y etapas que intervienen en los procesos agrícolas, especialmente cuando se requiere conocer su desarrollo y comportamiento de manera continua. Una gestión organizada permite establecer las actividades que deben realizarse, controlar los recursos disponibles, dar seguimiento a los procesos y disponer de información que facilite la toma de decisiones ante las distintas necesidades que puedan presentarse.

CAPÍTULO III

3 MARCO INVESTIGATIVO

3.1 Introducción.

Este capítulo define el enfoque, la estructura y los procedimientos que guiarán la investigación y el desarrollo del Sistema Web con Arquitecturas Frontend Dinámicas para la planificación de producción del Vivero Lastenia. Dada la naturaleza del trabajo, fue necesario utilizar una combinación de tipos y métodos de investigación: investigación de campo, investigación bibliográfica y métodos cualitativos y cuantitativos respectivos, ya que el objetivo principal es garantizar que la solución tecnológica no solo sea moderna y robusta, sino que también responda eficazmente a las necesidades operativas de las plántulas de cacao.

3.2 Tipo de Investigación.

La presente investigación se clasifica en tres tipos principales, los cuales garantizan la rigurosidad tanto en la fundamentación teórica como en la aplicación práctica de la solución:

3.2.1 Investigación Bibliográfica:

Kothari (2004) caracteriza la investigación bibliográfica como una de las modalidades fundamentales del método científico, orientada a la búsqueda, recopilación y análisis crítico de información ya publicada en fuentes impresas o digitales.

Para el trabajo realizado, basado en una investigación bibliográfica, se dio la elección de una arquitectura frontend dinámica y su respectivo patrón MVVM, ya que al analizar repositorios académicos y documentación técnica de frameworks modernos, se determinó las mejores prácticas para desarrollo de aplicaciones reactivas, e igualmente fundamentados por fuentes especializadas en gestión agrícola, se estructuraron fundamentos teóricos de los algoritmos de planificación que se utilizaron en el sistema web.

3.2.2 Investigación de Campo:

La investigación de campo consiste en la recolección de datos primarios directamente de la realidad donde ocurren los hechos, sin manipular las variables de estudio (Creswell, 2018) .

Se llevó a cabo una investigación de campo mediante visitas directas a las instalaciones del Vivero Lastenia para observar de primera mano el flujo de producción de las plántulas de cacao. Durante estas visitas se interactuó con el personal técnico para identificar los cuellos de botella en la planificación manual y la gestión de inventarios, y con eso se ajustaron los requisitos funcionales del software a las condiciones reales del vivero.

3.3 Métodos de Investigación

Para alcanzar los objetivos planteados, la investigación se apoyó en métodos teóricos y empíricos:

3.3.1 Método Cualitativo

Busetto, Wick y Gumbinger (2020) señalan que el método cualitativo busca comprender fenómenos complejos desde la perspectiva de quienes los viven, principalmente a través de entrevistas semiestructuradas, observación y estudio de documentos.

Se aplicó este método para identificar los requerimientos funcionales y las expectativas de experiencia de usuario en el Vivero Lastenia. Los principales problemas que el administrador y los técnicos tenían con las plántulas de cacao se obtuvieron directamente de ellos, y también se cumplieron sus requisitos para la interfaz dinámica, que luego se utilizaron para desarrollar las historias de usuario para los desarrolladores.

3.3.2 Método Cuantitativo

Para Rahi (2017) el método cuantitativo se define como un método sistemático que utiliza la recopilación de datos numéricos y el análisis estadístico para encontrar patrones, poner a prueba teorías y medir la magnitud de los fenómenos estudiados. El autor afirma que este método, respaldado por instrumentos como encuestas, permite obtener resultados

generalizables y verificables mediante la medición rigurosa de indicadores de desempeño y variables del proceso analizado.

En el proyecto, este método se utilizó para el análisis de datos críticos para la producción de viveros, como la cantidad exacta de insumos requeridos por lote, los porcentajes históricos de pérdidas y los tiempos promedio de germinación e injerto. Con esos datos se definió la estructura de la base de datos y se programó la lógica de los algoritmos de planificación, además de automatizar el cálculo de stock y generar reportes estadísticos para el administrador.

3.4 Fuentes de Información de Datos.

Para la recolección de información relevante, se emplearon dos categorías de fuentes:

3.4.1 Fuentes Primarias.

Ajayi (2023) describe las fuentes primarias como la evidencia original recolectada de primera mano, directamente de la realidad bajo estudio, sin haber sido sometida a interpretaciones previas de terceros.

En esta investigación, las fuentes primarias constituyeron el pilar del levantamiento de requerimientos del sistema. Se utilizaron entrevistas semi-estructuradas al administrador y técnicos del Vivero Lastenia, así como observación directa de los procesos de injertación y control de inventarios, con lo que se capturaron las necesidades exactas de los usuarios y las reglas de negocio reales de la producción de cacao.

3.4.2 Fuentes Secundarias.

Las fuentes secundarias son el conjunto de información ya recolectada, procesada y publicada por otros investigadores con propósitos distintos al estudio actual (Mohajan, 2018).

Los artículos científicos indexados para el desarrollo del sistema justificaron el uso de una arquitectura frontend dinámica y el patrón MVVM, y los manuales técnicos y la documentación oficial de los marcos de trabajo seleccionados, junto con los registros de

producción y los registros históricos del vivero, permitieron comprender las tendencias de rendimiento y pérdidas antes de automatizar los cronogramas en la plataforma web.

3.5 Estrategia operacional para la recolección de datos.

Majid (2018) destaca lo importante que es planificar con cuidado los datos recopilados para obtener información más fiable, y con el nivel de detalle que se necesita para un estudio.

Para este proyecto se hizo un plan de recopilación de datos, el cual refleja la lógica de negocio del vivero Lastenia, todo esto mediante el contacto directo con los responsables de la zona, los cuales brindaron datos importantes sobre el tiempo de ciclo para los lotes y los flujos de trabajo que se emplean sobre los mismos. Esto permitió tener información empírica para estructurar una base de datos, ajustar cronogramas automatizados a las condiciones biológicas presentes en las plántulas.

3.5.1 Población

Majid (2018) sostiene que delimitar la población es el primer paso crítico para garantizar que los datos recolectados tengan validez externa, pues constituye el conjunto de todos los individuos que podrían verse afectados o beneficiados por los resultados de la investigación.

Para el desarrollo de esta investigación se seleccionó exclusivamente a los 19 profesores de las carreras de agropecuaria y agronegocios, áreas de conocimiento que intervienen de forma directa en las labores del vivero. Esta segmentación de 19 profesores está conformada por agrónomos expertos en el área y planificación del vivero Lastenia. A partir de este grupo se planificó la recolección de información mediante encuestas y entrevistas, garantizando que el sustento del sistema web tuviera una base académica y operativa vinculada directamente con el entorno institucional donde se origina el proyecto.

3.5.2 Segmentación.

Dividir la población en subgrupos homogéneos para aplicar técnicas de investigación diferenciadas, según criterios como la experiencia técnica o la capacidad de toma de decisiones, es lo que se conoce como segmentación (Kothari, 2004).

Una vez definida la población general, se procedió a realizar una segmentación estratégica para filtrar a los sujetos que guardan una relación técnica con el objeto de estudio. Para el caso específico de este proyecto, se estableció la segmentación por conveniencia intencionada, incluyendo a los 19 profesores señalados en la población, incluyendo al autor más importante dentro del vivero, el ingeniero Pedro Nivelá. Se decidió segmentar con este mismo grupo porque posee el conocimiento técnico necesario para validar la problemática y definir los requerimientos de la arquitectura dinámica. Este proceso de segmentación fue crucial para asegurar que la información recolectada proviniera de personal con experiencia en la producción de cacao, descartando perfiles ajenos a la dinámica agrícola y centrando el análisis en quienes interactúan con los procesos de planificación y control que el software busca automatizar.

3.5.3 Tamaño de la Muestra.

Para Majid (2018), el tamaño de la muestra determina la parte representativa de la población segmentada que será efectivamente estudiada, y para contextos de investigación pequeños, tal como el de este proyecto, es recomendable el muestreo no probabilístico, el cual pone como prioridad la relevancia técnica de los participantes por encima de la representatividad estadística, siempre y cuando los criterios de selección estén bien justificados.

Para este proyecto no aplica la muestra, ya que se definió ya una cantidad reducida de 19 profesores tanto en la población como en la segmentación. Trabajando con dicha cantidad se aseguran datos más exactos y afines al proyecto, datos provenientes de los autores directos que ejercen actividades en el vivero.

3.5.4 Análisis de las herramientas de recolección de datos a utilizar.

3.5.4.1 Entrevista Semi-estructurada.

Alase (2017) presenta la entrevista semi-estructurada como un método de recolección de datos flexible, que permite profundizar en las experiencias y necesidades de los participantes mediante un diálogo guiado pero abierto.

Para la obtención de los requerimientos del sistema se aplicó una entrevista semi-estructurada únicamente al Ing. Pedro Nivelá, coordinador del Vivero Lastenia. Mediante este proceso se identificaron los puntos críticos en la planificación de la producción de cacao, como la dificultad para predecir las fechas de injertación y la falta de un registro centralizado de mermas, y con esa información se definieron las funcionalidades prioritarias de la arquitectura dinámica del frontend.

3.5.4.2 Encuesta.

La encuesta es una técnica que utiliza cuestionarios estructurados para obtener mediciones cuantitativas sobre opiniones, actitudes o características de un grupo específico (Rahi, 2017) .

En esta investigación, se realizaron encuestas a 18 de los profesores segmentados. Los datos recolectados permitieron cuantificar que la falta de alertas automáticas y la dificultad de reprogramar tareas se consideran problemas críticos, lo que sirvió como sustento estadístico para priorizar el desarrollo del módulo de organización y planificación dentro del sistema web.

3.5.4.3 Estructura de lo(s) instrumento(s) de recolección de datos aplicados.

Según lo recabado, se diseñaron dos instrumentos de recolección de datos, los cuales se complementan para una comprensión completa de las necesidades que presenta el vivero Lastenia, ya que cada uno aborda problemáticas que van desde perspectivas técnicas a perspectivas operativas.

A. Guía de Entrevista Semi-estructurada

El primer eje, lógica de planificación, correspondiente a las preguntas P1 a P5, busca identificar los hitos obligatorios del ciclo de producción de cacao, los plazos técnicos entre cada fase y la forma en que se gestiona la reprogramación ante condiciones climáticas imprevistas. Un segundo bloque donde la temática es de gestión operativa y monitorización (P6 y P7), corresponde a la asignación de tareas, las prácticas que se usan para monitorización y los criterios que se consideran como prioridad, estos definirán el flujo de trabajo que el sistema debe seguir. Para un tercer bloque, se busca definir detalles como requisitos de interfaz

y experiencia de usuario (P8 a p10), con dicha información se busca establecer lo que los usuarios necesitan visualizar de inmediato y condiciones reales de uso, como la utilidad del sistema baso luz solar, mediante esto se justifica el panel de control interactivo y una arquitectura de interfaz amigable y acorde al uso real del vivero.

B. Formato de Encuesta Estructurada

Se utilizó un formato de encuesta estructurada como instrumento de medición cuantitativa de 14 ítems, dirigido a un grupo selecto de 18 expertos y profesores del sector agrícola, con el objetivo de medir las dificultades operativas, los riesgos y las necesidades de adopción tecnológica en relación con la gestión de viveros, dado que el instrumento se compone de diferentes tipos de preguntas cerradas. Se emplearon preguntas con escala Likert para medir los niveles de precisión, dificultad y complejidad, mientras que las preguntas dicotómicas se utilizaron para validar la necesidad de automatización. Se aplicaron rangos numéricos para recopilar estimaciones de tiempos, porcentajes y frecuencias operativas. Por lo tanto, la encuesta se estructuró en torno a tres dimensiones críticas: eficiencia operativa, riesgo y viabilidad, y adopción tecnológica.

Dimensiones Críticas:

La primera dimensión, eficiencia operativa, evalúa el tiempo dedicado a cálculos manuales, que supera los 60 minutos, y la frecuencia de retrasos en las actividades de planificación y seguimiento, sirviendo así como base para evaluar la optimización que podría proporcionar el sistema propuesto. La segunda dimensión, riesgo y viabilidad, evalúa la probabilidad de error humano, retrasos en el seguimiento y pérdida de plántulas, que se estima entre el 6 % y más del 10 % cuando no se dispone de un seguimiento oportuno, y determina los riesgos que el sistema debe reducir mediante la automatización y un mejor control de la información. La tercera dimensión, la adopción tecnológica, evalúa la necesidad percibida de una herramienta digital moderna y la disposición de los usuarios a adoptar interfaces dinámicas y receptivas; en consecuencia, esta información respalda la relevancia de implementar un sistema web adaptado a las condiciones operativas del vivero. Validación de Arquitectura (Ítem 8 de la versión final): Identifica temores técnicos como la falta de conectividad, lo que sustenta técnicamente la incorporación de un Nodo Móvil nativo (Capacitor + SQLite) con capacidades de sincronización offline dentro de la arquitectura general del ERP Lastenia.

3.5.5 Plan de recolección de datos

Tabla 1: Plan de recolección de datos

Fecha	Actividad	Personal	Tipo de Instrumento
28/01/2026	Aplicar Encuesta a licenciados-ingenieros que tienen relación con el Vivero Lastenia	Profesores	Encuesta
28/01/2026	Aplicar Entrevista a dirigentes y encargados del Vivero Lastenia	Dirigente vivero Lastenia; Ing. Pedro Nivela	Entrevista

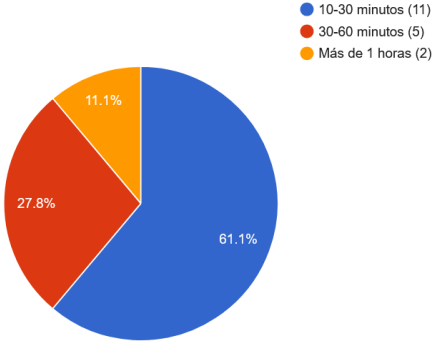
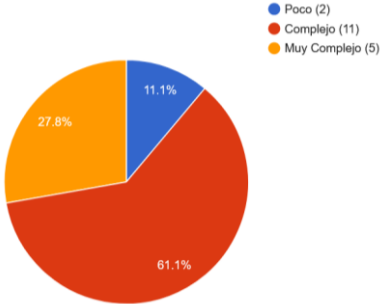
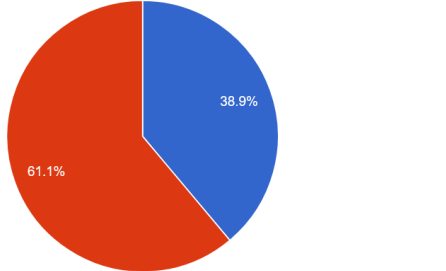
3.6 Análisis y presentación de resultados

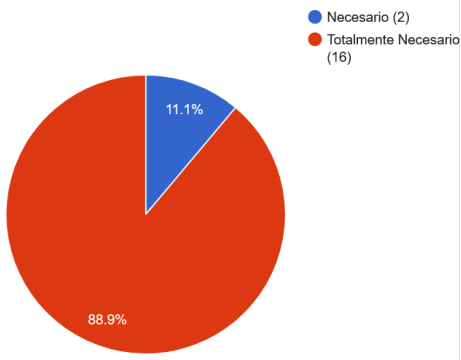
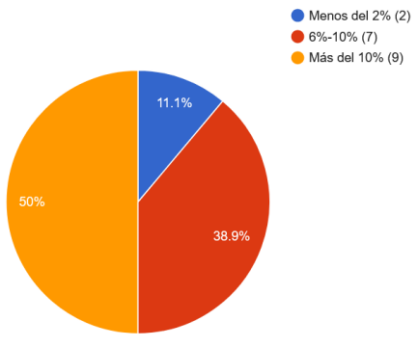
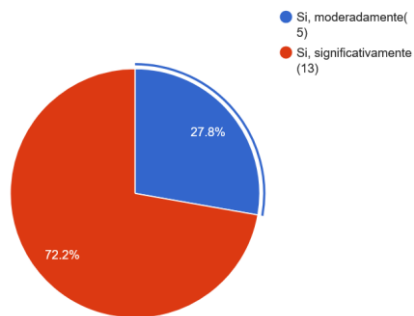
En esta sección se detallan los hallazgos obtenidos mediante la aplicación de los instrumentos de investigación a los docentes e ingenieros vinculados al Vivero Lastenia. El análisis se centra en validar la necesidad de una herramienta digital que sustituya la planificación manual por un sistema de interfaces dinámicas capaz de gestionar cronogramas en tiempo real

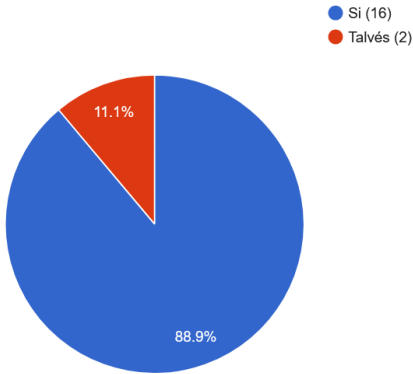
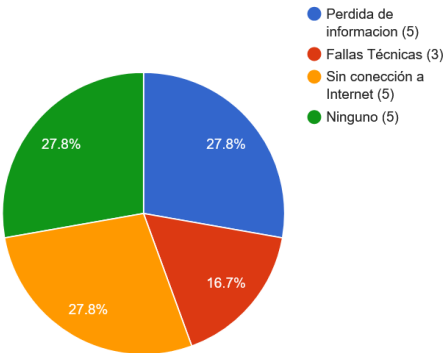
3.6.1 Tabulación y análisis de los datos

Tabulación y resultados de la Encuesta:

Tabla 2: Tabulación y análisis de los datos

Pregunta	Gráfico	Interpretación
Pregunta 1: ¿Cuánto tiempo le toma calcular manualmente las fechas de un ciclo productivo completo?	 ● 10-30 minutos (11) ● 30-60 minutos (5) ● Más de 1 hora (2)	Entre los docentes que ejercen actividades dentro del vivero, dedican entre 10 y 30 minutos al cálculo, y un grupo importante supera la hora. Esto confirma el cuello de botella existente que justifica automatizarlo mediante algoritmos en el frontend.
Pregunta 2: ¿Cómo califica la complejidad de reprogramar actividades ante imprevistos?	 ● Poco (2) ● Complejo (11) ● Muy Complejo (5)	En su mayoría esta acción se lo califica como complejo o muy complejo, esto reafirmando la necesidad de una arquitectura dinámica capaz de reajustar cronogramas con unas pocas interacciones intuitivas y dejar atrás la rigidez actual.
Pregunta 3: ¿Con qué frecuencia detecta actividades fuera de fecha óptima por errores de cálculo?	 ● 0-1 vez al mes (7) ● 2-4 veces al mes (11)	Para mas de un 50% de docentes es habitual tener errores en base a la desincronización de calendarios, esta interpretación respalda que se necesita una lógica de planificación y de precisión para calcular fases y fechas de las mismas

Pregunta	Gráfico	Interpretación									
Pregunta 4: ¿Considera necesario automatizar el cálculo de fechas y cronogramas?	 ● Necesario (2) ● Totalmente Necesario (16) <table><thead><tr><th>Respuesta</th><th>Porcentaje</th><th>Cantidad</th></tr></thead><tbody><tr><td>Totalmente Necesario</td><td>88.9%</td><td>16</td></tr><tr><td>Necesario</td><td>11.1%</td><td>2</td></tr></tbody></table>	Respuesta	Porcentaje	Cantidad	Totalmente Necesario	88.9%	16	Necesario	11.1%	2	En su gran mayoría los docentes coinciden que la solución de software es relevante, tiene valor y debido a los problemas encontrados se vuelve una necesidad, justificando así que el módulo de planificación automatizada es de prioridad para los usuarios
Respuesta	Porcentaje	Cantidad									
Totalmente Necesario	88.9%	16									
Necesario	11.1%	2									
Pregunta 5: ¿Qué porcentaje de plántulas pierde por falta de seguimiento oportuno del cronograma?	 ● Menos del 2% (2) ● 6%-10% (7) ● Más del 10% (9) <table><thead><tr><th>Porcentaje</th><th>Cantidad</th></tr></thead><tbody><tr><td>Más del 10%</td><td>9</td></tr><tr><td>6%-10%</td><td>7</td></tr><tr><td>Menos del 2%</td><td>2</td></tr></tbody></table>	Porcentaje	Cantidad	Más del 10%	9	6%-10%	7	Menos del 2%	2	Las pérdidas por errores en la planificación manual son constantes. Un sistema con seguimiento en tiempo real puede reducir la mortalidad de plántulas al alertar sobre tareas críticas.	
Porcentaje	Cantidad										
Más del 10%	9										
6%-10%	7										
Menos del 2%	2										
Pregunta 6: ¿Cómo afecta la ausencia de registro histórico en sus decisiones?	 ● Si, moderadamente(5) ● Si, significativamente (13) <table><thead><tr><th>Respuesta</th><th>Porcentaje</th><th>Cantidad</th></tr></thead><tbody><tr><td>Si, significativamente</td><td>72.2%</td><td>13</td></tr><tr><td>Si, moderadamente</td><td>27.8%</td><td>5</td></tr></tbody></table>	Respuesta	Porcentaje	Cantidad	Si, significativamente	72.2%	13	Si, moderadamente	27.8%	5	Los encuestados señalan que esto afecta la toma de decisiones, lo que justifica un módulo que además de planificar, guarde el historial de ciclos anteriores para mejorar producciones futuras.
Respuesta	Porcentaje	Cantidad									
Si, significativamente	72.2%	13									
Si, moderadamente	27.8%	5									

Pregunta	Gráfico	Interpretación
Pregunta 7: ¿Participaría en una capacitación para usar el nuevo sistema?	 ● Si (16) ● Talvés (2)	Un 88.9% muestra alta disposición a usar el sistema, lo que confirma que su éxito depende de una interfaz intuitiva y refuerza el uso de arquitecturas frontend modernas y fáciles de aprender.
Pregunta 8: ¿Cuál es su principal temor ante la digitalización?	 ● Perdida de informacion (5) ● Fallas Técnicas (3) ● Sin conexión a Internet (5) ● Ninguno (5)	El 29.4% teme a la falta de internet en el campo. Este dato valida técnicamente mi decisión de utilizar tecnologías como un Nodo Móvil nativo (Capacitor + SQLite), que permiten la visualización de cronogramas incluso en condiciones de baja conectividad

Tabulación y resultado de la Entrevista:

Tabla 3: Tabulación y resultado de la Entrevista

Pregunta	Respuesta	Interpretación
Pregunta 1: ¿Cuáles son los hitos o etapas obligatorias desde la siembra hasta el despacho final de un lote de cacao?	Se confirman 5 fases técnicas que duran de 4 a 5 meses.	Estas fases deben ser los pilares del flujo de trabajo en el frontend, permitiendo al usuario visualizar el progreso de cada lote de forma secuencial.
Pregunta 2: ¿Cómo determinan actualmente las fechas para pasar de la etapa	Se requieren ~22 días para el injerto	Estos datos son las constantes matemáticas que mi sistema usará para calcular automáticamente las fechas

Pregunta	Respuesta	Interpretación
de germinación a la de injertación?	y ~50 para el despacho.	finales de producción desde el día de la siembra.
Pregunta 3: ¿Existen variaciones en estos tiempos según el tipo de semilla o técnica de injerto utilizada?	Se realiza planificación previa, pero es vulnerable a factores externos como la falta de agua, manejando un ciclo de 5 a 6 meses	Este resultado confirma que, aunque existe conocimiento técnico, el cálculo es manual y no considera variables dinámicas de infraestructura. Mi sistema resolverá esto permitiendo que el usuario ingrese restricciones en la interfaz dinámica para que el algoritmo reajuste automáticamente las fechas de transición entre fases, eliminando la incertidumbre del cálculo "al ojo".
Pregunta 4: Si ocurre un retraso (clima/insumos), ¿cómo reprograman las actividades y cómo afecta esto a las etapas posteriores?	El clima ha sido índice de retrasos al momento de efectuar actividades.	El sistema no puede ser rígido: el frontend tiene la necesidad de brindar ventanas de flexibilidad de 24 a 48 horas para reajustar el plan ante cambios esporádicos sin perder la trazabilidad.
Pregunta 5: ¿Qué tan útil sería que el sistema calcule automáticamente la fecha de despacho final basándose en el día de la siembra?	Aunque las plántulas no mueran, estas sí generan retrasos en la entrega y espacio ocupado innecesariamente en el vivero.	La ineficiencia se puede traducir como costos por mantener las plántulas sin despachar y espacio ocioso que se puede utilizar para un nuevo ciclo productivo. El módulo de planificación permitirá visualizar la ocupación de las platabandas en el tiempo, optimizando la rotación de lotes para producir más en el mismo espacio.
Pregunta 6: ¿Cómo asignan actualmente las tareas diarias a los trabajadores y cómo verifican que se cumplieron?	El entrevistado indica un proceso inadecuado de aplicaciones como Excel y explica que se usa por falta de otra alternativa.	Confirma que el proyecto si cubre un vacío tecnológico real, con una herramienta especializada que supera las limitaciones que surgen al utilizar opciones como Excel.

