



PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR

Unidad Académica de Formación Técnica y Tecnológica – PUCE TEC

“Desarrollo de una Aplicación Web de Comercio Electrónico para La Floresta
Mercado Agroecológico, Fomentando la Participación Comunitaria y la Sostenibilidad
Local.”

Proyecto de titulación previo a la obtención del título de:
Tecnología Superior En Desarrollo De Software

Autores:

Marco Antonio Gómez Vivar

Pedro Andrés Tamayo Pastuña

Víctor Fernando Guerrero Intriago

Tutor:

Christian Roberto Tapia Gaibor

Quito, Ecuador

2025

Índice

Capítulo I: Levantamiento de Requisitos y Diseño del Sistema

1.1 Antecedentes

1.2 Metodología Scrum

1.3 Actores Principales

1.4 Historias de Usuario y Casos de Uso

1.5 Etapas del Proceso

1.5.1 Levantamiento de Información

1.5.2 Análisis del Problema y Formalización de la Solución

1.5.3 Diseño e Implementación del Sistema

1.5.4 Pruebas del Sistema

1.5.5 Casos de Uso

1.6 Creación del Backlog

1.7 Viabilidad del Proyecto

1.7.1 Viabilidad Técnica

1.7.2 Viabilidad Económica

1.7.3 Viabilidad Operativa

1.8 Planificación de los Sprints

1.9 Reuniones de los Sprints

1.10 Herramientas y Software

1.11 Hardware Requerido

Capítulo II: Construcción del Sistema

2.1 Alcance

2.2 Análisis de Diseño

2.2.1 Arquitectura de Software

2.3 Mapa de Navegación

2.4 Lógica de Negocio

2.5 Modelo de Base de Datos

2.5.1 Entidades Principales

2.5.2 Relaciones

2.5.3 Alcance Actual y Futuras Implementaciones

2.6 Requisitos Funcionales

2.7 Requisitos No Funcionales

2.7.1 Rendimiento

2.7.2 Escalabilidad

2.7.3 Seguridad

2.7.4 Usabilidad

2.7.5 Mantenibilidad

2.7.6 Disponibilidad

2.7.7 Compatibilidad

2.8 Diseño de la Arquitectura del Sistema

2.8.1 Componentes Principales

2.8.2 Flujo de la Arquitectura

2.8.3 Beneficios de la Arquitectura

2.9 Diagrama de Arquitectura del Sistema

2.9.1 Capas Principales de la Arquitectura

2.9.2 Flujo de Interacción del Sistema

2.9.3 Beneficios del Diseño Arquitectónico

Capítulo III: Pruebas y Estabilización del Sistema

3.1 Introducción

3.2 Tipos de Pruebas Realizadas

3.2.1 Pruebas Funcionales

3.2.2 Pruebas de Integración

3.2.3 Pruebas de Interfaz (UI/UX)

3.2.4 Pruebas de Rendimiento

3.2.5 Pruebas de Seguridad

3.3 Herramientas Utilizadas

3.4 Casos de Prueba

3.5 Estabilización del Sistema

3.6 Resultados de la Estabilización

Capítulo IV: Resultados y Evaluación del Sistema

4.1 Evaluación Frente a los Objetivos Planteados

4.2 Comparación Antes y Después de la Implementación

4.3 Validación por Parte de Usuarios y Productores

4.4 Indicadores de Éxito

4.5 Limitaciones Encontradas

Capítulo V: Conclusiones y Recomendaciones

5.1 Conclusiones Generales del Proyecto

5.2 Recomendaciones para Mejoras Futuras

5.3 Proyecciones a Futuro

DECLARACIÓN y AUTORIZACIÓN

Yo, Marco Antonio Gómez Vivar con C.I. 1721511861 autor del trabajo de Titulación intitulada: “Desarrollo de una Aplicación Web de Comercio Electrónico para La Floresta Mercado Agroecológico, Fomentando la Participación Comunitaria y la Sostenibilidad Local”, previa a la obtención del título de Tecnólogo Superior en Desarrollo de Software en la Unidad Académica de Formación Técnica y Tecnológica PUCE TEC:

1.- Declaro tener pleno conocimiento de la obligación que tiene la Pontificia Universidad Católica del Ecuador, de conformidad con el artículo 144 de la Ley Orgánica de Educación Superior, de entregar a la SENESCYT en formato digital una copia del referido trabajo de graduación para que sea integrado al Sistema Nacional de Información de la Educación Superior del Ecuador para su difusión pública respetando los derechos de autor.