Pregunta	Respuesta	Interpretación
Pregunta 7: ¿Qué criterios utilizan para priorizar qué lote atender primero cuando hay recursos limitados?	Según la planificación se debe reformular siempre que los lotes necesiten más atención que otros.	Refuerza la necesidad de una arquitectura dinámica, la cual actualiza los datos al instante cada vez que se hace un ajuste.
Pregunta 8: ¿Qué información necesita ver "de un solo vistazo" (resumen) apenas abre el cronograma del día en la aplicación?	Necesita ver la actividad del día y el estado de humedad.	Define el diseño del dashboard principal: una interfaz reactiva que muestre las métricas críticas de forma visual y rápida.
Pregunta 9: ¿Con qué antelación necesitan recibir alertas sobre una actividad próxima (ej. 24h antes, 1 semana)?	Necesita alertas con al menos 24 horas de anticipación.	Aquí se define un requerimiento funcional crítico: que el sistema debe integrarse con un módulo de notificaciones que se active un día antes de cada hito productivo.
Pregunta 10: ¿En qué momentos y lugar físico revisaría el cronograma (bajo el sol en el vivero o en la oficina)?	Revisar las actividades y gestión dentro del campo, al alcance de un dispositivo móvil.	Justifica que se necesita diseño responsivo y móvil-primero, que ponga la planificación al alcance del trabajador y no solo se quede en la oficina.

3.6.2 Presentación y descripción de los resultados obtenidos

En cuanto a los resultados de la pregunta 4 de la encuesta, todos los encuestados coincidieron en que automatizar los cálculos de fechas y horarios sería una buena práctica para reducir la carga administrativa, lo cual también se confirma en la pregunta 6 de la entrevista, donde la persona a cargo afirma que automatizar la fecha de envío sería "absolutamente correcto" e indica que, actualmente, utilizan Excel y Microsoft Project porque no disponen de una herramienta especializada. Por lo tanto, ambos instrumentos confirman una demanda real

de automatización, lo que justifica el desarrollo de un módulo personalizado en lugar de seguir dependiendo de métodos manuales.

Respecto a los resultados de la pregunta 11 de la encuesta, se observa que el cálculo manual de un solo lote lleva entre 10 y 60 minutos, lo que constituye un claro cuello de botella operativo, y esta situación se explica en las preguntas 1 y 2 de la entrevista, ya que el ciclo de producción se organiza en cinco fases técnicas fijas: siembra, germinación, injerto, desarrollo y envío, con un período estándar aproximado de 72 días entre el injerto y el envío. Dado que estos tiempos siguen una lógica técnica estable, calcularlos manualmente se convierte en una tarea repetitiva e innecesaria, lo que justifica la implementación de algoritmos front-end capaces de generar estas fechas automáticamente.

En cuanto a los resultados de la pregunta 5 de la encuesta, se observa que la reprogramación de actividades tras eventos imprevistos se percibe como “compleja” o “muy compleja”, percepción que se confirma en las preguntas 4 y 10 de la entrevista, donde el responsable explica que la reprogramación es frecuente debido a las condiciones climáticas o la escasez de agua, y que el sistema debería permitir ventanas de flexibilidad de uno o dos días. Estos hallazgos justifican que se necesita una arquitectura frontend dinámica y reactiva que sea capaz de calcular y reformar horarios, reducir errores y también adaptarse a cambios suscitados en el cambiante entorno agrícola del vivero.

Analizando los resultados de la pregunta 8 de la encuesta se indica muestra que la falta de acceso a internet es la principal preocupación con respecto a la digitalización del flujo de trabajo dentro del vivero, esto coincide con la pregunta 8 de la entrevista la cual el Ing. Pedro Nivelá confirma que consultaría el horario principalmente desde un teléfono móvil mientras trabaja en el campo, por lo tanto, este resultado respalda el uso de tecnologías móviles nativas como Capacitor y SQLite según la arquitectura propuestas, ya que permitiría al los usuarios tener acceso al aplicativo incluso con conectividad limitada. Para las preguntas 9 y 13 de la encuesta se indica que la falta de alertas automáticas al no avisarse de procesos y actividades clave cuando es debido afectaría la calidad en el desarrollo y mantenimiento de las plántulas, esta falta de alerta puede generar que se incumplan con actividades que vital importancia y generaría errores de planificación, complementando a esta narrativa la pregunta 7 de la entrevista confirma que se debe solicitar alertas con al menos 24 horas de anticipación para coordinar suministros y mano de obra.

3.6.3 Informe final de análisis de datos

Tras el análisis de los datos recolectados, se determina que la gestión de planificación en el Vivero Lastenia presenta dificultades asociadas a la dependencia de métodos manuales y registros físicos. De acuerdo con la investigación, el cálculo manual de cronogramas constituye una de las principales dificultades operativas, debido a que la pregunta 11 de la encuesta evidencia que los técnicos pueden invertir hasta 60 minutos en calcular las fechas correspondientes a un solo lote. Este resultado permite identificar que el procedimiento manual requiere una cantidad considerable de tiempo para establecer los cronogramas de producción, constituyéndose en una condición que afecta la agilidad con la que se desarrolla esta actividad.

El acceso a información histórica y la reacción ante imprevistos también presentan dificultades dentro de la gestión de planificación. Tanto la pregunta 4 como la 5 de la entrevista coinciden en que la reprogramación de los calendarios debido a factores cambiantes o imprevistos dentro del vivero es una situación recurrente durante la operación y representa una dificultad para los coordinadores cuando se realiza manualmente. De igual manera, el uso de registros en papel y hojas de Excel limita la organización y consulta de la información relacionada con los cambios que se producen durante el proceso, generando una mayor carga administrativa y dificultando el monitoreo oportuno de las actividades. Estas condiciones pueden repercutir en el seguimiento del desarrollo de las plántulas al no disponer de información actualizada de manera ágil cuando se presentan modificaciones en los procesos.

Por otra parte, la pregunta 8 de la encuesta evidencia que la conectividad limitada disponible en el vivero constituye una condición que puede dificultar el acceso continuo a la información relacionada con las actividades de planificación y seguimiento. Esta situación adquiere mayor relevancia debido a que parte de las actividades se desarrollan directamente en el campo de acción del vivero, donde la disponibilidad de conexión puede variar. En consecuencia, la limitada conectividad representa una condición adicional que influye en la manera en que se consultan y gestionan los datos durante las actividades productivas.

CAPÍTULO IV:

4 MARCO PROPOSITIVO

4.1 Introducción

En base a la investigación y los datos recolectados, la solución propuesta es un sistema web construido en base a una arquitectura frontend monolítica y modular, el módulo de planificación estará dentro de un ERP, dicho modulo se complementará con un nodo nativo de Android desarrollado en React y empaquetado con Capacitor y respaldado por una base de datos SQLite local. Esta decisión de diseño responde a la inquietud de los docentes que efectúan actividades en el vivero respecto a la poca conectividad a internet mientras se trabaja en el campo. Para solucionar esto, el sistema ofrece una funcionalidad offline real, almacenando los datos localmente y sincronizándolos con el nodo central una vez restablecida la conexión, lo que permite al responsable gestionar la planificación de la producción desde su teléfono móvil incluso cuando la cobertura de red en el vivero es limitada o inestable.

4.2 Descripción de la propuesta

Se propone un Sistema Automatizado de Planificación y Gestión para el ciclo fenológico del cacao, con una interfaz reactiva. Este sistema concentra la lógica del ciclo del cacao. A diferencia de Excel, calcula automáticamente las cinco fases técnicas: siembra, germinación, injerto, crecimiento y envío, facilitando además la reprogramación de actividades ante problemas climáticos.

El concepto principal se basa en el uso de una interfaz dinámica con el patrón MVVM, que muestra las modificaciones de la programación en tiempo real. Por lo tanto, este sistema no se limita a almacenar datos, sino que busca transformar la gestión del vivero mediante la automatización de flujos de trabajo y la visualización de métricas críticas en tiempo real.

A) Módulo Web (Administración y Estrategia)

- **Gestión Dinámica de Cronogramas:** Calcula de forma automática las fechas de inicio y fin de un ciclo productivo, desde fases como siembra hasta despacho de las plántulas.

- **Dashboard Reactivo:** Muestra una inmediata y sencilla forma de entender los indicadores como capacidad y cronogramas de cada uno de los lotes
- **Panel de Reprogramación:** Modifica fechas dentro de los ciclos productivos de los lotes, teniendo la flexibilidad de cambios en la producción de las plántulas sin perder el registro del plan original.
- **Reportes Automatizados:** Genera estadísticas de producción, pérdidas y rendimiento para cada lote, lo que permite tomar decisiones estratégicas más precisas.

B) Nodo Móvil — Capacitor + SQLite (Operación en Campo)

- **Interfaz Mobile-First:** Diseño responsivo pensado para usarse en dispositivos móviles, uso que mayormente se dará bajo luz solar directa en el campo.
- **Sistema de Alertas Proactivas:** Incorpora un sistema de alertas proactivas que envía notificaciones 24 horas antes de cada actividad programada.
- **Modo Offline:** Dispone de un modo sin conexión que permite al usuario consultar los horarios y registrar el progreso incluso sin conexión a internet, sincronizando la información una vez restablecida la señal.
- **Monitoreo de Lotes:** visualización interactiva de la fase fenológica de cada planta por bancal, lo que permite conocer rápidamente el estado real del vivero en las operaciones diarias.

4.3 Determinación de recursos

4.3.1 Humanos:

Tabla 4: Recursos Humanos

Personal	Función
Tesista / Desarrollador	Diseño, desarrollo e implementación del sistema web, incluyendo la programación del frontend, backend y base de datos, así como la generación de diagramas y documentación técnica necesarios para el mantenimiento de la estructura y el funcionamiento del sistema
Director de tesis	Consultoría metodológica y técnica durante todo el proceso de desarrollo, incluyendo la revisión y validación del progreso en la fase de diseño propuesta, para asegurar que el sistema se ajuste a los objetivos establecidos.
Administrador del vivero	Validación de los requisitos funcionales está orientada a la planificación y gestión de la producción, así como a la definición de criterios de control y monitorización que permitan al sistema responder a las necesidades operativas reales del vivero.
Técnico o responsable operativo	Información directa del vivero sobre sus actividades actuales, plazos y procesos de producción, información esencial para la validación de registros, la comprobación de funcionalidades en campo y la verificación del correcto funcionamiento del sistema en condiciones operativas reales..

4.3.2 Tecnológicos:

Tabla 5: Recursos Tecnológicos de Hardware

Hardware	Características
PC	Windows 10 Pro RAM 8,00 GB

	Procesador Intel(R) Core(TM) i5-1035G1 CPU @ 1.00GHz 1.19 GHz SSD 256 GB de almacenamiento
Celular	RAM 4,00 GB ROM 128,00 GB CPU Octa-core Max 2.00GHz Android 12 / Google Chrome

Tabla 6: Recursos tecnológicos de Software

Software	Características
Entorno de desarrollo	Visual Studio Code (HTML5, TypeScript, CSS, PHP)
Servidor local	XAMPP / servidor local Laravel (Apache, PHP 8.2, MySQL)
Control de versiones	Git / GitHub
Frameworks y librerías	React 19 con TypeScript (librería JavaScript), Vite (empaquetador), Tailwind CSS (librería para diseño web), Laravel 12 (framework backend PHP), Recharts (visualización de datos mediante gráficos)
Navegadores	Google Chrome, Mozilla Firefox (pruebas de diseño responsivo)
Sistema Operativo	Windows 11

4.3.3 Económicos:

Tabla 7: Recursos Económicos

Cantidad	Concepto	Características	C/U	Subtotal
1	Laptop principal	Equipo para desarrollo del sistema	\$750,00	\$750,00
1	Celulares	Pruebas responsive y funcionamiento móvil	\$180,00	\$180,00
600	Horas de desarrollo	Tiempo estimado para análisis, desarrollo, pruebas y documentación	\$5,00	\$3.000,00
1 año	Hosting web	Servicio para despliegue del prototipo	\$60,00	\$60,00
1 año	Internet	Conectividad para desarrollo y pruebas	\$25,00	\$300,00
	Total			\$4.290,00

4.4 Desarrollo según metodología Scrum

El desarrollo del sistema se rige bajo un enfoque de metodologías ágiles (Scrum), permitiendo entregas funcionales en ciclos iterativos (sprints) de dos semanas. Este proceso asegura que la arquitectura frontend sea reactiva y se ajuste a las necesidades biológicas y climáticas del vivero detectadas en el diagnóstico.

4.4.1 Descripción del Producto

Se trata de un sistema web con arquitectura frontend dinámica, cuyo objetivo es automatizar la planificación y gestión de la producción en el vivero Lastenia. Desde una perspectiva técnica, es una aplicación web reactiva con un enfoque monolítico modular y un

nodo móvil nativo para Android desarrollado con Capacitor y SQLite, que ofrece una experiencia de usuario fluida, reactiva y adaptable tanto en entornos de escritorio como en dispositivos móviles utilizados directamente en el campo. El sistema es un motor de planificación centralizado que integra datos biológicos, climáticos y operativos, transformando la gestión empírica actual del vivero en un proceso estructurado y basado en datos.

4.4.1.1 Propósito del Producto

El propósito de este sistema es llenar el vacío entre la planificación manual y la ejecución diaria en el vivero, y de esta manera, automatiza el cálculo de los ciclos de producción de semillas de cacao, reduce los tiempos de planificación que actualmente superan los 60 minutos por lote, y también reduce los errores humanos que ocurren cuando las actividades no se siguen de manera oportuna.

4.4.1.2 Funcionalidades Clave

- **Registro y gestión de Viveros:** Crear vivero como un Workspace donde todas las funcionalidades accesibles (módulos) del sistema trabajan únicamente para dicho vivero creado.
- **Gestión de Metas de Producción:** Crear y modificar Metas de producción que den un objetivo específico a seguir, meta que estará ligada a la cantidad real de despacho y que refleje la cantidad real de plantulas esperadas y plántulas despachadas.
- **Registro Inteligente de Lotes:** Crear lotes que calculen la cantidad de su capacidad mediante operaciones geométricas basadas en alto y ancho de lote, en el diámetro de ocuparían las plántulas, y la cantidad y ancho de los pasillos/corredores entre filas de plántulas.
- **Mapa Interactivo de Lotes:** Panel visual interactivo donde se puede arrastrar y soltar la figura generada de cada lote en función a las dimensiones físicas reales, para que de una representación del campo más precisa e intuitiva.

- **Algoritmo de Planificación Automática:** Calcula al instante las 5 fases técnicas de un ciclo productivos de un lote, en cual según la fecha de inicio del lote calcula las fechas por las que pasaría cada fase hasta su entrega final.
- **Reprogramación Dinámica:** Interfaz reactiva donde se pueden ajustar los cronogramas antes generados para cubrir eventualidades cambiantes dentro del entorno real del vivero y que no se pierda el hilo de la planificación y gestión del vivero.
- **Resincronización Automática de Cronogramas:** Actualiza el cronograma de todos los lotes que están en uso al momento de hacer alguna modificación en el tiempo de duración de alguna fase.
- **Gestión de Fases de Producción:** Permite a cada vivero tener la opción de configurar datos como el nombre, la fecha de inicio de su ciclo productivo y visualizar el tiempo estimado de duración de cada fase con fechas calculadas.
- **Dashboard de Control y Alertas:** Una visión general de las actividades por hacer con su respectiva fecha, listado de alertas tipo recordatorio para la ejecución de dichas actividades, todo enlazado al cronograma estipulado de los procesos productivos de cada lote
- **Calendario de Lotes:** Presentación visual de las fases planificadas de cada lote, todo compacto en un calendario que permite el acceso directo a una reprogramación de fechas en caso de ser necesario.
- **Operatividad Offline (Nodo Móvil):** Permite consultar y registrar datos de forma offline, que luego serán sincronizados con la base de datos online para uso efectivo en campo con limitaciones de conexión a internet.
- **Seguimiento de Despachos Pendientes:** Reflejo real de las plántulas despachadas (en el módulo de seguimiento) respecto a las plántulas que el sistema ha planificado que se despacharan, despacho el cual se vería reflejado en el indicador de progreso de la Meta de producción.

4.4.1.3 Usuarios Objetivo

Tabla 8: Usuarios Objetivo

Tipo de usuario	Principales funcionalidades
Administrador del Vivero	Definir cuál será la meta de producción, gestionar viveros y lotes dentro de los viveros, supervisar la trazabilidad completa de los ciclos productivos de cada lote y consultar un historial referente a los despachos esperados y reales.
Personal Técnico y Docentes	Validar requerimientos técnicos, iniciar y reprogramar ciclos de producción, configurar las fases del cronograma y coordinar las actividades académicas vinculadas a la producción.
Personal Operativo y Estudiantes	Consultar el estado de los lotes y el cronograma vigente, registrar avances en campo desde dispositivos móviles y recibir alertas sobre las actividades próximas.

4.4.1.4 Condiciones de Éxito del Producto

El éxito del producto se medirá bajo los siguientes criterios de aceptación técnica y operativa, validados en el marco de trabajo Scrum:

- **Reducción del tiempo operativo:** El sistema debe disminuir el tiempo de registro y cálculo de planificación de los 30-60 minutos actuales a menos de 5 minutos.
- **Precisión algorítmica:** El software debe garantizar un 0% de error de cálculo en las fechas de los hitos técnicos una vez ingresada la fecha de inicio del ciclo.
- **Disponibilidad en campo:** El sistema debe ser funcional bajo luz solar intensa y sin conexión a internet en al menos el 90% del área física del vivero mediante el Nodo Móvil nativo (Capacitor + SQLite).
- **Usabilidad y adopción:** Lograr una satisfacción del usuario que permita la capacitación efectiva en sesiones de máximo 30 minutos, dada la alta disposición (88.2%) del personal actual.
- **Trazabilidad de reprogramaciones:** Toda reprogramación de fase debe quedar registrada con su motivo, conservando el historial de cambios sobre el cronograma original.

4.4.2 Historias de Usuario

Siguiendo la metodología Scrum, se han definido seis historias de usuario que capturan los requisitos críticos para el "motor de planificación" del sistema, permitiendo transformar las metas de producción en cronogramas ejecutables

4.4.2.1 Historia de Usuario 1: Registrar Meta de Producción

Tabla 9: Historia de usuario 1

HISTORIA DE USUARIO			
Id de historia	HU01	Título:	Registrar Meta de Producción
Prioridad:	Alta	Usuario:	Administrador o Docente planificador
Riesgo de desarrollo:	Medio	Tamaño de la tarea:	Mediana
Como:	Administrador o Docente planificador.		
Quiero:	Definir Meta de producción (cantidad de plántulas esperadas)		
Para:	Que el sistema pueda validar cual es la capacidad de los lotes disponibles y organice segun los ciclos necesarios para cumplir la meta.		
Criterios de aceptación:	- Validación automática de capacidad física en cada lote (ej. ~5,000 fundas por lote). - Calcular la viabilidad de la meta en función del espacio y de los recursos disponibles. - Asignar un identificador único al plan de producción del vivero.		
Dependencias:	- Vivero registrado en el sistema. - Usuario con rol de Administrador o Docente.		

4.4.2.2 Historia de Usuario 2: Calcular Cronograma Fenológico Automático

Tabla 10: Historia de usuario 2

HISTORIA DE USUARIO			
Id de historia	HU02	Título:	Calcular Cronograma Fenológico Automático
Prioridad:	Alta	Usuario:	Docente planificador
Riesgo de desarrollo:	Alto	Tamaño de la tarea:	Grande
Como:	Docente planificador.		

Quiero:	Que el sistema genere forma automática el cronograma de 5 fases por lote basándose en la fecha de inicio del ciclo
Para:	Eliminar el cálculo manual que tarde en planificar las fases de los lotes.
Criterios de aceptación:	- Cálculo de fechas en cronograma para todas las fases. - Precisión de fecha de despacho final. - Visualización interactiva de las fases de cada lote en un tablero de control.
Dependencias:	- Meta de producción activa para el vivero. - Fases de producción configuradas (con su duración estimada).