2.- Autorizo a la Pontificia Universidad Católica del Ecuador a difundir a través de sitio web de la Biblioteca de la PUCE el referido trabajo de titulación, respetando las políticas de propiedad intelectual de Universidad.

Quito, agosto 2025



Marco Antonio Gómez Vivar

C.I. 1721511861

DECLARACIÓN y AUTORIZACIÓN

Yo, Pedro Andrés Tamayo Pastuña con C.I. 1726903436 autor del trabajo de Titulación intitulada: “Desarrollo de una Aplicación Web de Comercio Electrónico para La Floresta Mercado Agroecológico, Fomentando la Participación Comunitaria y la Sostenibilidad Local”, previa a la obtención del título de Tecnólogo Superior en Desarrollo de Software en la Unidad Académica de Formación Técnica y Tecnológica PUCE TEC:

1.- Declaro tener pleno conocimiento de la obligación que tiene la Pontificia Universidad Católica del Ecuador, de conformidad con el artículo 144 de la Ley Orgánica de Educación Superior, de entregar a la SENESCYT en formato digital una copia del referido trabajo de graduación para que sea integrado al Sistema Nacional de Información de la Educación Superior del Ecuador para su difusión pública respetando los derechos de autor.

2.- Autorizo a la Pontificia Universidad Católica del Ecuador a difundir a través de sitio web de la Biblioteca de la PUCE el referido trabajo de titulación, respetando las políticas de propiedad intelectual de Universidad.

Quito, agosto de 2025



Pedro Andrés Tamayo Pastuña

C.I. 1726903436

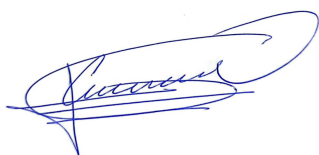
DECLARACIÓN y AUTORIZACIÓN

Yo, Víctor Fernando Guerrero Intriago con C.I. 1718138173 autor del trabajo de Titulación intitulada: “Desarrollo de una Aplicación Web de Comercio Electrónico para La Floresta Mercado Agroecológico, Fomentando la Participación Comunitaria y la Sostenibilidad Local”, previa a la obtención del título de Tecnólogo Superior en Desarrollo de Software en la Unidad Académica de Formación Técnica y Tecnológica PUCE TEC:

1.- Declaro tener pleno conocimiento de la obligación que tiene la Pontificia Universidad Católica del Ecuador, de conformidad con el artículo 144 de la Ley Orgánica de Educación Superior, de entregar a la SENESCYT en formato digital una copia del referido trabajo de graduación para que sea integrado al Sistema Nacional de Información de la Educación Superior del Ecuador para su difusión pública respetando los derechos de autor.

2.- Autorizo a la Pontificia Universidad Católica del Ecuador a difundir a través de sitio web de la Biblioteca de la PUCE el referido trabajo de titulación, respetando las políticas de propiedad intelectual de Universidad.

Quito, agosto de 2025



Víctor Fernando Guerrero Intriago

C.I. 1718138173

Agradecimientos Marco Gómez

Quiero expresar mi más sincero agradecimiento a mis hijas, Amelie y Giulia , por su amor y alegría, que me inspiran a seguir adelante cada día. A mi esposa, por su apoyo incondicional y su comprensión en cada paso de este camino. Y a mi madre, Luz Beatriz Vivar, por ser mi pilar y guía, siempre alentándome a alcanzar mis sueños. Su amor y apoyo han sido fundamentales en este logro. ¡Gracias a todos!

Agradecimientos Pedro Tamayo

A Dios, por darme la vida, la fuerza y la sabiduría para seguir adelante, incluso cuando el camino fue difícil.

A mis padres, por su amor, esfuerzo y apoyo incondicional. Por enseñarme que con humildad, trabajo y fe se pueden lograr los sueños.