4.4.2.3 Historia de Usuario 3: Reprogramación Dinámica por Variación Climática

Tabla 11: Historia de usuario 3

HISTORIA DE USUARIO			
Id de historia	HU03	Título:	Reprogramación Dinámica de Fases
Prioridad:	Alta	Usuario:	Docente planificador o Técnico
Riesgo de desarrollo:	Medio	Tamaño de la tarea:	Mediana
Como:	Docente planificador o Técnico.		
Quiero:	Ajustar manualmente alguna fecha que tenga que ser reprogramada debido a alguna eventualidad dentro del vivero.		
Para:	Que el sistema pueda volver a recalcular fechas posteriores el cambio y volver a reflejar una fecha estimada realista de fases productivas del lote.		
Criterios de aceptación:	- Elegir de forma manual una nueva fecha para una fase que será aplicada en un lote. - Recálculo automático de todo el cronograma del lote, cambios aplicados a las fechas posteriores a la fecha estipulada en el cambio realizado. - Registro del motivo por el que se hace la reprogramación y el historial de cambios sobre el cronograma original.		
Dependencias:	- Ciclo de lote en curso con una fase actual determinada por el sistema.		

4.4.2.4 Historia de Usuario 4: Monitoreo de Lotes y Capacidad de Campo

Tabla 12: Historia de usuario 4

HISTORIA DE USUARIO			
Id de historia	HU04	Título:	Monitoreo de Lotes y Capacidad de Campo
Prioridad:	Media	Usuario:	Técnico de campo
Riesgo de desarrollo:	Bajo	Tamaño de la tarea:	Mediana
Como:	Técnico de campo.		
Quiero:	Visualizar el estado de cada lote "de un solo vistazo" en un mapa interactivo		
Para:	asegurar el cumplimiento de los tiempos técnicos de cada fase.		
Criterios de aceptación:	- Tablero visual tipo mapa con la posición real de cada lote dentro del vivero. - Acceso responsivo desde dispositivos móviles para uso directo en el vivero. - Actualización reactiva de los datos de los lotes (patrón MVVM).		
Dependencias:	- Lotes registrados con su geometría y capacidad calculada.		

4.4.2.5 Historia de Usuario 5: Recepción de Alertas Proactivas en Campo

Tabla 13: Historia de usuario 5

HISTORIA DE USUARIO			
Id de historia	HU05	Título:	Recepción de Alertas Proactivas en Campo
Prioridad:	Media	Usuario:	Técnico de campo
Riesgo de desarrollo:	Alto	Tamaño de la tarea:	Grande
Como:	Técnico de campo.		
Quiero:	Recibir notificaciones con 24 horas de antelación sobre las actividades programadas		
Para:	coordinar insumos y mano de obra oportunamente.		
Criterios de aceptación:	- Notificación push enviada automáticamente 24 horas antes de cada hito.		

	- Funcionalidad offline para recibir y consultar alertas sin conexión a internet, mediante el Nodo Móvil (Capacitor + SQLite). - Sincronización de alertas al recuperar la señal de datos.
Dependencias:	- Nodo Móvil (Capacitor + SQLite) habilitado. - Sistema de notificaciones push configurado.

4.4.2.6 Historia de Usuario 6: Consulta de Memoria Histórica de Producción

Tabla 14: Historia de usuario 6

HISTORIA DE USUARIO			
Id de historia	HU06	Título:	Consulta de Memoria Histórica de Producción
Prioridad:	Baja	Usuario:	Administrador del Vivero
Riesgo de desarrollo:	Bajo	Tamaño de la tarea:	Pequeña
Como:	Administrador del Vivero.		
Quiero:	Acceder un registro histórico de ciclos de producción que se han dado en los diferentes lotes		
Para:	analizar tendencias que serán tomadas en cuenta en decisiones futuras.		
Criterios de aceptación:	- Almacenamiento estructurado de los datos dejados por ciclos terminados. - Filtros de búsqueda que ayuda a visualizar resultados según fecha, lote y meta de producción. - Generación de reportes comparativos donde visualizar la eficiencia operativa.		
Dependencias:	- Ciclos de producción cerrados en fase de despacho conectado al módulo de seguimiento.		

4.4.3 Diseño del Sistema / Descripción Técnica

Esta sección detalla la lógica de funcionamiento del módulo, enfocándose en la integración de los procesos biológicos del cacao con la arquitectura dinámica del software.

4.4.3.1 Caso de uso: Gestión de Metas y Cronograma de Producción

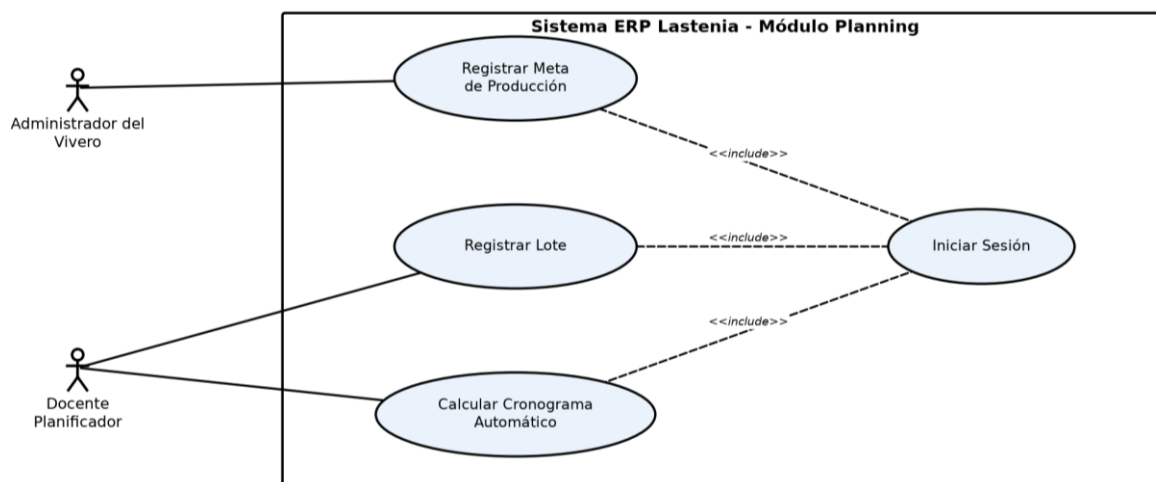


Ilustración 2: Caso de Uso - Gestión de Metas y Cronograma de Producción

Ilustración: Caso de uso: Gestión de Metas y Cronograma de Producción

4.4.3.1.1 Caso de uso: Gestión de Metas y Cronograma de Producción

Tabla 15: Caso de uso - Gestion de Metas y Cronograma de Producción

Documentación del caso de uso: Gestión de Metas y Cronograma de Producción	
Caso de uso N.º 001	Nombre del caso de uso: Gestión de Metas y Cronograma de Producción
Fecha: 08/07/2026	Elaborado por: Heyder Javier Verduga Calderón
Actores:	Administrador del Vivero Docente Planificador
Objetivo:	Permitir registrar la meta de producción y los lotes de un vivero, y calcular automáticamente el cronograma de las 5 fases del ciclo productivo a partir de la fecha de inicio.
Precondiciones:	Iniciar sesión. Vivero registrado en el sistema.
Postcondiciones:	Meta de producción, lote y cronograma de fases registrados en la base de datos.
Medios para realizar la acción:	Aplicación web (escritorio y dispositivo móvil).
Pasos	
1. Iniciar sesión. 2. Acceder al módulo de Viveros. 3. Registrar la meta de producción del vivero. 4. Registrar un lote (o seleccionar uno existente). 5. Seleccionar la fecha de inicio del ciclo. 6. El sistema calcula automáticamente las 5 fases del cronograma. 7. Visualizar el cronograma generado en el tablero de control.	

Documentación del caso de uso: Gestión de Metas y Cronograma de Producción

Situaciones excepcionales:

1. Sin conexión a internet durante el registro (modo offline del Nodo Móvil).
2. El vivero ya tiene una meta de producción activa.
3. El vivero no tiene fases de producción configuradas.

Revisado por: Ing. Mendoza Villamar Rocío Alexandra

4.4.3.2 Caso de uso: Monitoreo de Lotes

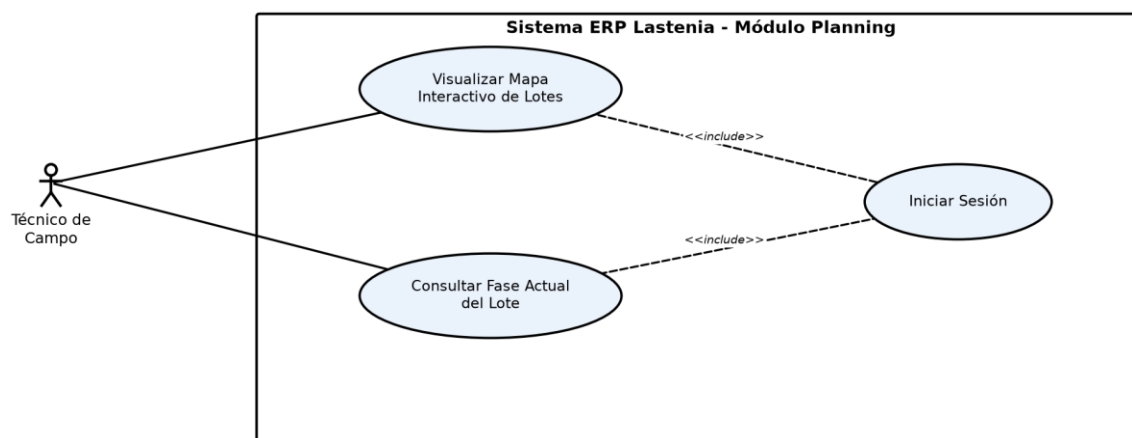


Ilustración 3: Caso de uso - Monitoreo de Lotes

4.4.3.2.1 Caso de uso: Monitoreo de Lotes

Tabla 16: Caso de uso - Monitoreo de Lotes

Documentación del caso de uso: Monitoreo de Lotes	
Caso de uso N.º 002	Nombre del caso de uso: Monitoreo de Lotes
Fecha: 08/07/2026	Elaborado por: Heyder Javier Verduga Calderón
Actores:	Técnico de Campo
Objetivo:	Permitir al técnico visualizar el estado y la fase actual de cada lote.
Precondiciones:	Iniciar sesión. Lotes registrados con posición en el mapa del vivero.
Postcondiciones:	Estado y fase actual de los lotes mostrados en pantalla.
Medios para realizar la acción:	Aplicación web / Nodo Móvil nativo, uso preferente en dispositivo móvil.
Pasos	
1. Iniciar sesión. 2. Acceder al módulo de Lotes. 3. Visualizar el mapa interactivo del vivero. 4. Seleccionar un lote del mapa. 5. Consultar la fase actual y la capacidad del lote.	
Situaciones excepcionales: 1. Sin conexión a internet en el área de campo (modo offline del Nodo Móvil). 2. El lote consultado no tiene un ciclo de producción activo.	
Revisado por: Ing. Mendoza Villamar Rocío Alexandra	

4.4.3.3 Caso de uso: Reprogramación de Cronograma

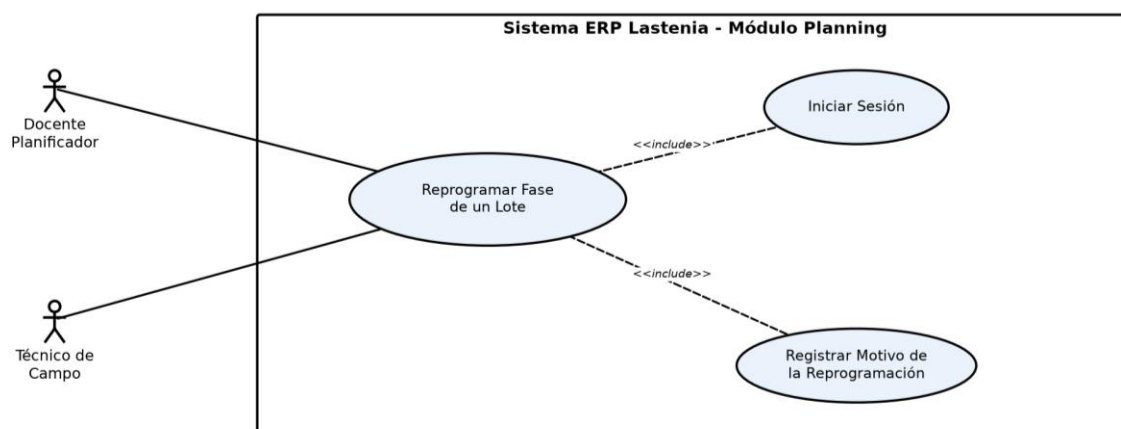


Ilustración 4: Caso de uso - Reprogramación de Cronograma

4.4.3.3.1 Caso de uso: Reprogramación de Fases

Tabla 17: Caso de uso - Reprogramación de Cronograma

Documentación del caso de uso: Reprogramación de Fases	
Caso de uso N.º 003	Nombre del caso de uso: Reprogramación de Actividades
Fecha: 08/07/2026	Elaborado por: Heyder Javier Verduga Calderón
Actores:	Docente Planificador Técnico de Campo
Objetivo:	Permitir ajustar manualmente la fecha de la fase actual de un lote ante un imprevisto operativo, recalculando en cascada el resto del cronograma.
Precondiciones:	Iniciar sesión. Lote con un ciclo de producción en curso.
Postcondiciones:	Cronograma recalculado en cascada y motivo de la reprogramación registrado en el historial.
Medios para realizar la acción:	Aplicación web (escritorio y dispositivo móvil).
Pasos	
1. Iniciar sesión. 2. Acceder al calendario de lotes. 3. Seleccionar el lote y la fase a reprogramar. 4. Ingresar la nueva fecha para la fase. 5. Registrar el motivo de la reprogramación. 6. El sistema recalcula en cascada las fechas de las fases posteriores. 7. Confirmar los cambios en el cronograma.	
Situaciones excepcionales:	
1. La fase seleccionada ya fue completada y no admite reprogramación. 2. La nueva fecha ingresada es anterior a la fecha actual de la fase.	
Revisado por: Ing. Mendoza Villamar Rocío Alexandra	

4.4.3.4 Caso de uso: Gestión de Alertas y Despachos

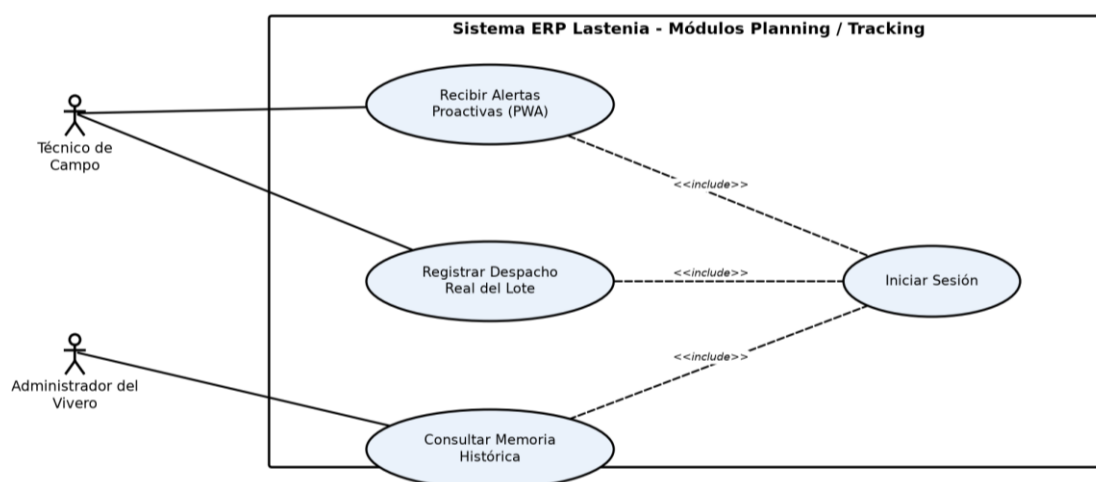


Ilustración 5: Caso de uso - Gestión de Alertas y Despachos

4.4.3.4.1 Caso de uso: Gestión de Alertas y Despachos

Tabla 18: Caso de uso - Gestión de Alertas y Despachos

Documentación del caso de uso: Gestión de Alertas y Despachos	
Caso de uso N.º 004	Nombre del caso de uso: Gestión de Alertas y Despachos
Fecha: 08/07/2026	Elaborado por: Heyder Javier Verduga Calderón
Actores:	Técnico de Campo Administrador del Vivero
Objetivo:	Permitir al técnico y al administrador del sistema recibir alertas sobre próximas actividades por realizar y registrar la cantidad real de despacho de plántulas al finalizar un ciclo, para así ir actualizando el avance de la meta de producción.
Precondiciones:	Iniciar sesión. Cerrar ciclo de lote, estado pendiente de despacho.
Postcondiciones:	Despacho registrado y meta de producción actualizada. Alerta entregada al usuario en su dispositivo.
Medios para realizar la acción:	Aplicación web / Nodo Móvil nativo, notificaciones push.
Pasos	
1. Iniciar sesión. 2. Consultar los despachos pendientes del vivero. 3. Ingresar la cantidad real de plántulas despachadas. 4. Confirmar el registro del despacho. 5. El sistema actualiza el avance de la meta de producción. 6. Recibir alerta proactiva 24 horas antes de la próxima actividad programada.	
Situaciones excepcionales:	
1. La cantidad despachada está fuera del rango permitido (1 a la capacidad del lote). 2. El ciclo ya tiene un despacho reportado previamente.	