Agradecimientos Fernando Guerrero

Quiero comenzar agradeciendo a Dios por haberme dado la fuerza y la sabiduría para no rendirme en este camino. A mi madre Luisa Intriago, que ha sido mi pilar, mi guía y mi mayor apoyo en todo momento. También agradezco profundamente a la Pontificia Universidad Católica del Ecuador por brindarme la oportunidad de formarme no solo como profesional, sino también como ser humano, con valores y principios que me inspiran a seguir creciendo.

Este logro representa un paso muy importante en mi vida, y deseo que mi esfuerzo sirva de ejemplo para otras personas que también sueñan con superarse. En especial, este proyecto va dedicado a mis hijos Jhosue y Luciana Guerrero, porque todo lo que hago es por y para ellos. Quiero que vean en mí a alguien que lucha, que no se detiene y que demuestra que sí se puede. Espero ser una inspiración en sus vidas, así como ellos lo son en la mía.

Introducción

En la actualidad, el comercio electrónico se ha consolidado como un recurso esencial para ampliar el alcance de productos y servicios, ofreciendo a productores y emprendedores la posibilidad de llegar a un público más amplio y diverso. No obstante, muchas iniciativas comunitarias aún enfrentan dificultades para acceder a herramientas tecnológicas que les permitan fortalecer su presencia digital y difundir sus productos de manera efectiva.

La Feria Agroecológica La Floresta, ubicada en Quito, Ecuador, es un espacio que promueve el consumo responsable y sostenible a través de la oferta de alimentos agroecológicos. A pesar de su relevancia para la comunidad, su visibilidad se ha visto limitada debido a la dependencia de canales presenciales y de redes sociales dispersas, lo que restringe el contacto con potenciales consumidores fuera del horario y lugar de la feria.

Con el propósito de responder a esta necesidad, se plantea el desarrollo de una aplicación web de catálogo virtual informativo que permita centralizar la exhibición de productos, publicar artículos educativos y fomentar la participación comunitaria. Este sistema no contempla procesos de pago en línea, sino que se enfoca en brindar información clara, atractiva y accesible para fortalecer el vínculo entre productores y consumidores.

La solución propuesta se implementa utilizando tecnologías de código abierto como Node.js, Express.js, React.js y PostgreSQL, organizadas bajo una arquitectura modular que facilita la escalabilidad, la seguridad y el mantenimiento. Además, se adopta la metodología ágil Scrum para gestionar el desarrollo de forma iterativa, permitiendo avances incrementales, retroalimentación continua y la participación activa de los involucrados.

Este proyecto no solo busca optimizar la difusión de los productos y contenidos de la feria, sino también generar un impacto positivo en la comunidad mediante la promoción de hábitos de consumo saludables y la valorización de la producción agroecológica local.

Asimismo, establece una base sólida para la incorporación de futuras mejoras que amplíen sus funcionalidades y beneficios a nivel regional.

CAPÍTULO I

LEVANTAMIENTO DE REQUISITOS Y DISEÑO DEL SISTEMA

1.1 Antecedentes

La Feria Agroecológica La Floresta, ubicada en Quito, Ecuador, promueve el consumo responsable y sostenible de alimentos agroecológicos. Sin embargo, enfrenta dificultades para visibilizar sus productos y educar a los consumidores sobre sus beneficios. Actualmente, los procesos de difusión son manuales y dependen de redes sociales dispersas.

Ante esta necesidad, se propone el desarrollo de una aplicación web de catálogo virtual informativo, cuyo propósito es:

Permitir que los productores publiquen sus productos con imágenes y descripciones.

Difundir historias y artículos sobre los beneficios de los alimentos agroecológicos.

Fomentar la participación comunitaria y la educación sobre hábitos de consumo saludable.

Este sistema no contempla procesos de pago, gestión de pedidos ni pasarela de pagos, ya que no es una plataforma transaccional.

1.2 Metodología Scrum

La gestión del proyecto se organizó utilizando la metodología ágil Scrum, adaptada a las necesidades y características del desarrollo. El trabajo se estructuró en sprints con una duración fija de 14 días, lo que permitió planificar, implementar y evaluar avances de forma continua, manteniendo un control preciso sobre el progreso del sistema.