3. Sin conexión a internet (modo offline del Nodo Móvil).

Revisado por: Ing. Mendoza Villamar Rocío Alexandra

4.4.3.5 Diagramas de Secuencia

4.4.3.5.1 Diagrama de secuencia: Cálculo de Cronograma Automático

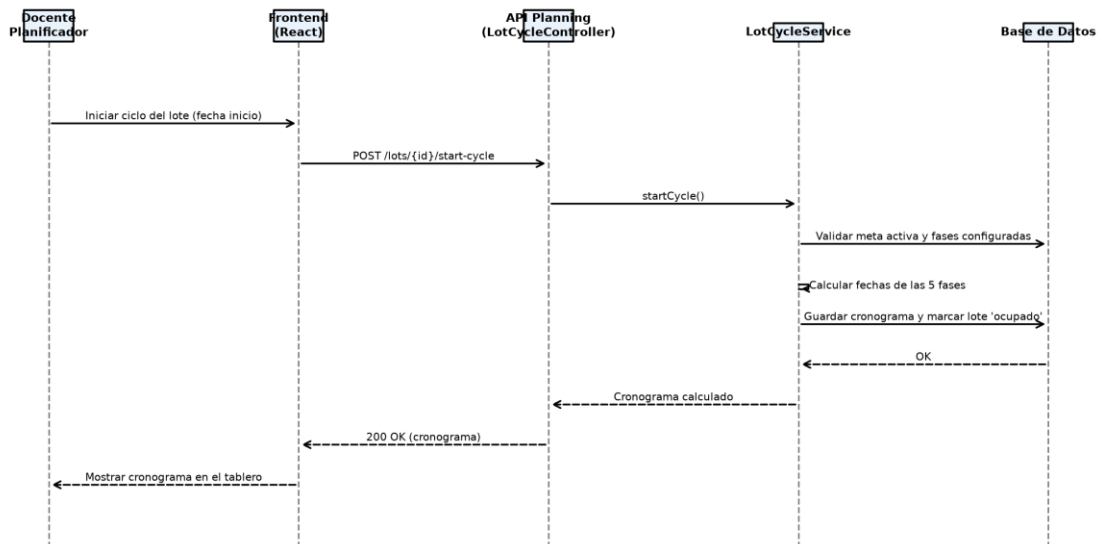


Ilustración 6: Diagrama de secuencia - Calculo de Cronograma Automático

4.4.3.5.2 Diagrama de secuencia: Monitoreo de Lotes

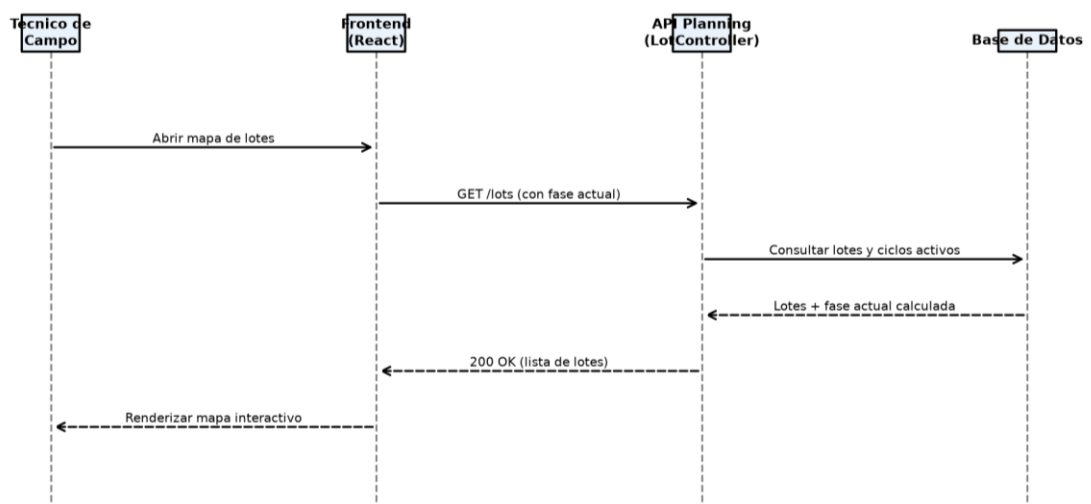


Ilustración 7: Diagrama de secuencia - Monitoreo de Lotes

4.4.3.5.3 Diagrama de secuencia: Reprogramación de Cronograma

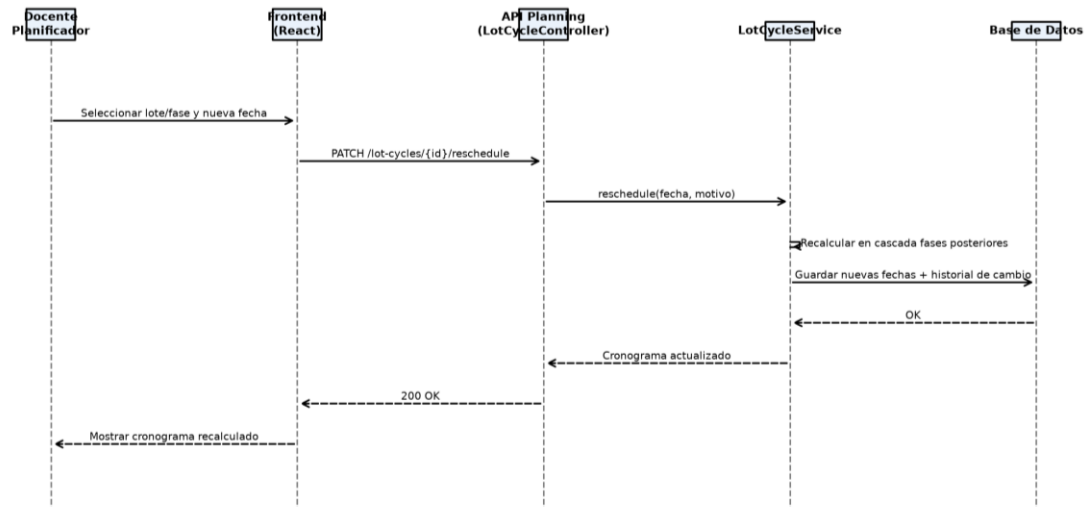


Ilustración 8: Diagrama de secuencia - Reprogramación de Cronograma

4.4.3.5.4 Diagrama de secuencia: Registro de Despacho

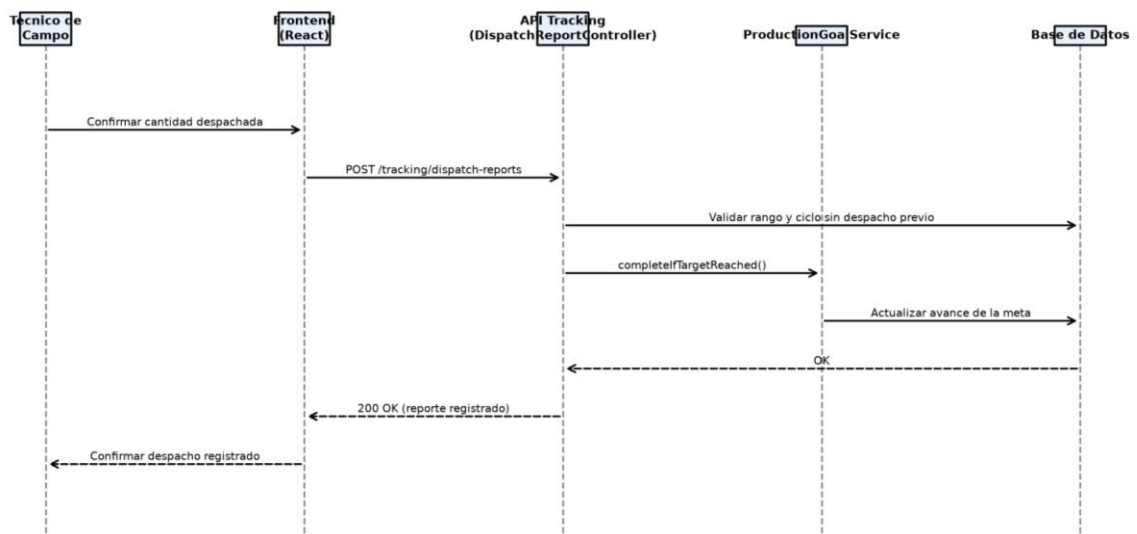


Ilustración 9: Diagrama de secuencia - Registro de Despacho

4.4.3.6 Diagramas de Estado

4.4.3.6.1 Diagrama de estado: Meta de Producción

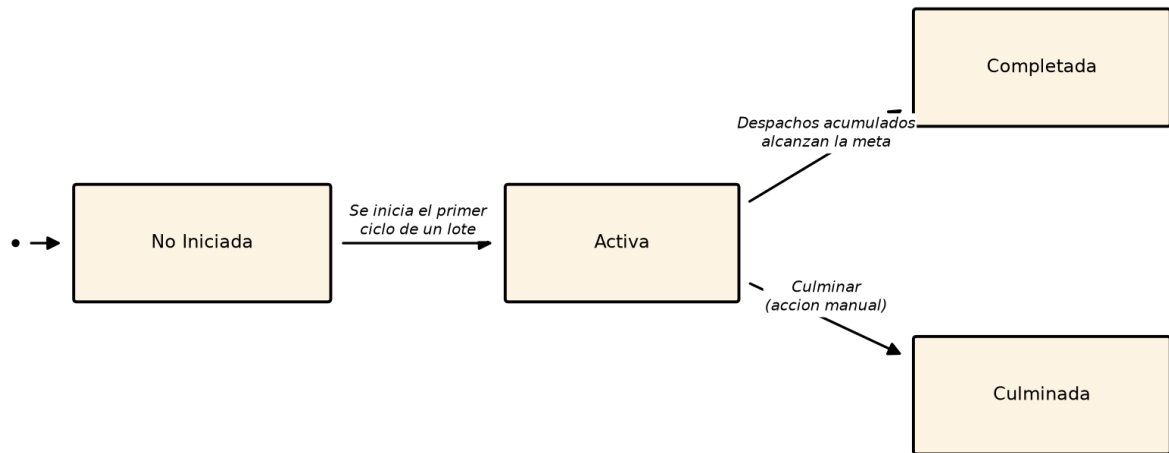


Ilustración 10: Diagrama de estado - Meta de Producción

4.4.3.6.2 Diagrama de estado: Lote

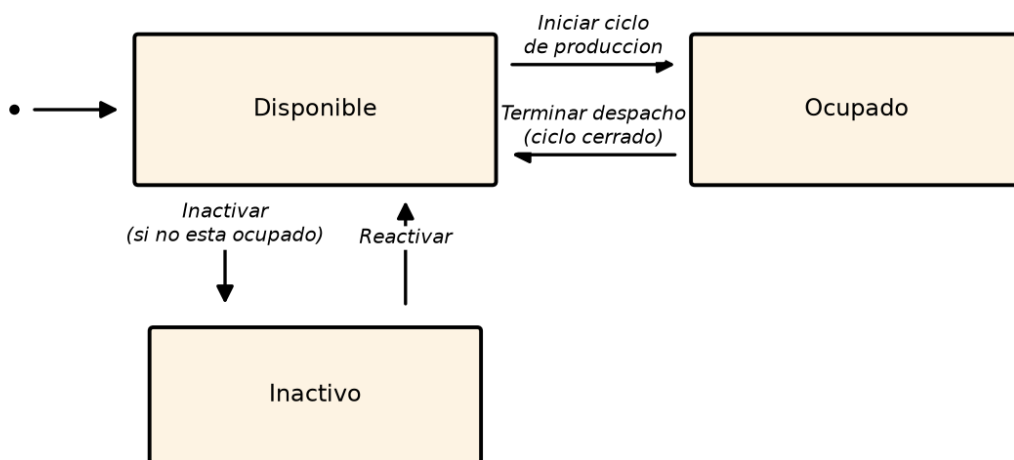


Ilustración 11: Diagrama de estado - Lote

4.4.3.6.3 Diagrama de estado: Ciclo de Producción del Lote (Fase Actual)

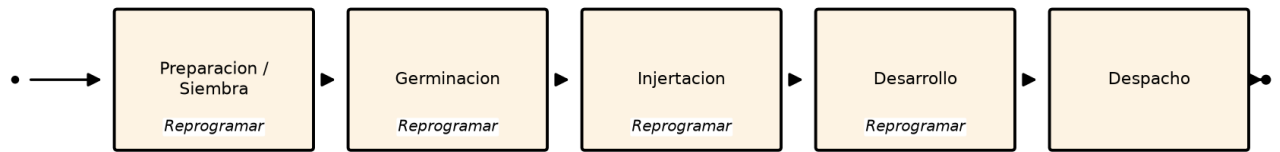


Ilustración 12: Diagrama de estado - Ciclo de Producción del Lote

Ilustración: Diagrama de estado: Ciclo de Producción del Lote (Fase Actual)

4.4.3.7 Diagramas de Base de Datos



Ilustración 13: Diagrama de Base de datos

4.4.4 Descripción Técnica / Arquitectura del Sistema

Esta sección detalla los componentes tecnológicos y las reglas de negocio que sustentan la reactividad del sistema y su capacidad para operar en entornos rurales con conectividad limitada.

4.4.4.1 Arquitectura del Sistema

El sistema ERP Lastenia trabaja bajo una arquitectura cliente-servidor de tipo Monolito Modular. El cliente (frontend) está desarrollado en React y se comunica mediante peticiones HTTP hacia una API REST construida en Laravel, la cual centraliza toda la lógica de negocio del sistema y expone los recursos que consume la interfaz para la planificación y el seguimiento de la producción del vivero.

En el backend, la arquitectura se organiza en capas: los Controllers reciben y validan las peticiones HTTP, los Services contienen la lógica de negocio y los algoritmos de planificación (cálculo de cronogramas, reprogramación en cascada, cierre de metas), los Repositories gestionan el acceso a los datos, y los Models representan las entidades persistidas en la base de datos MySQL. En el frontend se aplica el patrón MVVM, separando la vista de la lógica de presentación mediante hooks y viewmodels. Esta separación de responsabilidades permite que cada módulo del ERP (Planning, Tracking, y los módulos aún en diseño como Inventory, Logistics y Tasks) evolucione de forma independiente sin afectar al resto del sistema.

4.4.4.1.1 Tecnologías Utilizadas

- **React 19 con TypeScript:** Librería y superset tipado de JavaScript utilizados para construir la interfaz de usuario del sistema mediante componentes reutilizables y fuertemente tipados. Sobre ella se implementa el patrón MVVM del frontend, permitiendo que el tablero de control, el mapa de lotes y el calendario reaccionen de forma inmediata a los cambios de datos.
- **Vite:** Herramienta de compilación y empaquetado utilizada para el proyecto frontend, encargada de optimizar los tiempos de carga y de desarrollo de la aplicación web.

- **TailwindCSS:** Framework de estilos utilizado para construir una interfaz responsiva Mobile-First, adaptada tanto al uso en oficina como en el trabajo de campo del vivero.
- **Laravel 12:** Un Framework de PHP donde se utiliza la implementación de la API REST del sistema y la estructura modular con módulos independientes, cada uno de estos módulos tendrá sus propios controladores, servicios y modelos que permiten que el backend sea organizado y de fácil mantenimiento.
- **MySQL:** Base de datos que almacena la información principal dentro del sistema, datos tales como viveros, lotes, metas de producción, fases y despachos. Referente a cualquier cambio estructural realizado en el esquema de la base de datos se realizaría mediante migraciones de Laravel que mantiene controlada y rastreable los eventos que pasan en la base de datos
- **Capacitor + SQLite (Nodo Móvil):** Utilizados para la versión móvil del sistema, esta alternativa empaqueta el sistema frontend de react en una sola aplicación nativa de Android que se respalda con su respectiva base de datos SQLite local.

4.4.4.2 Requerimientos Funcionales

Se definen los requerimientos "Must Have" (obligatorios) para garantizar que el sistema resuelva los cuellos de botella del cálculo manual.

- Permitir que los usuarios registrados inicien sesión en el sistema.
- Registrar, editar, listar y eliminar viveros, generando automáticamente su set de fases de producción por defecto.
- Registrar una meta de producción por vivero dando énfasis en que la producción girará en torno a la misma.
- Registrar lotes que serán creados con un cálculo de su capacidad con base en operaciones geométricas.

- Mostrar un mapa interactivo donde se puedan visualizar y gestionar los lotes ya creados, teniendo una referencia visual más exacta de la distribución del vivero en el sistema.
- Generar de forma automática el cronograma por lotes con las respectivas fechas de las diferentes fases dentro de un ciclo productivo de un lote.
- Reprogramar manualmente fechas de fases que por eventualidades no se han podido cumplir de forma estandarizada, recalculando de forma progresiva a partir de la fecha modificada en el sistema y dando nuevas fechas para las siguientes fases.
- Configuración de parámetros establecidos por lotes junto a la visualización de la duración estimada de cada fase de producción dentro del lote.
- Mostrar indicadores clave, como distribución de plántulas por fases/lotes, asimismo como la proyección esperada y real de los despachos.
- Reflejar el despacho real de las plántulas en el progreso de la meta productiva.
- Historial consultable donde mediante filtros de tiempo y lotes se puedan visualizar ciclos productivos ya cerrados.
- Envío de notificaciones proactivas de eventos registrados en el sistema de tal modo que puedan definir con antelación cómo afrontarlos.

4.4.4.3 Requerimientos No Funcionales

- El sistema debe ser capaz de hacer el cálculo del cronograma completo en menos de segundos.
- La interfaz debe ser responsive y funcionar en diferentes tamaños de pantalla, desde computadoras hasta dispositivos móviles, priorizando el uso de este último.

- El sistema debe garantizar el continuo flujo de trabajo aun sin conexión a internet efectuando sincronizaciones cuando se hallan registrado cambios en un dispositivo que ha estado sin conexión.
- Las contraseñas guardadas en la base de dato deben estar cifradas.
- El sistema debe cumplir con requerimientos de seguridad estándares tales como el de evitar inyección SQL.
- El cambio entre pantallas debe ser fluido y sin recargas completas de página.
- El sistema debe mantener la trazabilidad de todo cambio realizado sobre un cronograma mediante un historial de reprogramaciones.
- La API debe responder bajo un formato JSON estandarizado para todas las peticiones.
- El sistema debe ser modular, es decir, los nuevos módulos como Inventario, Logística y Tareas deben poder integrarse fácilmente más adelante sin afectar ni dañar los módulos que ya están en producción.
- El backend debe seguir una arquitectura por capas (Controllers, Services, Repositories, Models) para facilitar su mantenimiento.

4.4.5 Roles y Responsabilidades

Tabla: Roles en el marco Scrum

Tabla 19: Roles en el marco Scrum

Rol en Scrum	Nombre	Responsabilidades
Product Owner	Verduga Calderon Heyder Javier	Define cual es la visión del producto, le da prioridad al backlog y valida los incrementos con los técnicos del vivero.

Scrum Master	Ing. Mendoza Villamar Rocío	Facilita ceremonias, elimina impedimentos técnicos y asegura el uso de arquitecturas dinámicas.
Equipo de Desarrollo	Verduga Calderon Heyder Javier	Diseña, codifica, prueba y despliega el sistema.

Tabla: Roles de Usuario en el Sistema

Tabla 20: Roles de Usuario en el Sistema

Rol en el sistema	Responsabilidades
Administrador / Docente	Gestionar el vivero con su respectiva meta de producción, aprovando cronogramas y ciclos de producción por lotes
Técnico de Campo	Consulta de actividades previamente creadas, con acceso al registro de avance por lotes y recepción de alertas.

4.4.6 Planificación del Sprint

La construcción del sistema se organiza en seis Sprints de dos semanas cada uno, enfocados en la entrega incremental de valor funcional.

4.4.6.1 Sprint 1: Diseño de la base de datos

Tabla 21: Sprint 1 - Diseño de base de datos

SPRINT 1: Diseño de la base de datos			
Duración:	2 semanas (05/01/2026 – 18/01/2026)	Prioridad:	Alta
Objetivo:	Diseñar e implementar el modelo de datos relacional en MySQL que sustente el motor de planificación automatizada, los cronogramas de las 5 fases técnicas y la trazabilidad de reprogramaciones, garantizando que la estructura soporte el patrón MVVM.		
Historias de usuario:	Base técnica para HU01 y HU02.		
Tareas de desarrollo:	• Creación de diagramas de entidad-relación que definen la estructura principal del sistema. • Generar las migraciones y módulos con Laravel basados en las relaciones y claves foráneas para mantener un orden y trazabilidad referentes a otros módulos del ERP. • Comprobar la integridad referencial y el rendimiento de las consultas para verificar que la base de datos soportara la lógica de planificación del sistema de forma consistente, fiable y eficiente.		
Plan de entrega	Modelado del diagrama entidad-relación. Diseñar la estructura de las tablas principales de la base de datos para viveros. Tiempo de desarrollo: 10 horas Fecha de inicio y fecha final: 05/01/2026 - 06/01/2026 Creación de migraciones y modelos. Implementar migraciones de Laravel y los modelos Eloquent correspondientes se construyen sobre esa estructura. Tiempo de desarrollo: 14 horas Fecha de inicio y fecha final: 07/01/2026 - 09/01/2026 Definición de relaciones y trazabilidad. Establecer las claves foráneas y la tabla de historial de reprogramaciones. Tiempo de desarrollo: 12 horas Fecha de inicio y fecha final: 12/01/2026 - 13/01/2026 Pruebas de integridad. Validar la integridad referencial y el tiempo de respuesta de las consultas de planificación. Tiempo de desarrollo: 8 horas Fecha de inicio y fecha final: 14/01/2026 - 18/01/2026		

4.4.6.2 Sprint 2: Motor de Planificación y Algoritmos

Tabla 22: Sprint 2 - Motor de Planificación y Algoritmos

SPRINT 2: Motor de Planificación y Algoritmos			
Duración:	2 semanas (19/01/2026 – 01/02/2026)	Prioridad:	Alta
Objetivo:	Implementar el núcleo algorítmico que automatiza el cálculo de las 5 fases técnicas del cacao a partir de la fecha de inicio del ciclo.		
Historias de usuario:	HU01: Registrar Meta de Producción. HU02: Calcular Cronograma Fenológico Automático.		
Tareas de desarrollo:	• Servicio de registro de metas de producción con validación de capacidad. • Algoritmo que calcula automáticamente el cronograma de las 5 fases. • Endpoints REST para crear y consultar cronogramas. • Pruebas unitarias sobre la lógica de cálculo de fechas.		
Plan de entrega	Servicio de metas de producción. Registrar metas validando que no exista otra meta activa al mismo tiempo. Tiempo de desarrollo: 12 horas Fecha de inicio y fecha final: 19/01/2026 - 20/01/2026 Algoritmo de cronograma automático. Desarrollar el cálculo encadenado de las fechas de las 5 fases a partir de la fecha de inicio. Tiempo de desarrollo: 30 horas Fecha de inicio y fecha final: 21/01/2026 - 25/01/2026 Endpoints REST de cronogramas. Exponer los endpoints de creación y consulta del cronograma por lote. Tiempo de desarrollo: 16 horas Fecha de inicio y fecha final: 26/01/2026 - 28/01/2026 Pruebas unitarias. Cubrir con pruebas unitarias la lógica de cálculo de fechas del motor de planificación. Tiempo de desarrollo: 10 horas Fecha de inicio y fecha final: 29/01/2026 - 01/02/2026		

4.4.6.3 Sprint 3: Dashboard Reactivo e Interfaz de Usuario (UI)

Tabla 23: Sprint 3 - Dashboard Reactivo e Interfaz de Usuario

SPRINT 3: Dashboard Reactivo e Interfaz de Usuario (UI)			
Duración:	2 semanas (02/02/2026 – 15/02/2026)	Prioridad:	Alta
Objetivo:	Desarrollar la interfaz reactiva en React y TailwindCSS, centrada en el mapa interactivo de lotes y el dashboard de resumen operativo del vivero.		
Historias de usuario:	HU04: Monitoreo de Lotes y Capacidad de Campo.		
Tareas de desarrollo:	• Desarrollo mediante la modularización siguiendo el patrón MVVM, lo que permitió separar la lógica de presentación del comportamiento de la interfaz y organizar el sistema de forma más mantenible. • El mapa interactivo de lotes se implementó con la función de arrastrar y soltar. • Desarrollo del panel de control se desarrolló para mostrar indicadores clave y un diagrama de distribución de fases, de modo que el usuario pueda consultar rápidamente el estado de la producción. • Ajustes al diseño responsivo mobile-first.		
Plan de entrega	Componentización MVVM. Estructurar el proyecto mediante vistas, hooks y modelos de vista siguiendo el patrón MVVM.. Tiempo de desarrollo: 14 horas Fecha de inicio y fecha final: 02/02/2026 - 04/02/2026 Mapa interactivo de lotes. Desarrollar mapa con funcionalidad de arrastrar y soltar, de modo que los gráficos puedan reposicionarse según la disposición actual del vivero. Tiempo de desarrollo: 26 horas Fecha de inicio y fecha final: 05/02/2026 - 09/02/2026 Dashboard de resumen. Creación de tarjetas KPI y gráficos donde el usuario pueda ver e interpretar de forma sencillas el estado productivo del vivero. Tiempo de desarrollo: 24 horas Fecha de inicio y fecha final: 10/02/2026 - 12/02/2026 Diseño responsivo. Ajustar la interfaz para el uso de pantallas móviles teniendo en cuenta su uso bajo el sol, que es un factor recurrente en el vivero. Tiempo de desarrollo: 12 horas Fecha de inicio y fecha final: 13/02/2026 - 15/02/2026		

4.4.6.4 Sprint 4: Motor de Reprogramación y Resiliencia.

Tabla 24: Sprint 4 - Motor de Reprogramación y Resiliencia

SPRINT 4: Motor de Reprogramación y Resiliencia.			
Duración:	2 semanas (16/02/2026 – 01/03/2026)	Prioridad:	Alta
Objetivo:	Implementar la lógica de reprogramación manual que permite ajustar la fecha de la fase actual de un lote y recalcular en cascada las fases posteriores.		
Historias de usuario:	HU03: Reprogramación Dinámica de Actividades.		
Tareas de desarrollo:	• Servicio de reprogramación con recálculo en cascada. • Registro del historial de reprogramaciones donde se registran el motivo y las fechas anteriores, lo que permite la trazabilidad del plan original. • Modal reprogramación se integra directamente en el calendario de lotes y los cambios se pueden realizar desde la misma vista de planificación.. • Regresión del cronograma recalculado para verificar que los cambios no afecten la coherencia del ciclo de producción.		
Plan de entrega	Servicio de reprogramación. El recálculo en cascada de las fases subsiguientes se produce a partir de la nueva fecha seleccionada. Tiempo de desarrollo: 22 horas Fecha de inicio y fecha final: 16/02/2026 - 19/02/2026 Historial de reprogramaciones. Registrar el motivo y las fechas anteriores de cada reprogramación realizada. Tiempo de desarrollo: 10 horas Fecha de inicio y fecha final: 20/02/2026 - 22/02/2026 Modal de reprogramación. Integrar la reprogramación a la vista de calendario de lotes en el frontend. Tiempo de desarrollo: 16 horas Fecha de inicio y fecha final: 23/02/2026 - 25/02/2026 Pruebas de regresión. Validar que el recálculo no altere las fases ya transcurridas del cronograma. Tiempo de desarrollo: 10 horas Fecha de inicio y fecha final: 26/02/2026 - 01/03/2026		

4.4.6.5 Sprint 5: Alertas y Notificaciones

Tabla 25: Sprint 5 - Alertas y Notificaciones

SPRINT 5: Alertas y Notificaciones			
Duración:	2 semanas (02/03/2026 – 15/03/2026)	Prioridad:	Media
Objetivo:	Desarrollar el sistema de notificaciones proactivas y la funcionalidad offline del Nodo Móvil (Capacitor + SQLite) para que el técnico de campo reciba alertas sin necesidad de conexión activa.		
Historias de usuario:	HU05: Recepción de Alertas Proactivas en Campo.		
Tareas de desarrollo:	• Configuración de la persistencia local del Nodo Móvil (SQLite vía Capacitor) para el funcionamiento offline. • Implementación del sistema de notificaciones push con 24 horas de anticipación. • Centro de notificaciones dentro de la aplicación. • Pruebas de sincronización de alertas al recuperar la conexión.		
Plan de entrega	Configuración del Nodo Móvil. Crear la versión móvil del proyecto con Capacitor y construir la base de datos SQLite local de la aplicación. Tiempo de desarrollo: 20 horas Fecha de inicio y fecha final: 02/03/2026 - 05/03/2026 Notificaciones push. Enviar notificaciones con 24 horas de anticipación a cada actividad o cada evento que requiera supervisión de un administrador o trabajador. Tiempo de desarrollo: 24 horas Fecha de inicio y fecha final: 06/03/2026 - 10/03/2026 Centro de notificaciones. Construir una sección para consultar y gestionar las alertas recibidas, un buzón de notificaciones. Tiempo de desarrollo: 14 horas Fecha de inicio y fecha final: 11/03/2026 - 12/03/2026 Pruebas de sincronización offline. Validar la sincronización de alertas pendientes al recuperar la señal de datos. Tiempo de desarrollo: 10 horas Fecha de inicio y fecha final: 13/03/2026 - 15/03/2026		

4.4.6.6 Sprint 6: Pruebas de Integración y Rendimiento.

Tabla 26: Sprint 6 - Pruebas de Integración y Registro de Despachos

SPRINT 6: Pruebas de Integración y Rendimiento.			
Duración:	2 semanas (16/03/2026 – 29/03/2026)	Prioridad:	Alta
Objetivo:	Implementar el módulo de seguimiento de despachos (Tracking) y ejecutar pruebas de integración sobre el sistema completo de planificación.		
Historias de usuario:	HU06: Consulta de Memoria Histórica de Producción.		
Tareas de desarrollo:	• Servicio para registrar los envíos reales y cerrar las metas de producción. • Módulo de consulta de datos históricos con filtros. • Pruebas de integración de extremo a extremo del motor de planificación. • Métricas comparativas entre el proceso manual y el sistema automatizado.		
Plan de entrega	Servicio de registro de despachos. Registrar los despachos reales realizados y avanza automáticamente en el seguimiento del objetivo de producción. Tiempo de desarrollo: 18 horas Fecha de inicio y fecha final: 16/03/2026 - 19/03/2026 Memoria histórica. Crear una consulta donde se pueda aplicar filtros por lote y tiempo para el historial de ciclos ya cerrados. Tiempo de desarrollo: 14 horas Fecha de inicio y fecha final: 20/03/2026 - 22/03/2026 Pruebas de integración. Probar y hacer la validación de todo el flujo de trabajo, desde que se crea un viero y lote hasta el despacho de plántulas al final de un ciclo productivo. Tiempo de desarrollo: 20 horas Fecha de inicio y fecha final: 23/03/2026 - 26/03/2026 Documentación de métricas. Consolidar documentado métricas de comparación de tiempo en cuanto a una gestión manual contra a la gestión dentro del sistema. Tiempo de desarrollo: 8 horas Fecha de inicio y fecha final: 27/03/2026 - 29/03/2026		

4.4.7 Backlog del Producto

El Product Backlog representa la lista ordenada y priorizada de todo lo que se conoce que es necesario en el producto. Es la única fuente de requisitos para cualquier cambio a realizarse en el sistema de planificación. A continuación, se detalla la evolución del backlog desde su estado inicial hasta el cierre de la fase de integración.

4.4.7.1 Backlog Inicial

Tabla 27: Backlog Inicial

Nro.	Tarea	Prioridad	Estado
1	Modelado del diagrama entidad-relación	Alta	Completada
2	Creación de migraciones y modelos Eloquent	Alta	Completada
3	Definición de relaciones y trazabilidad de reprogramaciones	Alta	Completada
4	Pruebas de integridad referencial	Alta	Completada
5	Servicio de registro de metas de producción	Alta	Completada
6	Algoritmo de cálculo automático del cronograma	Alta	Completada
7	Endpoints REST de creación y consulta de cronogramas	Alta	Completada
8	Pruebas unitarias del motor de planificación	Alta	Completada
9	Componentización de la interfaz bajo MVVM	Alta	Completada
10	Desarrollo del mapa interactivo de lotes	Alta	Completada
11	Desarrollo del dashboard de resumen operativo	Alta	Completada
12	Ajustes de diseño responsivo Mobile-First	Media	Completada
13	Servicio de reprogramación con recálculo en cascada	Alta	Completada
14	Registro del historial de reprogramaciones	Alta	Completada
15	Modal de reprogramación en el calendario de lotes	Media	Completada
16	Pruebas de regresión del cronograma recalculado	Alta	Completada
17	Configuración del Nodo Móvil (Capacitor + SQLite)	Media	Completada
18	Implementación de notificaciones push	Media	Completada
19	Centro de notificaciones en la aplicación	Media	Completada
20	Pruebas de sincronización de alertas offline	Media	Completada
21	Servicio de registro de despachos y cierre de metas	Alta	Completada

Nro.	Tarea	Prioridad	Estado
22	Módulo de consulta de memoria histórica	Media	Completada
23	Pruebas de integración end-to-end	Alta	Completada
24	Documentación de métricas comparativas	Media	Completada

4.4.7.2 Backlog: Fin del Sprint 1

En este corte se verifica la implementación de la estructura relacional que sustenta el sistema en MySQL.

Tabla 28: Backlog - Fin del Sprint 1

Nro.	Tarea	Prioridad	Estado
1	Modelado del diagrama entidad-relación	Alta	Completada
2	Creación de migraciones y modelos Eloquent	Alta	Completada
3	Definición de relaciones y trazabilidad de reprogramaciones	Alta	Completada
4	Pruebas de integridad referencial	Alta	Completada
5	Servicio de registro de metas de producción	Alta	Completada
6	Algoritmo de cálculo automático del cronograma	Alta	Completada
7	Endpoints REST de creación y consulta de cronogramas	Alta	Completada
8	Pruebas unitarias del motor de planificación	Alta	Completada
9	Componentización de la interfaz bajo MVVM	Alta	Completada
10	Desarrollo del mapa interactivo de lotes	Alta	Completada
11	Desarrollo del dashboard de resumen operativo	Alta	Completada
12	Ajustes de diseño responsivo Mobile-First	Media	Completada
13	Servicio de reprogramación con recálculo en cascada	Alta	Completada
14	Registro del historial de reprogramaciones	Alta	Completada
15	Modal de reprogramación en el calendario de lotes	Media	Completada
16	Pruebas de regresión del cronograma recalculado	Alta	Completada
17	Configuración del Nodo Móvil (Capacitor + SQLite)	Media	Completada
18	Implementación de notificaciones push	Media	Completada
19	Centro de notificaciones en la aplicación	Media	Completada
20	Pruebas de sincronización de alertas offline	Media	Completada

Nro.	Tarea	Prioridad	Estado
21	Servicio de registro de despachos y cierre de metas	Alta	Completada
22	Módulo de consulta de memoria histórica	Media	Completada
23	Pruebas de integración end-to-end	Alta	Completada
24	Documentación de métricas comparativas	Media	Completada

4.4.7.3 Backlog: Fin del Sprint 2

Refleja la culminación de la lógica algorítmica central que elimina el cálculo manual de 60 minutos por lote.

Tabla 29: Backlog - Fin del Sprint 2

Nro.	Tarea	Prioridad	Estado
1	Modelado del diagrama entidad-relación	Alta	Completada
2	Creación de migraciones y modelos Eloquent	Alta	Completada
3	Definición de relaciones y trazabilidad de reprogramaciones	Alta	Completada
4	Pruebas de integridad referencial	Alta	Completada
5	Servicio de registro de metas de producción	Alta	Completada
6	Algoritmo de cálculo automático del cronograma	Alta	Completada
7	Endpoints REST de creación y consulta de cronogramas	Alta	Completada
8	Pruebas unitarias del motor de planificación	Alta	Completada
9	Componentización de la interfaz bajo MVVM	Alta	Completada
10	Desarrollo del mapa interactivo de lotes	Alta	Completada
11	Desarrollo del dashboard de resumen operativo	Alta	Completada
12	Ajustes de diseño responsivo Mobile-First	Media	Completada
13	Servicio de reprogramación con recálculo en cascada	Alta	Completada
14	Registro del historial de reprogramaciones	Alta	Completada
15	Modal de reprogramación en el calendario de lotes	Media	Completada
16	Pruebas de regresión del cronograma recalculado	Alta	Completada
17	Configuración del Nodo Móvil (Capacitor + SQLite)	Media	Completada
18	Implementación de notificaciones push	Media	Completada
19	Centro de notificaciones en la aplicación	Media	Completada

20	Pruebas de sincronización de alertas offline	Media	Completada
21	Servicio de registro de despachos y cierre de metas	Alta	Completada
22	Módulo de consulta de memoria histórica	Media	Completada
23	Pruebas de integración end-to-end	Alta	Completada
24	Documentación de métricas comparativas	Media	Completada

4.4.7.4 Backlog: Fin del Sprint 3

Cierre del desarrollo de la interfaz de usuario moderna, minimalista y responsiva (Mobile-First).

Tabla 30: Backlog - Fin del Sprint 3

Nro.	Tarea	Prioridad	Estado
1	Modelado del diagrama entidad-relación	Alta	Completada
2	Creación de migraciones y modelos Eloquent	Alta	Completada
3	Definición de relaciones y trazabilidad de reprogramaciones	Alta	Completada
4	Pruebas de integridad referencial	Alta	Completada
5	Servicio de registro de metas de producción	Alta	Completada
6	Algoritmo de cálculo automático del cronograma	Alta	Completada
7	Endpoints REST de creación y consulta de cronogramas	Alta	Completada
8	Pruebas unitarias del motor de planificación	Alta	Completada
9	Componentización de la interfaz bajo MVVM	Alta	Completada
10	Desarrollo del mapa interactivo de lotes	Alta	Completada
11	Desarrollo del dashboard de resumen operativo	Alta	Completada
12	Ajustes de diseño responsivo Mobile-First	Media	Completada
13	Servicio de reprogramación con recálculo en cascada	Alta	Completada
14	Registro del historial de reprogramaciones	Alta	Completada
15	Modal de reprogramación en el calendario de lotes	Media	Completada
16	Pruebas de regresión del cronograma recalculado	Alta	Completada
17	Configuración del Nodo Móvil (Capacitor + SQLite)	Media	Completada
18	Implementación de notificaciones push	Media	Completada
19	Centro de notificaciones en la aplicación	Media	Completada

20	Pruebas de sincronización de alertas offline	Media	Completada
21	Servicio de registro de despachos y cierre de metas	Alta	Completada
22	Módulo de consulta de memoria histórica	Media	Completada
23	Pruebas de integración end-to-end	Alta	Completada
24	Documentación de métricas comparativas	Media	Completada

4.4.7.5 Backlog: Fin del Sprint 4

Finalización de la funcionalidad de flexibilidad algorítmica ante imprevistos ambientales.

Tabla 31: Backlog - Fin del Sprint 4

Nro.	Tarea	Prioridad	Estado
1	Modelado del diagrama entidad-relación	Alta	Completada
2	Creación de migraciones y modelos Eloquent	Alta	Completada
3	Definición de relaciones y trazabilidad de reprogramaciones	Alta	Completada
4	Pruebas de integridad referencial	Alta	Completada
5	Servicio de registro de metas de producción	Alta	Completada
6	Algoritmo de cálculo automático del cronograma	Alta	Completada
7	Endpoints REST de creación y consulta de cronogramas	Alta	Completada
8	Pruebas unitarias del motor de planificación	Alta	Completada
9	Componentización de la interfaz bajo MVVM	Alta	Completada
10	Desarrollo del mapa interactivo de lotes	Alta	Completada
11	Desarrollo del dashboard de resumen operativo	Alta	Completada
12	Ajustes de diseño responsivo Mobile-First	Media	Completada
13	Servicio de reprogramación con recálculo en cascada	Alta	Completada
14	Registro del historial de reprogramaciones	Alta	Completada
15	Modal de reprogramación en el calendario de lotes	Media	Completada
16	Pruebas de regresión del cronograma recalculado	Alta	Completada
17	Configuración del Nodo Móvil (Capacitor + SQLite)	Media	Completada
18	Implementación de notificaciones push	Media	Completada
19	Centro de notificaciones en la aplicación	Media	Completada

20	Pruebas de sincronización de alertas offline	Media	Completada
21	Servicio de registro de despachos y cierre de metas	Alta	Completada
22	Módulo de consulta de memoria histórica	Media	Completada
23	Pruebas de integración end-to-end	Alta	Completada
24	Documentación de métricas comparativas	Media	Completada

4.4.7.6 Backlog: Fin del Sprint 5

Implementación del soporte para trabajo en campo y notificaciones proactivas.

Tabla 32: Backlog - Fin del Sprint 5

Nro.	Tarea	Prioridad	Estado
1	Modelado del diagrama entidad-relación	Alta	Completada
2	Creación de migraciones y modelos Eloquent	Alta	Completada
3	Definición de relaciones y trazabilidad de reprogramaciones	Alta	Completada
4	Pruebas de integridad referencial	Alta	Completada
5	Servicio de registro de metas de producción	Alta	Completada
6	Algoritmo de cálculo automático del cronograma	Alta	Completada
7	Endpoints REST de creación y consulta de cronogramas	Alta	Completada
8	Pruebas unitarias del motor de planificación	Alta	Completada
9	Componentización de la interfaz bajo MVVM	Alta	Completada
10	Desarrollo del mapa interactivo de lotes	Alta	Completada
11	Desarrollo del dashboard de resumen operativo	Alta	Completada
12	Ajustes de diseño responsivo Mobile-First	Media	Completada
13	Servicio de reprogramación con recálculo en cascada	Alta	Completada
14	Registro del historial de reprogramaciones	Alta	Completada
15	Modal de reprogramación en el calendario de lotes	Media	Completada
16	Pruebas de regresión del cronograma recalculado	Alta	Completada
17	Configuración del Nodo Móvil (Capacitor + SQLite)	Media	Completada
18	Implementación de notificaciones push	Media	Completada
19	Centro de notificaciones en la aplicación	Media	Completada
20	Pruebas de sincronización de alertas offline	Media	Completada

21	Servicio de registro de despachos y cierre de metas	Alta	Completada
22	Módulo de consulta de memoria histórica	Media	Completada
23	Pruebas de integración end-to-end	Alta	Completada
24	Documentación de métricas comparativas	Media	Completada

4.4.7.7 Backlog: Fin del Sprint 6

Estado final del backlog tras el aseguramiento de calidad y validación de métricas de éxito.

Tabla 33: Backlog - Fin del Sprint 6

Nro.	Tarea	Prioridad	Estado
1	Modelado del diagrama entidad-relación	Alta	Completada
2	Creación de migraciones y modelos Eloquent	Alta	Completada
3	Definición de relaciones y trazabilidad de reprogramaciones	Alta	Completada
4	Pruebas de integridad referencial	Alta	Completada
5	Servicio de registro de metas de producción	Alta	Completada
6	Algoritmo de cálculo automático del cronograma	Alta	Completada
7	Endpoints REST de creación y consulta de cronogramas	Alta	Completada
8	Pruebas unitarias del motor de planificación	Alta	Completada
9	Componentización de la interfaz bajo MVVM	Alta	Completada
10	Desarrollo del mapa interactivo de lotes	Alta	Completada
11	Desarrollo del dashboard de resumen operativo	Alta	Completada
12	Ajustes de diseño responsivo Mobile-First	Media	Completada
13	Servicio de reprogramación con recálculo en cascada	Alta	Completada
14	Registro del historial de reprogramaciones	Alta	Completada
15	Modal de reprogramación en el calendario de lotes	Media	Completada
+16	Pruebas de regresión del cronograma recalculado	Alta	Completada
17	Configuración del Nodo Móvil (Capacitor + SQLite)	Media	Completada
18	Implementación de notificaciones push	Media	Completada
19	Centro de notificaciones en la aplicación	Media	Completada
20	Pruebas de sincronización de alertas offline	Media	Completada

21	Servicio de registro de despachos y cierre de metas	Alta	Completada
22	Módulo de consulta de memoria histórica	Media	Completada
23	Pruebas de integración end-to-end	Alta	Completada
24	Documentación de métricas comparativas	Media	Completada

4.4.8 Interfaz de Usuario (UI) / Prototipos

El diseño de la interfaz se fundamenta en un enfoque **minimalista y moderno**, utilizando los colores institucionales (verde #229e38, blanco #ffffff y negro #0f1911) para garantizar legibilidad bajo luz solar intensa. Se prioriza una arquitectura **Mobile-First**, respondiendo a que el personal revisará el cronograma principalmente desde el celular en el campo.

4.4.8.1 Mapa de Navegación del Sistema

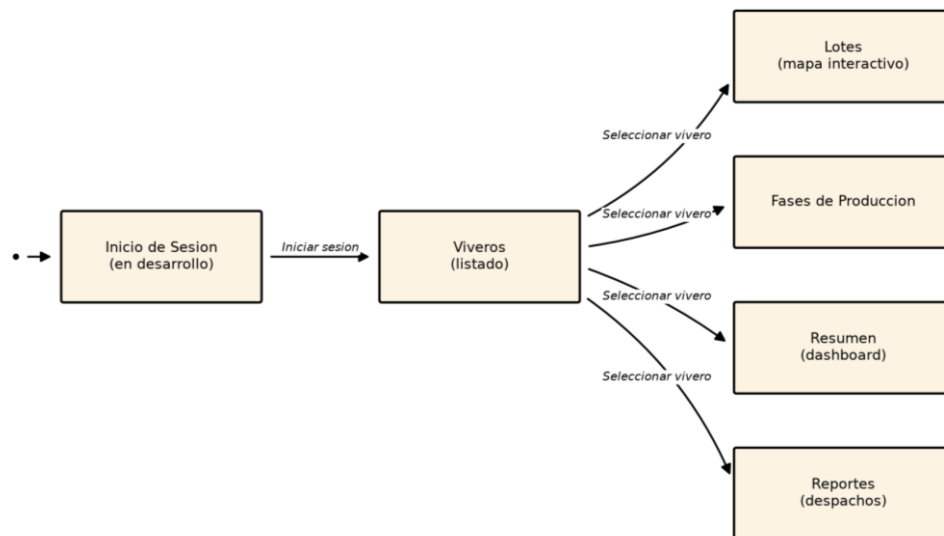


Ilustración 14: Mapa de Navegación del Sistema

4.4.8.2 Pantallas del Sistema

A continuación, se describen las vistas principales del módulo de Planificación y Seguimiento del sistema ERP Lastenia.

4.4.8.2.1 Pantalla de Inicio de Sesión

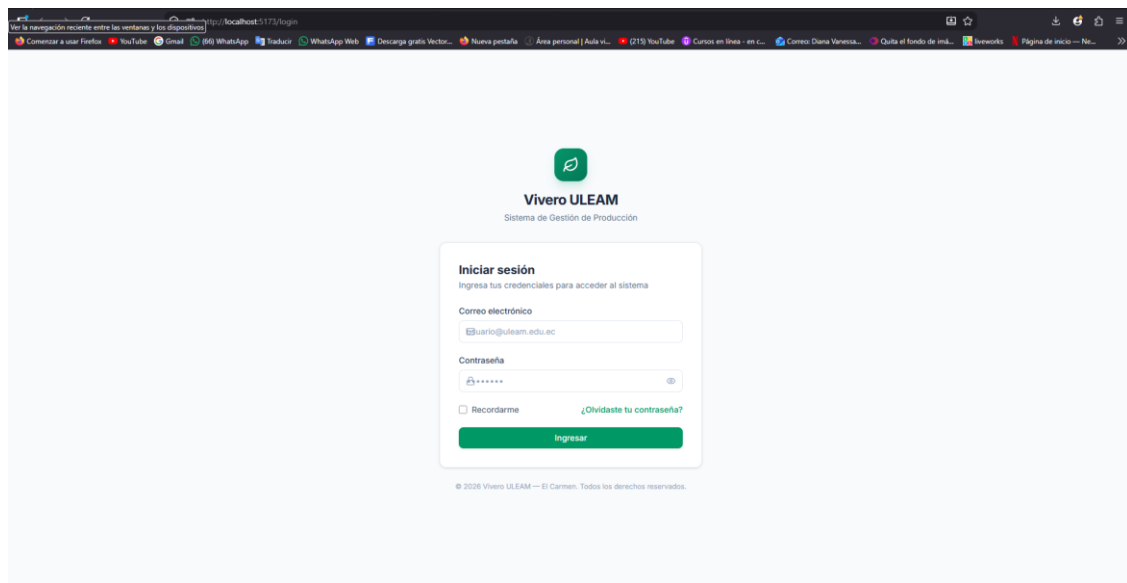


Ilustración 15: Pantalla de Inicio de Sesión

Aunque el módulo de autenticación se encuentra en desarrollo, esta pantalla permitirá el ingreso de credenciales del usuario para acceder al sistema, validando el rol correspondiente (Administrador, Docente planificador o Técnico de campo) antes de mostrar el tablero principal.

4.4.8.2.2 Pantalla de Viveros

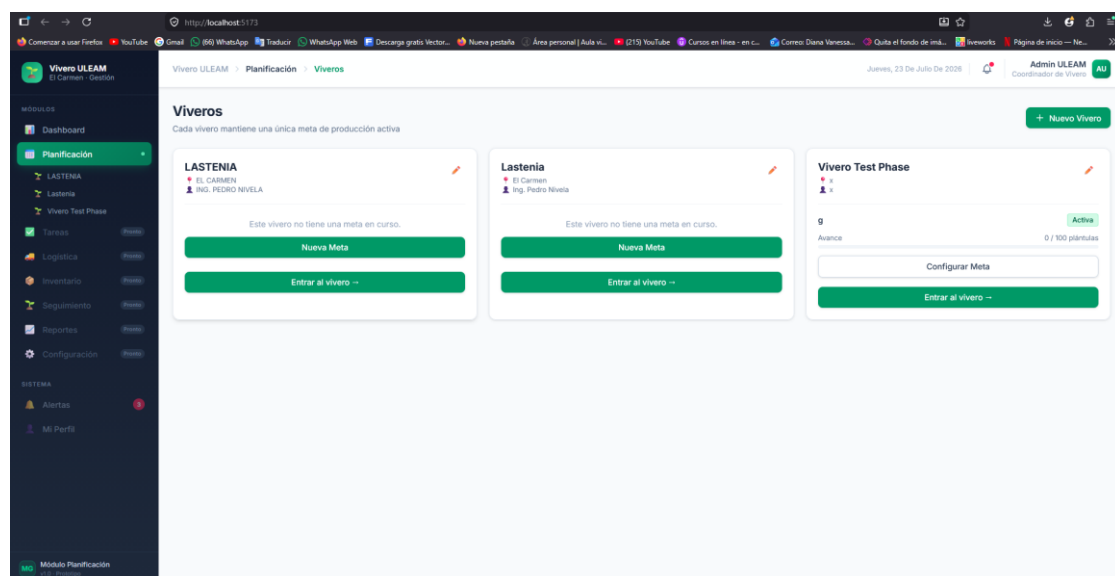


Ilustración 16: Pantalla creación de Viveros

Presenta el listado de viveros registrados en tarjetas, permitiendo crear un nuevo vivero, editarlo, y gestionar su meta de producción activa (crear, editar o culminar) desde un modal dedicado.