Roles definidos para el proyecto:

Scrum Master: Tapia Gaibor Christian Roberto. Encargado de asegurar la correcta aplicación de Scrum, coordinar las reuniones, eliminar obstáculos que afectaran el avance y supervisar el cumplimiento de los plazos establecidos.

Product Owner: Administración de la Feria Agroecológica La Floresta. Responsable de establecer las prioridades del backlog, definir los objetivos de cada sprint y aprobar las entregas parciales, garantizando que el producto final respondiera a las necesidades reales de la feria.

Parámetros operativos aplicados:

Duración de los sprints: 14 días, iniciando los lunes y finalizando los domingos de la segunda semana.

Planificación de sprint: Realizada al comienzo de cada ciclo, definiendo las historias de usuario y tareas a desarrollar.

Reuniones diarias (Daily Scrum): Sesiones virtuales de 15 minutos a las 20:00 horas mediante Microsoft Teams, donde cada integrante informaba sus avances, tareas pendientes y posibles impedimentos.

Revisión de sprint: Presentación de los avances al Scrum Master y Product Owner al finalizar cada sprint para su aprobación.

Retrospectiva de sprint: Reunión de evaluación en la que se identificaban fortalezas, aspectos por mejorar y ajustes para el siguiente ciclo.

La aplicación de estos parámetros permitió mantener una comunicación constante, optimizar la coordinación del equipo y garantizar que cada entrega cumpliera con los objetivos funcionales y de calidad establecidos para el proyecto.

Tabla 1

Comparación de metodologías

Característica	Scrum	Cascada	Kanban	XP (Programación Extrema)
-----------------------	--------------	----------------	---------------	--

Entregas parciales	Sí	No	Sí	Sí
Adaptación a cambios	Alta	Baja	Media	Alta
Definición de roles	Clara	Sí	No	Clara
Duración de ciclos	2 semanas	Todo el proyecto	Flujo continuo	1-2 semanas
Reuniones de control	Diarias	Limitadas	Opcionales	Constantes
Enfoque en el cliente	Alto	Medio	Medio	Alto
Aseguramiento de calidad	Durante cada sprint	Al final	En el flujo	Pruebas permanentes
Colaboración del equipo	Alta	Baja	Media	Alta

1.3 Actores principales

Frontend: Andrés Pedro Tamayo – encargado de la creación de componentes, páginas y la estructuración de rutas, utilizando librerías como Bootstrap y CSS para un diseño atractivo y responsivo.

Backend: Fernando Guerrero – responsable del desarrollo de APIS , la conexión a la base de datos y la implementación de validaciones de seguridad utilizando librerías como JWT y Bycrypt.

Full Stack: Marco Antonio Gómez Vivar – encargado de integrar las interfaces frontend y backend, probar APIs con Postman y realizar un seguimiento general del proyecto aplicando las prácticas Scrum.

1.4 Historias de Usuario y Casos de Uso

El análisis de las historias de usuario permitió definir los requerimientos y priorizar las funcionalidades a implementar. Esto ayudó a entender qué necesitaba cada tipo de usuario y a organizar las tareas para el desarrollo de la aplicación.

Tabla 2

Historias de usuario

ID	Historias de Usuario	Rol	Prioridad	Estimación
US01	Como usuario, quiero registrarme en la plataforma, para poder hacer pedidos.	Usuario	Alta	3
US02	Como usuario, quiero iniciar sesión de forma segura, para acceder a mis productos y historias.	Usuario	Alta	3
US03	Como usuario, quiero ver una lista de productos disponibles, para conocer lo que ofrece la feria.	Usuario	Alta	5
US04	Como usuario, quiero ver detalles de un producto, para tomar una mejor decisión de	Usuario	Alta	3

	compra.			
US05	Como usuario, quiero agregar productos a un carrito, para organizar mis compras.	Usuario	Alta	5
US06	Como usuario, quiero realizar pedidos desde el carrito, para comprar productos.	Usuario	Alta	5
US07	Como productor, quiero agregar nuevos productos, para vender en la plataforma.	Productor	Alta	4
US08	Como productor, quiero editar y eliminar productos, para mantener actualizado mi catálogo.	Productor	Alta	4
US11	Como Productor, quiero publicar artículos en el blog, para informar a los usuarios.	Productor	Media	3
US12	Como visitante, quiero ver un botón de WhatsApp, para contactar fácilmente con los productores.	Visitante	Media	2
US13	Como usuario, quiero leer artículos informativos, para conocer los beneficios de los	Usuario	Media	3

	productos agroecológicos.			
US14	Como productor, quiero revisar la web con mis productos cargados, para verificar su correcta visualización.	Producto	Media	2
US15	Como usuario, quiero registrar fecha y hora de entrega, para coordinar mi pedido.	Usuario	Alta	3

1.5 Etapas del proceso

Levantamiento de información: Entrevistas con productores para identificar necesidades.