4.4.8.2.3 Pantalla de Lotes

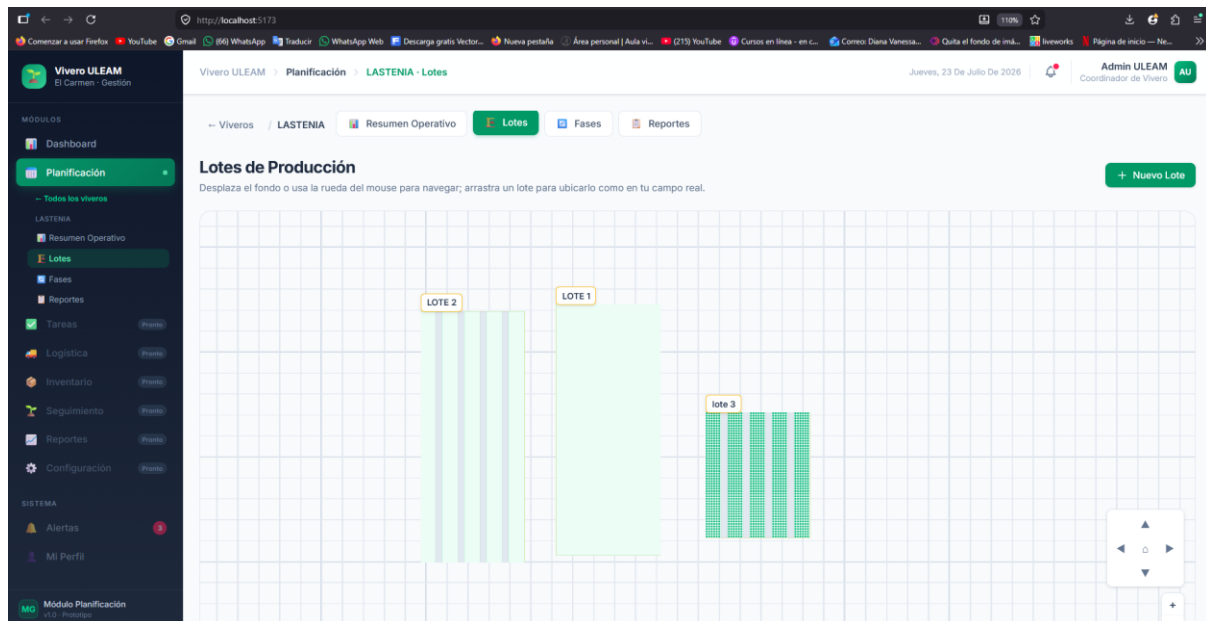


Ilustración 17: Pantalla Monitoreo de Lotes

Muestra el mapa interactivo del vivero, donde cada lote puede arrastrarse y reposicionarse para reflejar su ubicación física real. Desde el modal de detalle de un lote se puede iniciar un ciclo de producción, reprogramar su fase actual o terminar su despacho.

4.4.8.2.4 Pantalla de Fases de Producción

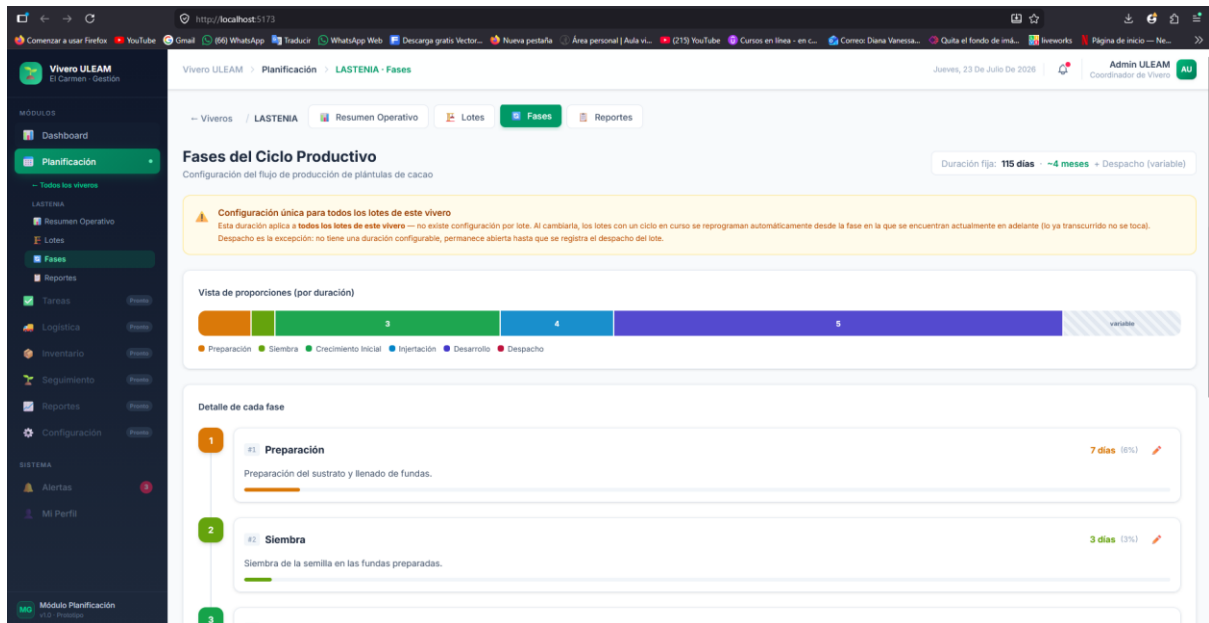


Ilustración 18: Pantalla de Fases de Producción

Los usuarios pueden personalizar el nombre, el orden, el color y la duración estimada de cada fase del ciclo de producción del vivero, y también se muestra la duración total acumulada del ciclo completo.

4.4.8.2.5 Pantalla de Resumen (Dashboard)

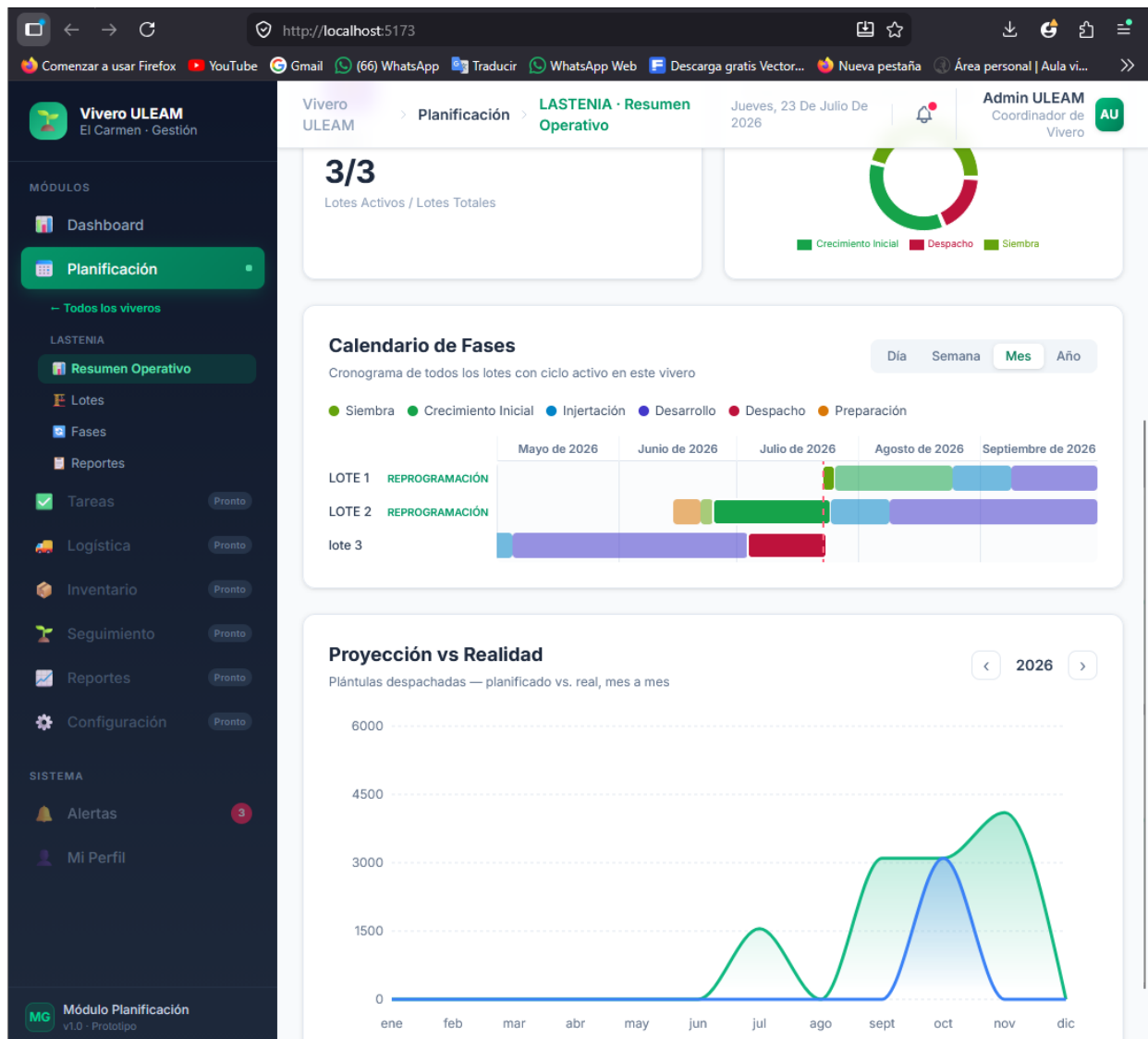


Ilustración 19: Pantalla de Resumen

Un panel de control con tarjetas indicadoras clave para el objetivo actual, los lotes en producción y la capacidad disponible incluye un gráfico de distribución de plántulas por fase, un calendario de lotes y un gráfico comparativo anual entre los envíos previstos y los reales.

4.4.9 Definición de Hecho (DoD)

Para asegurar la calidad y robustez del sistema, se han establecido criterios de aceptación que deben cumplirse antes de dar por terminado cualquier incremento dentro de los Sprints de Scrum.

4.4.9.1 Criterios Generales

- El código está escrito respetando las convenciones del proyecto (PSR-12 en el backend, ESLint en el frontend), sin errores de compilación.
- Todas las funcionalidades del módulo se han probado y corregido según lo solicitado en las historias de usuario.
- La interfaz es intuitiva y se adapta tanto a escritorio como a dispositivos móviles, siguiendo un diseño mobile-first.
- El frontend se comunica sin interrupciones con la API REST de Laravel.
- Las validaciones de los formularios están implementadas, son claras y funcionan correctamente.
- No hay errores de comunicación entre el cliente, la API y la base de datos.
- Cada endpoint responde en un formato JSON estandarizado y consistente.
- Se evaluaron los casos límite y excepcionales de cada funcionalidad antes de darla por terminada.
- Las consultas a la base de datos utilizan el ORM de Laravel (Eloquent), evitando inyecciones SQL.
- El código sigue la arquitectura por capas definida (Controllers, Services, Repositories, Models).

4.4.9.2 Criterios Específicos del Proyecto

- El cálculo del cronograma de las 5 fases se ejecuta correctamente a partir de la fecha de inicio del ciclo.

- Cálculo automático de la capacidad de cada lote según una operación geométrica basada en el alto y ancho del lote, incluyendo parámetros como el diámetro que ocuparía cada plántula, así como la cantidad y el ancho de corredores entre filas de plántulas.
- Reprogramación de la fecha de una fase la cual también hacer el recalcu lo automático de las nuevas fechas para las fases venidera a partir de la fecha reprogramada.
- Solo se permitiría tener una meta de producción a la vez por lote, asegurando así la trazabilidad de la meta impuesta.
- La meta de producción se actualiza y se registran avances en la misma según los despachos reales que se vayan registrando (módulo de seguimiento).
- El mapa de lotes refleja la posición y el estado real de cada lote del vivero.
- Toda reprogramación queda registrada con su motivo en el historial de cambios del cronograma.
- El dashboard muestra información actualizada de la capacidad y la distribución de los lotes por fase.
- Las contraseñas de los usuarios se almacenan cifradas en la base de datos.

4.4.10 Eventos Scrum

En esta sección se documentan las reuniones de revisión de cada ciclo de desarrollo, donde se validaron los incrementos funcionales con los interesados directos del vivero.

4.4.10.1 Sprint Review 1: Diseño de la base de datos

Fecha: 18/01/2026

Duración: 05/01/2026 - 18/01/2026

Participantes: Heyder Javier Verduga Calderón (Scrum Master / Developer), Ing. Mendoza Villamar Rocío Alexandra (Product Owner / Tutora), Ing. Pedro Nivelá (Stakeholder, coordinador del Vivero Lastenia).

4.4.10.1.1 Revisión de actividades

Se diseñó e implementó el esquema relacional en MySQL con las tablas de viveros, metas de producción, lotes, fases y ciclos, incluyendo las relaciones necesarias para la trazabilidad de reprogramaciones. Se validó la integridad referencial y el tiempo de respuesta de las consultas de planificación.

4.4.10.2 Sprint Review 2: Motor de Planificación y Algoritmos

Fecha: 01/02/2026

Duración: 19/01/2026 - 01/02/2026

Participantes: Heyder Javier Verduga Calderón (Scrum Master / Developer), Ing. Mendoza Villamar Rocío Alexandra (Product Owner / Tutora), Ing. Pedro Nivelá (Stakeholder, coordinador del Vivero Lastenia).

4.4.10.2.1 Revisión de actividades

Se implementó el algoritmo que calcula automáticamente las fechas de las 5 fases técnicas del cacao a partir de la fecha de inicio del ciclo, junto con los endpoints REST de creación y consulta de cronogramas. Las pruebas unitarias confirmaron la precisión del cálculo.

4.4.10.3 Sprint Review 3: Dashboard Reactivo e Interfaz de Usuario

Fecha: 15/02/2026

Duración: 02/02/2026 - 15/02/2026

Participantes: Heyder Javier Verduga Calderón (Scrum Master / Developer), Ing. Mendoza Villamar Rocío Alexandra (Product Owner / Tutora), Ing. Pedro Nivelá (Stakeholder, coordinador del Vivero Lastenia).

4.4.10.3.1 Revisión de actividades

Se desarrolló el mapa interactivo de lotes y el dashboard de resumen operativo bajo el patrón MVVM, con un diseño responsivo Mobile-First validado para el uso en campo.

4.4.10.4 Sprint Review 4: Motor de Reprogramación y Resiliencia

Fecha: 01/03/2026

Duración: 16/02/2026 - 01/03/2026

Participantes: Heyder Javier Verduga Calderón (Scrum Master / Developer), Ing. Mendoza Villamar Rocío Alexandra (Product Owner / Tutora), Ing. Pedro Nivelá (Stakeholder, coordinador del Vivero Lastenia).

4.4.10.4.1 Revisión de actividades

Se implementó el servicio de reprogramación manual con recálculo en cascada de las fases posteriores, junto con el registro del historial de cambios y su integración en el calendario de lotes del frontend.

4.4.10.5 Sprint Review 5: Alertas y Notificaciones / Nodo Móvil

Fecha: 15/03/2026

Duración: 02/03/2026 - 15/03/2026

Participantes: Heyder Javier Verduga Calderón (Scrum Master / Developer), Ing. Mendoza Villamar Rocío Alexandra (Product Owner / Tutora), Ing. Pedro Nivelá (Stakeholder, coordinador del Vivero Lastenia).

4.4.10.5.1 Revisión de actividades

Se avanzó en la configuración del Nodo Móvil (Capacitor + SQLite) para la operación offline, y en el sistema de notificaciones push con 24 horas de anticipación, junto con el centro de notificaciones dentro de la interfaz.

4.4.10.6 Sprint Review 6: Pruebas de Integración y Rendimiento

Fecha: 29/03/2026

Duración: 16/03/2026 - 29/03/2026

Participantes: Heyder Javier Verduga Calderón (Scrum Master / Developer), Ing. Mendoza Villamar Rocío Alexandra (Product Owner / Tutora), Ing. Pedro Nivelá (Stakeholder, coordinador del Vivero Lastenia).

4.4.10.6.1 Revisión de actividades

Se implementó el servicio de registro de despachos y el módulo de consulta de memoria histórica, y se ejecutaron las pruebas de integración end-to-end del motor de planificación completo, documentando las métricas comparativas frente al proceso manual.

4.4.11 Proceso de Pruebas

Se ejecutó un plan de pruebas sobre las funcionalidades del módulo de Planificación y Seguimiento, combinando pruebas de caja negra (validación de los formularios desde la perspectiva del usuario) y pruebas de caja blanca (validación de la lógica interna de los servicios del backend). Todas las pruebas descritas a continuación resultaron satisfactorias.

4.4.11.1 Pruebas de caja negra

4.4.11.1.1 Formulario de Registro de Meta de Producción

Tabla 34: Formulario de Registro de Meta de Producción

Nombre del campo	Tipo de campo	Valor permitido	Observación
Título de la meta	Text	Letras y espacios, máximo 100 caracteres.	Funciona correctamente y valida campos vacíos.
Cantidad objetivo	Number	Numero entero positivo mayor a 0.	El sistema no permite valores menores o igual a cero
Descripción	Textarea	Área de texto libre, máximo 255 caracteres.	Es un campo opcional, el registro puede quedar sin observaciones.

4.4.11.1.2 Formulario de Registro de Lote

Tabla 35: Formulario de Registro de Lote

Nombre del campo	Tipo de campo	Valor permitido	Observación
Nombre del lote	Text	Letras, números, signos y espacios.	Genera un código único automáticamente si se deja vacío.
Ancho / Largo	Number	Valores decimales positivos (metros).	Calcula la capacidad automáticamente al guardar.
Diámetro de funda	Number	Valor decimal positivo (cm).	Validado correctamente contra el ancho de los corredores.

4.4.11.1.3 Formulario de Reprogramación de Cronograma

Tabla 36: Formulario de Reprogramación de Cronograma

Nombre del campo	Tipo de campo	Valor permitido	Observación
Nueva fecha	Date	Fecha posterior a la fecha actual de la fase.	El sistema no permite fechas anteriores a la fecha de la fase en curso.

Nombre del campo	Tipo de campo	Valor permitido	Observación
Motivo de reprogramación	Textarea	Área de texto libre y de máximo 255 caracteres.	Campo obligatorio y se valida que no quede vacío.

4.4.11.2 Pruebas de caja blanca

4.4.11.2.1 Formulario de Registro de Meta de Producción

Tabla 37: Prueba de caja blanca - Formulario de Registro de Meta de Producción

Método	Acción esperada	Acción obtenida	Observación
ProductionGoalService::create()	Validar que no exista ya una meta activa asignada al vivero.	Lanza una excepción de negocio (409) si ya existe una meta de producción para el vivero.	Funcionó correctamente en todas las pruebas realizadas.
Validación de cantidad objetivo	Rechazar valores que sean iguales o menores que cero.	Responde con un error de validación (422).	Sin observaciones.

4.4.11.2.2 Formulario de Registro de Lote

Tabla 38: Prueba de caja blanca - Formulario de Registro de Lote

Método	Acción esperada	Acción obtenida	Observación
LotService::calculateCapacity()	Calcular la capacidad de cada lote a partir de un cálculo geométrico.	Devuelve el valor numérico real de la cantidad del lote en todas las pruebas realizadas.	La precisión del cálculo ha sido validada frente al cálculo manual.
LotRepository::generateUniqueCode()	Generar un código único para el lote creado.	Devuelve el código generado, el cual nunca es repetido ni duplicado de alguno ya existente.	Funcionó sin conflictos, genera códigos usables en otros módulos.

4.4.11.2.3 Formulario de Reprogramación de Cronograma

Tabla 39: Prueba de caja blanca - Formulario de Reprogramación de Cronograma

Método	Acción esperada	Acción obtenida	Observación
LotCycleService::reschedule()	Recalcular de forma progresiva fases posteriores a una reprogramación.	Actualiza correctamente las fechas de las fases posteriores.	Las fases ya transcurridas no se ven afectadas.

4.4.11.2.4 Validación de progreso de Meta de Producción

Tabla 40: Prueba de caja blanca - Validación de Progreso de Meta de Producción

Método	Acción esperada	Acción obtenida	Observación
ProductionGoalService::completeIfTargetReached()	Completar la meta cuando el despacho acumulado alcanza el objetivo.	Cambia el estado de la meta a 'Completada' automáticamente.	Validado con datos reales de prueba.

4.4.12 Incremento y Entregables

A continuación, se detalla el incremento funcional entregado al finalizar cada Sprint, correspondiente al software potencialmente entregable del módulo de Planificación y Seguimiento del vivero.

4.4.12.1 Sprint 1: Diseño de la Base de Datos

Entregable: Modelo de datos relacionales para MySQL el cual dará soporte al motor de planificación automatizada y específicamente diseñada para ser utilizado como servicios del backend por otros módulos del sistema ERP.

Componente: Deja definido cuales son las entidades básicas del módulo, tales como viveros, metas de producción, lotes, fases productivas y ciclos por lote, dichos datos establecidos que puedan almacenarse y relacionarse de forma consistente.

Estado Funcional: La base de datos permite registrar viveros, metas de producción y lotes, y sustenta correctamente las consultas de planificación con tiempos de respuesta óptimos, aunque todavía sin exponer las funcionalidades a través de pantallas.

4.4.12.2 Sprint 2: Motor de Planificación y Algoritmos

Entregable: Motor de cálculo automático del cronograma de producción capaz de generar las 5 fases técnicas de la producción de cacao a partir de la fecha de inicio del ciclo.

Componente: El módulo cuenta con el servicio `ProductioGoalService` para que se registren y validen metas de producción por vivero, con el algoritmo `LotCycleService::startCycle()` con algoritmo principal para calcular y vincular fechas de las fases y consultar cronogramas de producción basados en la Meta de producción.

Estado Funcional: El sistema permite crear una Meta de producción para posteriormente iniciar ciclos productivos bajo este objetivo propuesto.

4.4.12.3 Sprint 3: Dashboard Reactivo e Interfaz de Usuario

Entregable: Una interfaz dinámica y reactiva para el módulo de planificación, la cual contiene un mapa interactivo de lotes y un panel de resumen dinámico que permite al usuario visualizar el estado productivo del vivero.

Componente: Incluye vistas renderizadas de React como `ViverosPage`, `LotesPage`, `lotWorkSpace`, `ResumenPage`, todas desarrolladas bajo el patrón MVVM con hooks y viewmodels respectivamente.

Estado Funcional: Se le permite al usuario visualizar indicadores con métricas de interés, junto a la gestión de lotes interactiva con funcionalidad de reposicionamiento (arrastrar y soltar) para mayor interactividad y visualización de los lotes.

4.4.12.4 Sprint 4: Motor de Reprogramación

Entregable: Funcionalidad para reprogramar manualmente alguna fase del ciclo productivo el cual recalcula progresivamente fechas de las demás fases apartir de la fecha reprogramada.

Componente: Función LotCycleService::reschedule() usada para recalcular las fases subsiguientes a partir de la nueva fecha que se ha modificado y con LorCucleReschedule registra para guardar en la base de datos el historial de cambios.

Estado Funcional: El usuario puede reprogramar fechas de una fase activa en un ciclo de producción, cubriendo espacios de atrasos o adelantamientos de fechas generadas por eventos cambiantes dentro del vivero.

4.4.12.5 Sprint 5: Alertas y Notificaciones

Entregable: Un sistema de notificaciones para eventos del módulo que requieran ser recordadas con al menos 24 horas de antelación para una coordinación oportuna de los operarios del vivero.

Componente: Incluye notificaciones PUSH con aviso previo a un evento con fecha estipulada en el sistema, este tendrá su centro de notificaciones donde visualizar como si de un buzón se tratase todas las notificaciones que se generan el sistema.

Estado Funcional: El sistema notifica al administrador y operario de campo con 24 horas de antelación de eventos que se suscitaran dentro del vivero, se puede generar una consulta de todas las notificaciones enviadas.

4.4.12.6 Sprint 6: Pruebas de Integración y Registro de Despachos

Entregable: Módulo de seguimiento de despachos y validación completa del sistema de planificación mediante pruebas de integración.

Componente: DispatchReportService para registrar los despachos reales y cerrar automáticamente los objetivos de producción cuando se alcanza la cantidad objetivo.

Estado Funcional: El sistema permite registrar el despacho real de un lote, actualiza automáticamente el avance de la meta de producción y permite consultar la memoria histórica de ciclos cerrados.

CAPÍTULO V

5. EVALUACIÓN DE RESULTADOS

5.1 Introducción

Las pruebas realizadas para el sistema web de planificación constan de dos dimensiones: pruebas funcionales y no funcionales. Las pruebas funcionales midieron la eficiencia y precisión del algoritmo de planificación automática para proyectar las 5 fases técnicas del cacao (siembra, germinación, injerto, desarrollo y despacho) y la capacidad de reprogramación dinámica ante imprevistos climáticos. Estas pruebas demostraron que el sistema procesa cronogramas con una precisión técnica superior a los métodos manuales.

Por otro lado, las pruebas no funcionales se centraron en la usabilidad móvil (Mobile-First), la reactividad de la interfaz mediante el patrón MVVM. La funcionalidad offline en campo, prevista para el Nodo Móvil (Capacitor + SQLite), queda fuera del alcance de las pruebas de este incremento por corresponder a una fase posterior del proyecto. Se compararon los métodos tradicionales (Excel/papel) con la solución implementada para cuantificar la optimización en tiempos de respuesta y precisión de datos.

5.2 Presentación y monitoreo de resultados

Se establecieron criterios de validación para los procesos más críticos del módulo de planificación, definiendo los métodos de medición y los resultados esperados para mitigar la "discontinuidad estructural" detectada en el diagnóstico.

5.2.1 Planificación de la Evaluación

Tabla 41: Planificación de la Evaluación

Proceso evaluado	Método de validación	Resultado esperado
Cálculo de cronograma	Comparar el tiempo de cálculo manual frente al tiempo de respuesta del sistema en 6 lotes de prueba.	El sistema debe generar un cronograma completo para el lote en menos de 5 segundos.

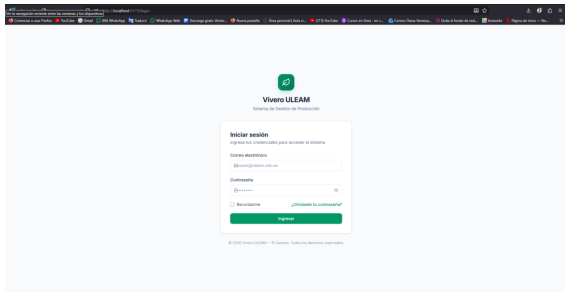
Proceso evaluado	Método de validación	Resultado esperado
Reprogramación de actividades	Se compara cuánto tarda en recalcular el cronograma a mano, comparado a hacerlo automáticamente en el sistema cuando cambia una fecha.	El recálculo del cronograma del lote no debe tomar más de 10 segundos.
Registro de despachos	Se compara el tiempo de registrar el despacho de plántulas manualmente frente a hacerlo automáticamente dentro sistema.	Registrar el despacho seguido de actualizar la meta automáticamente debe tomar menos de 10 segundos.

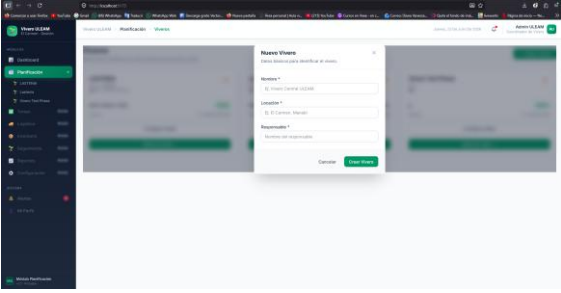
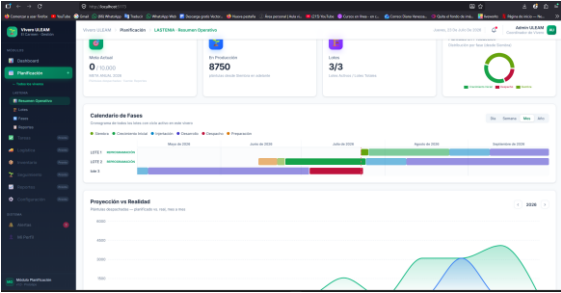
5.2.2 Ejecución del Monitoreo

Este proceso evalúa la capacidad del "motor de planificación" para transformar un objetivo de producción en un plan estructurado.

5.2.2.1 Planificación Automática de Cronogramas

Tabla 42: Planificación Automática de Cronogramas

1	 <i>Ilustración 20: Ingreso de Usuario en el login</i>	El usuario (docente o administrador) inicia sesión mediante autenticación JWT para acceder al módulo estratégico.
---	--	---

2	 <i>Ilustración 21: Definir meta de producción del Vivero</i>	Se define la meta global (ej. 5,000 plántulas CCN-51) y la fecha de siembra inicial en la interfaz reactiva.
3	 <i>Ilustración 22: Ejecución de algoritmo de calculo de fases</i>	El sistema ejecuta el algoritmo de 5 fases, asigna los lotes disponibles y genera las alertas proactivas con 24h de antelación.

Este proceso evalúa la capacidad del motor de planificación para calcular automáticamente el cronograma de las 5 fases técnicas al iniciar un ciclo de producción, comparando el tiempo frente al cálculo manual que se realizaba anteriormente en hojas de cálculo.

5.2.2.1.1 Toma de datos con el sistema y sin el sistema: Planificación de Cronograma

Tabla 43: Toma de datos Planificación de Cronogramas

Nro.	Lote evaluado	Tiempo manual	Tiempo con el sistema	Observación
1	Lote A1	00:52:00	00:00:03	Cálculo correcto
2	Lote A2	00:47:00	00:00:02	Sin errores
3	Lote B1	00:55:00	00:00:04	Tiempo óptimo
4	Lote B2	00:44:00	00:00:02	Cálculo inmediato
5	Lote C1	00:51:00	00:00:03	Sin observaciones
6	Lote C2	00:49:00	00:00:03	Respuesta rápida

Promedio	-	00:49:40	00:00:03	Reducción superior al 99%
----------	---	----------	----------	---------------------------

5.2.2.2 Reprogramación de Cronograma

Este proceso evalúa la capacidad del sistema para recalcular en cascada el cronograma de un lote cuando se reprograma manualmente la fecha de su fase actual, frente al recálculo manual de todas las fases posteriores.

5.2.2.2.1 Toma de datos con el sistema y sin el sistema: Reprogramación de Cronograma

Tabla 44: Toma de datos Reprogramación del Cronograma

Nro.	Lote evaluado	Tiempo manual	Tiempo con el sistema	Observación
1	Lote A1	00:19:00	00:00:05	Recálculo correcto
2	Lote A2	00:22:00	00:00:07	Sin errores
3	Lote B1	00:18:00	00:00:05	Tiempo óptimo
4	Lote B2	00:21:00	00:00:06	Cascada correcta
5	Lote C1	00:20:00	00:00:06	Sin observaciones
6	Lote C2	00:20:00	00:00:07	Historial registrado
Promedio	-	00:20:00	00:00:06	Reducción superior al 99%

5.2.2.3 Registro de Despachos

Este proceso evalúa el tiempo de registrar el despacho real de un lote y actualizar el avance de la meta de producción, frente al registro manual en cuadernos u hojas de cálculo utilizado anteriormente en el Vivero Lastenia.

5.2.2.3.1 Toma de datos con el sistema y sin el sistema: Validación de Despachos

Tabla 45: Toma de datos Validación de Despachos

Nro.	Lote evaluado	Tiempo manual	Tiempo con el sistema	Observación
1	Lote A1	00:11:00	00:00:04	Meta actualizada
2	Lote A2	00:13:00	00:00:05	Sin errores
3	Lote B1	00:10:00	00:00:04	Tiempo óptimo
4	Lote B2	00:12:00	00:00:06	Registro correcto

5	Lote C1	00:14:00	00:00:05	Sin observaciones
6	Lote C2	00:11:00	00:00:04	Meta completada
Promedio	-	00:11:50	00:00:05	Reducción superior al 99%

5.2.2.4 Tabla comparativa

Resumen comparativo del impacto de la solución tecnológica frente a la gestión tradicional del vivero.

Tabla 46: Tabla comparativa sin sistema y con sistema

Caso	Sin sistema	Con sistema
Calcular el cronograma completo de un lote con sus ciclos productivos.	Cálculo manual en aproximadamente 49 min	Cálculo automático hecho por el sistema en 3 segundos.
Reprogramar fecha de una fase y cálculo automático de fases subyacentes.	Cálculo manual en aproximadamente 20 minutos	Cálculo automático y progresivo en 5 segundos.
Visualización de los lotes creados	Se necesita el recorrido en físico del vivero.	Mapa interactivo reajutable, una representación digital de la disposición de lotes del vivero
Trazabilidades ante cambios	Sin historial registrado que pueda servir como base de decisiones para futuras producciones	Se genera el historial automáticamente a medida se va operando el sistema.

Tabla: Comparativa de eficiencia (manual vs. sistema web)

5.3 Interpretación Objetiva

Para el vivero Lastenia, el cronograma de producción se calculaba manualmente en un promedio de 49 minutos, mientras que el sistema lo hace en 3 segundos, eliminando el principal cuello de botella administrativo detectado en la fase de diagnóstico. Lo mismo ocurre con la reprogramación en caso de imprevistos.

Ajustar manualmente todas las fases posteriores requería alrededor de 20 minutos, mientras que ahora el sistema recalcula el cronograma actualizado en un promedio de 6 segundos. De manera similar, el registro de despachos y la actualización de la meta de producción se redujo de aproximadamente 12 minutos a 5 segundos. Estos resultados confirman que la arquitectura reactiva del sistema reduce significativamente el tiempo administrativo y el riesgo de errores humanos en la planificación del vivero.

CAPITULO VI

6 CONCLUSIONES Y RECOMENDACIONES

6.1 Conclusiones

La culminación de esta etapa del proyecto de titulación permite determinar que la implementación del módulo de Planificación y Seguimiento del sistema ERP Lastenia constituye un avance concreto frente a la gestión manual que afectaba al Vivero Lastenia. Tras el desarrollo y las pruebas realizadas, y en relación directa con los objetivos específicos planteados en el Capítulo I, se establecen las siguientes conclusiones:

La recopilación bibliográfica permitió fundamentar teóricamente la elección de una arquitectura frontend dinámica bajo el patrón MVVM y de una arquitectura backend en capas (Controllers, Services, Repositories, Models), sustentando técnicamente las decisiones de diseño del sistema.

Se identificó con precisión los requerimientos reales: automatizar el cálculo del cronograma de producción, poder reprogramar actividades ante imprevistos y contar con un registro centralizado de lotes y despachos. Estos hallazgos fueron los que orientaron el diseño de las historias de usuario del sistema.

Se logró implementar el módulo digital de Planificación y Seguimiento. Este permite registrar metas de producción y lotes, calcular automáticamente el cronograma de las 5 fases técnicas, reprogramar actividades con recálculo en cascada, visualizar el estado de los lotes en un mapa interactivo y registrar los despachos reales, con lo cual un proceso que antes tomaba hasta 49 minutos por lote pasó a resolverse en segundos.

Las pruebas de caja negra y caja blanca confirmaron que las funcionalidades implementadas cumplen con los criterios de aceptación definidos. Sin embargo, como el sistema todavía está en desarrollo activo y quedan pendientes varios módulos complementarios (autenticación, Inventario, Logística, Tareas y Sincronización), validar el sistema con usuarios finales en un ciclo de producción real queda como el siguiente paso inmediato del proyecto.

6.2 Recomendaciones

Con base en los resultados obtenidos y en el estado actual de desarrollo del sistema, se proponen las siguientes recomendaciones para asegurar la continuidad y sostenibilidad del proyecto:

Para el equipo de desarrollo, se recomienda priorizar la implementación del módulo de autenticación (inicio de sesión, roles y permisos), actualmente pendiente, ya que es un requisito previo para habilitar el acceso diferenciado de Administradores, Docentes planificadores y Técnicos de campo al sistema.

Sobre los módulos complementarios, conviene seguir desarrollando Inventario, Logística de Aproveccionamiento, Administración de Tareas y Sincronización bajo la misma arquitectura modular ya usada en Planning y Tracking, para que el ERP se consolide como una plataforma integral de gestión del vivero.

Para el nivel administrativo y académico de la ULEAM, se recomienda planificar la adopción institucional del sistema una vez esté habilitado el módulo de autenticación, comenzando por registrar un ciclo productivo real que alimente la memoria histórica del vivero y permita validar el sistema en condiciones de trabajo reales.

En síntesis, con el mantenimiento técnico, es importante establecer un protocolo de copias seguridad periódicas para la base de datos MySQL. Además, se debe avanzar en la implementación del Nodo Móvil (Capacitor + SQLite), de manera que el procedimiento continúe funcionando y no sea afectado en zonas del campo sin conexión que sea limitada.

En ese contexto, se recomienda la capacitación del personal operativo del Vivero del Lastenia, después que estén disponibles las pantallas de autenticación y notificaciones, para el uso del sistema priorizando reuniones breves que faciliten a los técnicos del campo incorporen la herramienta a las actividades diarias.

BIBLIOGRAFÍA

- Acosta-Vargas, Patricia, Salvador-Acosta, Belén, Salvador-Ullauri, Luis, Villegas-Ch., William, y Gonzalez, Mario (2021). Accessibility in Native Mobile Applications for Users with Disabilities: A Scoping Review. *Applied Sciences*, 11(12). <https://doi.org/10.3390/app11125707>
- Ajayi, Victor Oluwatosin (2023). A Review on Primary Sources of Data and Secondary Sources of Data. *SSRN*. <https://doi.org/https://dx.doi.org/10.2139/ssrn.5378785>
- Alase, Abayomi (2017). The Interpretative Phenomenological Analysis (IPA): A Guide to a Good Qualitative Research Approach. *International Journal of Education and Literacy Studies*, 5(2), 9-91.
- Alvarado, Zabala., Carrera Maridueña, y López Goyez. (2025). Tecnologías al servicio del Agro: Impacto de un sistema web para la gestión del mantenimiento en cultivos de cacao. *Universidad y Sociedad*, 17(3), e5162. <https://doi.org/https://orcid.org/0000-0002-9101-7435>
- Alvarado, Zabala., Carrera Maridueña., y López Goyez. (2025). Tecnologías al servicio del Agro: Impacto de un sistema web para la gestión del mantenimiento en cultivos de cacao. *Universidad y Sociedad*, 17(3). <https://doi.org/https://rus.ucf.edu.cu/index.php/rus/article/view/5162>
- Alzate, Paola., y Giraldo, David (2023). Tendencias de investigación del blockchain en la cadena de suministro: transparencia, trazabilidad y seguridad. *Revista Universidad & Empresa*, 25(44), 1-29. <https://doi.org/https://doi.org/10.12804/revistas.urosario.edu.co/empresa/a.12451>
- Anwar, Adeem Ali (2022). A survey of semantic web (Web 3.0), its applications, challenges, future and its relation with Internet of things (IoT). *Web Intelligence*, 20(3), 173–202. <https://doi.org/https://doi.org/10.3233/WEB-210491>

- Ara, Jinat, Sik-Lanyi, Cecilia, y Kelemen, Arpad (2024). Accessibility engineering in web evaluation process: a systematic literature review. *Universal Access in the Information Society (Springer)*, 23, 653–686. <https://doi.org/10.1007/s10209-023-00967-2>
- Bhimanapati, Viharika, Goel, Punit, y Jain, Ujjawal (2024). Leveraging Selenium and Cypress for Comprehensive Web Application Testing. *Journal of Quantum Science and Technology (JQST)*, 1(1), 66–79. <https://doi.org/10.36676/jqst.v1.i1.10>
- Busetto, Loraine, Wick, Wolfgang, y Gumbinger, Christoph (2020). How to use and assess qualitative research methods. *Neurological Research and Practice*, 2. <https://doi.org/https://doi.org/10.1186/s42466-020-00059-z>
- Can, Çiğdem Selçuk, y Gökteş, Büşra Nur (2023). Comparative study of front-end web development frameworks: Angular, React, and Vue.js. *Procedia Computer Science (Elsevier)*, 218, 2530–2539. <https://doi.org/10.1016/j.procs.2023.01.096>
- Chiriboga-Zamora (2024). Gestión Inteligente en el Agro: Un Enfoque Basado en Business Intelligence para Asociaciones de Producción Agrícola en Riobamba. *Polo del Conocimiento*, 9(2), 347-364. <https://doi.org/https://mail.polodelconocimiento.com/ojs/index.php/es/article/view/6531>
- Corella-Parra. (2023). Desarrollo de un sistema de control de inventario para una empresa comercializadora de sistemas de riego. *Ingeniería, Investigación y Tecnología*, 24(1), 1-10. <https://doi.org/https://doi.org/10.22201/fi.25940732e.2023.24.1.006>
- Creswell, John Ward, y Creswell, David (2018). *Research Design: Qualitative, Quantitative, and Mixed Methods Approaches*. SAGE Publications.
- Custodio, Jose Miguel, Billones, Robert Kerwin, Concepcion, Ronnie, y Vicerra, Ryan Rhay (2024). Optimization of crop harvesting schedules and land allocation through linear programming. *Process Integration and Optimization for Sustainability*, 8(1), 123-134. <https://doi.org/https://doi.org/10.1007/s41660-023-00357-4>

- De León López. &. (2024). Implicaciones en Adoptar la Agricultura Digital. Una Visión Genérica. *Ciencia Latina Revista Científica Multidisciplinar*, 8(2), 6818-6826. https://doi.org/https://doi.org/10.37811/cl_rcm.v8i2.11095
- Díaz Ramírez, Lizette Maritza, Bravo Parra, Aura Maria, Delgado Cuene, Ivan Rodrigo, Rosas Sandoval, Geovani, Gutierrez García, Gustavo Adolfo, Vanegas Cubillos, Martha Cristina, Lombo Ortiz, Darwin Fabian, Castro Nunez, Augusto Carlos, y Hurtado Bermudez, Jhon Jairo (2023). Estrategia para la sostenibilidad de la cadena de cacao en Caquetá: Ruta hacia la acción climática y la construcción de paz. *Centro Internacional de Agricultura Tropical (CIAT)*(565), 158. <https://doi.org/https://orcid.org/0000-0002-5005-4264>
- Ehsan, Aleem, Abuhaliqa, Mohammed A., Catal, Cagatay, y Mishra, Deepti (2022). RESTful API Testing Methodologies: Rationale, Challenges, and Solution Directions. *Applied Sciences*, 12(9). <https://doi.org/10.3390/app12094369>
- Esteso, Ana, Alemany, María del Mar Eva, Ortiz, Ángel, y Iannacone, Rina (2022). Crop planting and harvesting planning: Conceptual framework and sustainable multi-objective optimization for plants with variable molecule concentrations and minimum time between harvests. *Applied Mathematical Modelling*, 112, 136-155. <https://doi.org/https://doi.org/10.1016/j.apm.2022.07.023>
- Estrella Quispe, Alex Fabian, Segura Núñez, Germán Patricio, Martínez Rosas, Carlos Eduardo, Sánchez Chila, Angie Yesenia, y Bravo López, Estefanía Berenice (2023). Sistemas de Gestión de Indicadores Clave de Desempeño (KPIs) en procesos industriales. *Ciencia Latina Revista Científica Multidisciplinar*, 7(4), 125-141. https://doi.org/https://doi.org/10.37811/cl_rcm.v7i4.6853
- Falcón-Suárez (2023). Software para el apoyo a la toma de decisiones en el sector agrícola. *Revista Ingeniería Agrícola*, 13(3), 48-55. <https://doi.org/https://www.redalyc.org/articulo.oa?id=586275623008>
- Gómez Gómez, Jorge, Parra Urueta, Camilo Andrés, y Guzmán Pineda, Juan José (2025). Sistema de trazabilidad de cacao soportado en Internet de las cosas. *Revista*

- Colombiana de Tecnologías de Avanzada (RCTA)*, 1(45), 204-215.
<https://doi.org/https://doi.org/10.24054/rcta.v1i45.3283>
- Ibrahim, Asaad Khaleel (2021). Evolution of the Web: from Web 1.0 to 4.0. *Qubahan Academic Journal*, 1(3), 20–28. <https://doi.org/https://doi.org/10.48161/qaj.v1n3a75>
- Izquierdo Joshep (2025). Modelo integrado de adopción de tecnologías en la agricultura. Caso de estudio: IA e IoT aplicadas en producción de cacao. *Revista Espacios*, 46(3).
<https://doi.org/https://doi.org/10.48082/espacios-a25v46n03p38>
- Jain, Ronak, Shrivastava, Vishal, Pandey, Akhil, y Sharma, Aarti (2024). Modern Web Development using CSS & HTML. *International Journal of Emerging Science and Engineering (IJESE)*, 12(6), 13–16. [https://doi.org/OI 10.35940/ijese.G2574.12060524](https://doi.org/OI%2010.35940/ijese.G2574.12060524)
- Jayaraman, Kumaresan, y Kumar Avneesh. (. (2024). Optimizing Single Page Applications (SPA) Through Angular Framework Innovations. *International Journal of Research in Modern Engineering and Emerging Technology (IJRMEET)*, 12(12), 516–532.
<https://doi.org/10.63345/ijrmeet.org.v12.i12.24>
- John German Vera Luzuriaga (2023). Aplicación de los sensores IOT en la agroindustria: estudio taxonómico del modelo GMAAS según el procesamiento de datos. *Polo del Conocimiento*, 8(3), 100-126. <https://doi.org/10.23857/pc.v8i3>
- Karić, Anes, y Durmić, Nermina (2024). Comparison of JavaScript Frontend Frameworks – Angular, React, and Vue. *International Journal of Innovative Science and Research Technology (IJISRT)*, 9(6). <https://doi.org/10.38124/ijisrt/IJISRT24JUN600>
- Karlsson, Stefan, Jongeling, Robbert, Čaušević, Adnan, y Sundmark, Daniel (2024). Exploring behaviours of RESTful APIs in an industrial setting. *Software Quality Journal*, 32, 1287–1324. <https://doi.org/10.1007/s11219-024-09686-0>
- Khoddamzadeh, Amir Ali, y Souza Costa, Bárbara Nogueira (2023). Best Nitrogen Management Practices Using Sensor-Based Smart Agriculture in Nursery Production

of Cacao. *Horticulturae*, 9(4).
<https://doi.org/https://doi.org/10.3390/horticulturae9040454>

Kitole, Felician Andrew, Mkuna, Eliaza, y Sesabo, Jennifer Kasanda (2024). Digitalization and agricultural transformation in developing countries: Empirical evidence from Tanzania agriculture sector. *Smart Agricultural Technology*, 7.
<https://doi.org/https://doi.org/10.1016/j.atech.2023.100379>

Kothapalli, Mounika (2021). The Evolution of Component-Based Architecture in Front-End Development. *Journal of Scientific and Engineering Research (JSAER)*, 78(7), 261–264. <https://doi.org/https://jsaer.com/download/vol-8-iss-7-2021/JSAER2021-8-7-261-264.pdf>

Kothari (2004). *Research Methodology: Methods and Techniques*. New Age International Publishers.