Análisis del problema: Falta de visibilidad digital y herramientas para la difusión de productos.

Diseño e implementación: Definición de módulos de catálogo, artículos y publicación de historias.

Pruebas: Validación funcional del sistema.

1.5.1 Levantamiento de información

En la etapa inicial del proyecto se realizó un levantamiento de datos directamente con los productores de la Feria Agroecológica La Floresta, con el objetivo de conocer su oferta y necesidades.

Para ello, se diseñó y aplicó una encuesta estructurada en formato físico a cinco (5) productores, la cual recopiló información relevante como el nombre del producto,

presentación, precio y observaciones. Este instrumento permitió contar con datos claros para la futura elaboración del catálogo virtual.

Además, se sostuvieron entrevistas presenciales de tipo semiestructurado, que facilitaron profundizar en aspectos específicos relacionados con la difusión de los productos y el uso de herramientas tecnológicas.

Del análisis de la información obtenida se identificaron requerimientos clave:

Disponer de una plataforma digital propia para promocionar los productos.

Posibilidad de que los productores editen y actualicen la información publicada.

Incorporar contenidos que sensibilicen al público sobre los beneficios de consumir alimentos agroecológicos.

Los formatos completos de las encuestas y los registros detallados se presentan en el Anexo 1.

1.5.2 Análisis del problema y formalización de la solución

Con base en la información obtenida, se determinó que el principal problema era la escasa visibilidad de los productos agroecológicos en entornos digitales y la falta de herramientas de autogestión para los productores.

Como solución, se definió el desarrollo de una aplicación web de catálogo virtual informativo, que permita a cada productor:

Crear, editar y administrar su propio catálogo de productos.

Publicar información adicional como historias y artículos educativos.

Ofrecer a los usuarios un espacio centralizado para conocer los beneficios de los productos y obtener información de contacto de los productores.

1.5.3 Diseño e implementación del sistema

El diseño funcional del sistema contempló los siguientes módulos:

Registro e inicio de sesión de usuarios y productores.

Gestión de productos: Creación, edición y eliminación de productos.

Publicación de artículos informativos: Sección para difundir contenido educativo sobre alimentación saludable.

Aprobación de productos por administrador.

Visualización de catálogo: Interfaz amigable para usuarios y visitantes.

El desarrollo técnico se distribuyó de la siguiente manera:

Backend: Implementado con Node.js y Express, utilizando controladores, middleware de autenticación y Prisma ORM para gestionar la base de datos PostgreSQL.

Frontend: Desarrollado en React.js con componentes reutilizables, empleando librerías como Bootstrap y React Icons para una interfaz clara y responsiva.

Control de versiones: Se utilizó GitHub con la estrategia Git Flow para organizar ramas y mantener control sobre el desarrollo.

Gestión de tareas: Jira fue la herramienta seleccionada para planificar y dar seguimiento a los sprints.

La aplicación fue estructurada siguiendo un enfoque modular para garantizar escalabilidad y fácil mantenimiento.

1.5.4 Pruebas del sistema

Las pruebas se ejecutaron de manera iterativa durante cada sprint, asegurando la calidad de las funcionalidades.

Se llevaron a cabo:

Pruebas de API (Postman): Validación de los endpoints del backend.

Pruebas funcionales: Verificación de los módulos de registro, inicio de sesión, gestión de productos y publicación de artículos.

Pruebas de interfaz (UI/UX): Evaluación de la experiencia de usuario y diseño responsivo.

Pruebas de seguridad: Validación de autenticación mediante JWT y protección de rutas.

1.5.5 Casos de uso

El desarrollo de los casos de uso facilitó la comprensión de cómo interactúan los usuarios con el sistema. Gracias a ellos se diseñaron las funciones clave de cada módulo, simplificando el desarrollo y las pruebas.