Kothari (2004). *Research Methodology: Methods and Techniques*. New Age International Publishers Nueva Delhi.

Luis Felipe Vargas-Rojas., y. Victor Guerrero. (2020). arquitectura para el análisis de datos agronómicos en un ambiente de Big Data. *Investigación e Innovación en Ingenierías*, 8(2), 233-250. <https://doi.org/https://doi.org/10.17081/invinno.8.2.4211>

Lehtinen, Timo., Itkonen, Juha, y Lassenius, Casper (2017). Recurring opinions or productive improvements—what agile teams actually discuss in retrospectives. *Empirical Software Engineering*, 22(5), 2409-2452.
<https://doi.org/https://doi.org/10.1007/s10664-016-9464-2>

Luis Gabriel Santero Montalvo (2023). *El Cultivo de Cacao como Agente Transformador de los Territorios Rurales: Un Referente de la Innovación Socia*. Universidad Nacional Abierta y a Distancia UNAD.

- Mahnic, Viljan (2011). A Case Study on Agile Estimating and Planning Using Scrum. *Elektronika ir Elektrotechnika*, 111(5), 123-128. <https://doi.org/https://doi.org/10.5755/j01.eee.111.5.372>
- Majid, Umair (2018). *Research Fundamentals: Data Collection, Data Analysis, and Ethics*. Undergraduate Research in Natural and Clinical Science and Technology Journal, Vol. 2.
- Majid, Umair (2018). Research Fundamentals: Study Design, Population, and Sample Size. *Undergraduate Research in Natural and Clinical Science and Technology Journal*, 2(1), 1-7. <https://doi.org/https://doi.org/10.26685/urncst.16>
- Majid, Umair (2018). Research Fundamentals: Study Design, Population, and Sample Size. *Undergraduate Research in Natural and Clinical Science and Technology Journal*, 2(1), 1-7. <https://doi.org/https://doi.org/10.26685/urncst.16>
- Mayorga Vásquez, Luis Carlos, Riccardi Martillo, Gustavo Andrés, Bermeo Almeida, Oscar Xavier, y Guevara Arias, Verónica Isabel (2022). Sistema Web para los procesos administrativos y de producción en viveros del Cantón Milagro. *Ingeniería y sus Alcances. Revista de Investigación*, 6(16), 200-213. <https://doi.org/https://doi.org/10.33996/revistaingenieria.v6i16.100>
- Mayorga Vásquez, Luis Carlos, Riccardi Martillo, Gustavo Andrés, Bermeo Almeida, Oscar Xavier, y Guevara Arias, Verónica Isabel (2022). Sistema Web para los procesos administrativos y de producción en viveros del Cantón Milagro. *Ingeniería y sus Alcances. Revista de Investigación*, 6(16), 200-213. <https://doi.org/https://doi.org/10.33996/revistaingenieria.v6i16.100>
- Mendoza Fernández, Marco Vinicio (2022). *Desarrollar sistema web para la gestión administrativa de plantas de cacao en el Vivero Díaz en Zhucay*. Universidad Agraria del Ecuador.

- Mohajan, Haradhan Kumar (2018). Qualitative Research Methodology in Social Sciences and Related Subjects. *Journal of Economic Development, Environment and People*, 7(1), 23-48.
- Moń, Michał, y Pańczyk, Beata (2025). A comparative analysis of web application test automation tools. *Journal of Computer Sciences Institute (JCSI)*. <https://doi.org/10.35784/jcsi.7119>
- Mylsamy, Sekar, y Mehrotra, Tushar (2025). Continuous Integration and Continuous Delivery (CI/CD) Pipelines. *Journal of Quantum Science and Technology (JQST)*, 2(2), 495–504. <https://doi.org/10.63345/jqst.v2i2.284>
- Nama, Prathyusha (2022). Optimizing automation systems with AI: A study on enhancing workflow efficiency through intelligent decision-making algorithms. *World Journal of Advanced Engineering Technology and Sciences (WJAETS)*, 7(2), 296–307. <https://doi.org/10.30574/wjaets.2022.7.2.0118>
- Palomeque Solano, María Eugenia, y Urdaneta Montiel, Armando José (2022). Planificación y control de la gestión administrativa de la producción en el sector agrícola de la provincia de El Oro. *Polo del Conocimiento*, 7(6), 302-326. <https://doi.org/https://doi.org/10.23857/pc.v7i6.4075>
- Parlakkiliç, Alaattin (2021). Evaluating the effects of responsive design on the usability of academic websites in the pandemic. *Education and Information Technologies (Springer)*. <https://doi.org/10.1007/s10639-021-10650-9>
- Piastou, Mikita (2023). Comprehensive Performance and Scalability Assessment of Front-End Frameworks: React, Angular, and Vue.js. *World Journal of Advanced Engineering Technology and Sciences (WJAETS)*, 9(2), 366–376. <https://doi.org/10.30574/wjaets.2023.9.2.0153>
- Rahi, Samar (2017). Research Design and Methods: A Systematic Review of Research Paradigms, Sampling Issues and Instruments Development. *International Journal of Economics & Management Sciences*, 6(2).

- Rahi, Samar (2017). Research Design and Methods: A Systematic Review of Research Paradigms, Sampling Issues and Instruments Development. *International Journal of Economics & Management Sciences*, 6(2).
- Sánchez-Calle Jeison. Eli (2022). Algoritmos y su efecto en la agricultura: automatización de procesos. *Revista Científica de Sistemas e Informática*, 2(2). <https://doi.org/https://doi.org/10.51252/rcsi.v2i2.386>
- Savani, Nilesh (2023). The Future of Web Development: An In-depth Analysis of Micro-Frontend Approaches. *International Journal of Computer Trends and Technology (IJCTT)*, 71(11), 65–69. <https://doi.org/10.14445/22312803/IJCTT-V71I11P109>
- Schwaber Ken. & Jeff Sutherland (2020). *The Scrum Guide*. Scrum.org.
- Stray Viktoria Sjøberg, y Dybå, Tore (2016). The daily stand-up meeting: A grounded theory study. *Journal of Systems and Software*, 114, 101-124. <https://doi.org/https://doi.org/10.1016/j.jss.2016.01.004>
- Torres Chillogalli Maria (2023). *Aplicativo web para la gestión y control de producción de un vivero de cacao*. Universidad Agraria del Ecuador.
- Torres Chillogalli Maria (2023). *Aplicativo web para la gestión y control de producción de un vivero de cacao*. Universidad Agraria del Ecuador.
- Cristina Urbina-Céspedes, Mauricio Alonso Campos, Juan Bautista Nuñez (2026). Planificación Agrícola con Modelos Cuantitativos. *Revista Académica Arjé*, 9, 1-17. <https://doi.org/https://doi.org/10.23857/pc.v7i6.4075>
- Valecillos Barreto, Dinora Ibeth, y Sarmiento Gil, Francis Adil (2026). Sistema de indicadores para fortalecer la planificación operativa en los entes agrícolas del estado Trujillo. *Momboy*(26), 72-84. <https://doi.org/https://doi.org/10.70219/mby-262026-438>

- Vásquez-García Adela (2017). Desarrollo de una aplicación web para evaluar cultivos agrícolas a través del método de la MAP. *Revista mexicana de ciencias agrícolas*, 8(8), 1813-1825. <https://doi.org/10.29312/remexca.v8i8.704>
- Verwijns Christiaan. & Daniel Russo (2024). Do Agile Scaling Approaches Make A Difference? An Empirical Comparison of Team Effectiveness Across. *Popular Scaling Approaches. Empirical Software Engineering*, 29(75). <https://doi.org/10.1007/s10664-024-10481-5>
- Villa López (2023). *Sistema web de gestión de las actividades de producción en el cultivo de cacao para la finca cacaotera Álvarez El Triunfo*. Escuela Superior Politécnica del Litoral.
- Jessenia Marlene Calderon Vivanco, Ángel Daniel Analuisa Iza, Wilson Gustavo Chango Sailema, Pedro Stalyn Aguilar Encarnación (2024). Aplicación web progresiva para gestionar pronósticos agrícolas en las plantaciones de cacao en Orellana. *Mikarimin. Revista Científica Multidisciplinaria*, 10(2). <https://doi.org/10.61154/mrcm.v10i2.3211>
- Weichbroth, Paweł (2024). Usability Testing of Mobile Applications: A Methodological Framework. *Applied Sciences*, 14(5). <https://doi.org/10.3390/app14051792>
- Žáček, Miroslav, Hamplová, Adéla, Tyrychtr, Jan, y Vrana, Ivan (2025). Improvements for the Planning Process in the Scrum Method. *Applied Sciences*, 15(1). <https://doi.org/10.3390/app15010202>

ANEXOS

Anexo 1: Correo de Asignación de Tutor



Universidad Laica Eloy Alfaro de Manabí

Periodo 2025-2 - Notificación de **tutor asignado - SOFTWARE 2024 - NS - (EL CARMEN)**

Estimad@
Docente y Estudiante
Uleam

En cumplimiento de lo establecido en la Ley, el Reglamento de Régimen Académico y las disposiciones estatutarias de la Uleam, por medio de la presente se oficializa la dirección y **tutoría** en el desarrollo del Trabajo de Integración curricular / Trabajo de Titulación del siguiente estudiante:

Tema: SISTEMA WEB CON ARQUITECTURAS FRONTEND DINÁMICAS PARA LA PLANIFICACIÓN DE PRODUCCIÓN DEL VIVERO LASTENIA DE ULEAM EL CARMEN

Estado de aprobación: Aprobado

Tipo de titulación: Trabajo de Integración Curricular

Tipo de proyecto: Trabajo de Integración Curricular / Trabajo de titulación se articula con proyectos y programas de Investigación.

Apellidos y nombres del **tutor asignado:** MENDOZA VILLAMAR ROCIO ALEXANDRA

Apellidos y nombres del estudiante: VERDUGA CALDERON HEYDER JAVIER

Carrera: SOFTWARE 2024 - NS - (EL CARMEN)

Periodo de Inducción: Periodo 2025-2

Sírvase cumplir con lo dispuesto en el Manual de Procedimientos de TITULACIÓN DE ESTUDIANTES DE GRADO: TRABAJO DE INTEGRACIÓN CURRICULAR Y UNIDAD DE TITULACIÓN <https://departamentos.uleam.edu.ec/gestion-aseguramiento-calidad/files/2020/06/PAT-04-Titulacion-de-Estudiantes-de-grado-UIC-y-UT.pdf>

Particular que se informa para los fines consiguientes.

Atentamente,

Comisión Académica y Responsable de Titulación.

Anexo 2: Evidencia de Encuesta parte 1

1. Cuanto tiempo le toma calcular y registrar manualmente las fechas de un ciclo productivo completo para un lote

- ☐ Menos 10 min
- ☐ 10-30 min
- ☐ 30-60 min
- ☐ Mas 1 hora

2. Que tan complejo es reprogramar todas las actividades cuando ocurre un imprevisto

- ☐ Muy complejo
- ☐ Complejo
- ☐ Poco
- ☐ Nada complejo

3. Con que frecuencia detecta que una actividad se realizo fuera de la fecha optima por error de calculo

- ☐ 0-1 vez/mes
- ☐ 2-4 veces/mes
- ☐ Mas 5 veces/mes

4. Considera necesario automatizar el calculo de fechas y cronogramas

- ☐ Totalmente necesario
 - ☐ Necesario
 - ☐ Poco
 - ☐ Innecesario
-

Anexo 3: Evidencia de Encuesta parte 2

5. Que porcentaje de plantulas se pierde por falta de seguimiento oportuno del cronograma

- ☐ Menos 2%
- ☐ 2-5%
- ☐ 6-10%
- ☐ Mas 10%

6. La ausencia de registro historico de proveedores afecta la calidad de sus decisiones de recompra

- ☐ Si, significativamente
- ☐ Si, moderadamente
- ☐ No mucho
- ☐ No

7. Participaria en una capacitacion de 30 min para usar el nuevo sistema

- ☐ Si
- ☐ No
- ☐ Tal vez

8. Que temor principal tiene ante la digitalizacion de procesos

- ☐ Perdida de informacion
 - ☐ Fallas tecnicas
 - ☐ Sin conexion a internet
 - ☐ Ninguno
-

Anexo 4:Evidencia de Entrevista Semi-Estructurada

Preguntas

1. ¿Cuáles son las etapas obligatorias del ciclo desde la recogida de la tierra hasta el despacho final?
2. ¿Qué tiempo transcurre entre injerto y púa?
3. ¿Cómo determina las fechas para pasar de una fase a otra?
4. ¿Cómo reprograma las actividades ante condiciones climáticas y cómo afecta esto a las posteriores fases?
5. ¿Con qué antelación necesita las alertas de actividades?
6. ¿En qué momento y lugar físico revisaría su cronograma?
7. ¿Qué información necesita ver de un solo vistazo?
8. ¿Continuaría estrictamente las actividades del cronograma en el sistema o solicitaría recalcular ante una condición anómala?
9. ¿Existe algún registro de actividades y cómo se verifica su cumplimiento?
10. ¿Qué criterios utiliza para priorizar algún lote en caso de recursos limitados?

Anexo 5: Pruebas técnicas en el Viveo Lastenia



Anexo 6: Visita al vivero Lastenia



Anexo 7: Certificado de detección de IA Compilatio

**Certificado de análisis**
Compilatio Magister+ | ULEAM-ECU

TESIS HEYDER 3
ID : 4f134523a58957670cf77666f007da2d8bd6d501

**8%**
Textos sospechosos

Nombre del fichero : TESIS HEYDER 3.txt
Tamaño del archivo original : 4,05 MB
Número de palabras : 26.016

Depositante : ROCIO MENDOZA VILLAMAR
Fecha de depósito : 3 de agosto de 2025
Tipo de carga : interface
fecha de fin de análisis : 3 de agosto de 2025

 **Resumen** (sección 1/2)

Localización de los textos sospechosos en el documento :



Incluido en el porcentaje de textos sospechosos :

 **Similitudes** <1%

Pasajes con similitudes a fuentes encontradas en diferentes colecciones.



 **Detección de IA** 2%

Textos estilísticamente próximos a un texto generado por una IA.
Este índice es un indicador y no una prueba. Comprueba con el autor si domina los conocimientos mencionados en el documento.



 **Idiomas no reconocidos** 6%

Pasajes en los que parte del vocabulario utilizado no forma parte del diccionario de la lengua.
Puede tratarse de un intento del autor de modificar el texto para evitar ser detectado.



No incluido en el porcentaje de textos sospechosos :

 **Textos entre comillas** 1%

Pasajes entre comillas, a menudo indicativos de una cita.



GLOSARIO

Conceptos Tecnológicos y de Software:

- **API REST:** Interfaz que permite la comunicación estandarizada entre el frontend (la cara visual) y el backend (el servidor) mediante el intercambio de datos, facilitando que distintas tecnologías trabajen juntas.
- **Arquitectura Frontend Dinámica:** Estructura visual de una aplicación que permite interactuar con los datos de forma inmediata sin necesidad de recargar la página completa, brindando una experiencia de usuario fluida y reactiva.
- **Backlog del Producto:** Lista priorizada que contiene todos los requisitos, tareas y funcionalidades necesarias para desarrollar el sistema, sirviendo como la única fuente de trabajo para el equipo.
- **Capacitor:** Tecnología que permite empaquetar aplicaciones web para que funcionen como aplicaciones móviles nativas, permitiendo el acceso a funciones específicas del celular como las notificaciones.
- **Componentización:** Técnica de desarrollo que consiste en dividir la interfaz de usuario en piezas pequeñas y reutilizables, facilitando la creación de sistemas complejos y su posterior mantenimiento.
- **ERP (Enterprise Resource Planning):** Sistema de gestión integral que centraliza y automatiza los procesos operativos y administrativos de una organización en una sola plataforma de software.
- **Framework (Marco de Trabajo):** Conjunto de herramientas y reglas ya establecidas que ayudan a los programadores a desarrollar aplicaciones de forma más rápida, organizada y siguiendo estándares de calidad.

- **Historias de Usuario:** Descripciones breves y sencillas de una funcionalidad del sistema contadas desde la perspectiva del usuario final, utilizadas para entender qué necesita exactamente el cliente.
- **Laravel:** Framework de desarrollo en lenguaje PHP utilizado para construir el "backend" o cerebro del sistema, donde se gestionan la seguridad, la base de datos y las reglas del negocio.
- **MVVM (Model-View-ViewModel):** Patrón de diseño que separa la lógica de los datos de la interfaz visual, permitiendo que cualquier cambio en la información se refleje instantáneamente en la pantalla.
- **MySQL:** Sistema de gestión de bases de datos que permite almacenar y organizar toda la información del vivero, como los registros de lotes, metas y cronogramas, de forma segura y eficiente.
- **Nodo Móvil:** Componente del sistema diseñado específicamente para dispositivos móviles que permite a los técnicos trabajar directamente en el campo, incluso cuando no hay conexión a internet.
- **React:** Biblioteca de programación utilizada para construir interfaces de usuario interactivas basadas en componentes, permitiendo que la aplicación responda rápidamente a las acciones del usuario.
- **Responsive / Mobile-First:** Estrategia de diseño que asegura que la aplicación se adapte a cualquier tamaño de pantalla, priorizando siempre la facilidad de uso en teléfonos móviles para el trabajo de campo.
- **Scrum:** Metodología ágil que organiza el desarrollo de software en ciclos cortos de trabajo, permitiendo entregar avances funcionales cada pocas semanas y adaptarse a cambios constantes.

- **SPA (Single Page Application):** Tipo de aplicación web que carga una sola vez y actualiza el contenido dinámicamente, lo que evita esperas molestas por recargas de página y mejora la velocidad.
- **Sprint:** Periodo de tiempo fijo (en este proyecto de dos semanas) durante el cual el equipo de desarrollo se compromete a completar un conjunto específico de tareas para entregar un avance real.
- **SQLite:** Base de datos ligera que se guarda directamente en el dispositivo móvil, permitiendo que la aplicación funcione sin internet y guarde los datos para sincronizarlos después.
- **TailwindCSS:** Herramienta de diseño que permite aplicar estilos visuales de forma moderna y rápida, garantizando que la interfaz sea atractiva y fácil de leer bajo la luz solar.

Conceptos Agrícolas y Operativos

- **Capacidad de Lote:** Cantidad máxima de plántulas que pueden caber en un espacio físico determinado, calculada mediante fórmulas geométricas basadas en el tamaño de las fundas y los pasillos.
- **Ciclo Productivo (Fenológico):** Conjunto de etapas biológicas que atraviesa la planta de cacao desde que se siembra la semilla hasta que la plántula está lista para ser entregada al agricultor.
- **Despacho:** Fase final del proceso en el vivero que consiste en la entrega o salida física de las plántulas listas para su siembra definitiva en el campo.
- **Fases Técnicas:** Las cinco etapas obligatorias del proceso en el vivero Lastenia: preparación/siembra, germinación, injertación, desarrollo y despacho.
- **Lote de Producción:** Grupo específico de plántulas de cacao que comparten una misma fecha de inicio y ubicación, y que se gestionan bajo un cronograma de trabajo unificado.

- **Meta de Producción:** Objetivo específico de cantidad de plántulas que el vivero se propone producir en un tiempo determinado para satisfacer la demanda o proyectos académicos.
- **Plántula:** Planta joven de cacao en sus primeras etapas de desarrollo, cultivada en condiciones controladas dentro del vivero antes de ser trasladada a una plantación.
- **Reprogramación en Cascada:** Funcionalidad del sistema que, al cambiar la fecha de una fase por imprevistos climáticos, recalcula automáticamente las fechas de todas las fases siguientes del lote.
- **Trazabilidad:** Capacidad de seguir el historial completo de un lote, permitiendo saber en todo momento en qué fase está, qué cambios ha tenido y quién ha realizado las actividades