Tabla 3

Código	Nombre del Caso de Uso	Actor/es	Descripción clara y breve
CU01	Registro de usuarios	Usuario	El usuario crea su cuenta para usar la plataforma.
CU02	Inicio de sesión	Usuario, Productor, Admin	El sistema permite ingresar con usuario y clave.
CU03	Ver catálogo	Usuario, Visitante	Muestra los productos disponibles en la feria.
CU04	Ver detalle de producto	Usuario	Muestra más información de un producto.
CU05	Agregar al carrito	Usuario	Permite seleccionar productos para crear lista.
CU06	Hacer pedido	Usuario	Confirma la compra y

			envía el pedido por WhatsApp.
CU07	Gestionar productos	Productor	Permite subir, editar o eliminar sus productos.
CU10	Publicar en el blog	Administrador	Permite subir artículos informativos.
CU12	Revisar productos cargados	Productor	Verifica cómo se ven sus productos en la web.

Casos de uso

1.6 Creación del backlog

La creación del backlog permitió ordenar y planificar las funcionalidades del sistema según su importancia. Este listado sirvió como guía para dividir el trabajo en sprints y asegurar entregas parciales del producto.

Tabla 4

Historias de usuario

ID	Historia de Usuario	Rol	Prioridad	Estimación	Sprint sugerido	Estado
US01	Como usuario, quiero registrarme en la plataforma, para poder hacer	Usuario	Alta	3	Sprint 1 – Registro e Inicio de sesión	Terminado

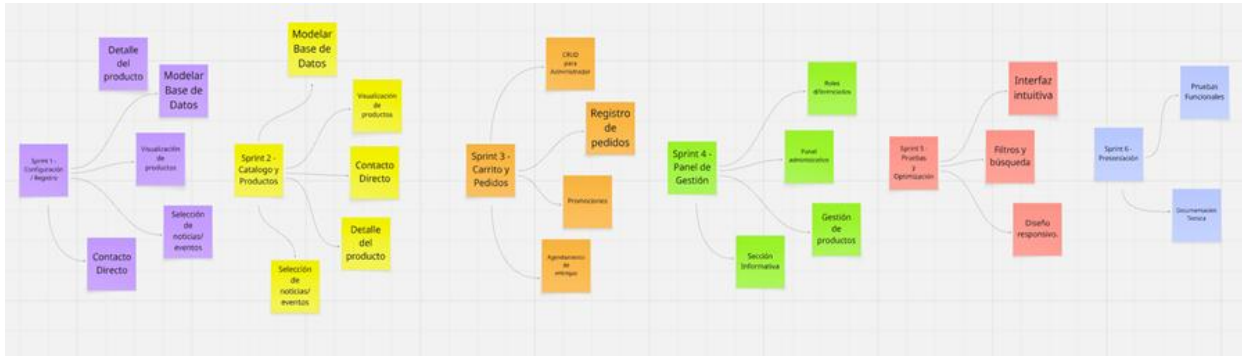
	pedidos.					
US02	Como usuario, quiero iniciar sesión de forma segura, para acceder a mis datos y pedidos.	Usuario	Alta	3	Sprint 1 – Registro e Inicio de sesión	Terminado
US03	Como usuario, quiero ver una lista de productos disponibles.	Usuario	Alta	5	Sprint 2 – Catálogo y Productos	Terminado
US04	Como usuario, quiero ver detalles de un producto.	Usuario	Alta	3	Sprint 2 – Catálogo y Productos	Terminado
US05	Como usuario, quiero agregar productos a un carrito.	Usuario	Alta	5	Sprint 3 – Carrito y Pedidos	Terminado
US06	Como usuario, quiero realizar pedidos desde el carrito.	Usuario	Alta	5	Sprint 3 – Carrito y Pedidos	Terminado
US07	Como productor,	Productor	Alta	4	Sprint 4 –	Terminado

	quiero agregar nuevos productos.				Panel de Gestión	
US08	Como productor, quiero editar y eliminar productos.	Productor	Alta	4	Sprint 4 – Panel de Gestión	Terminado
US11	Como administrador, quiero publicar artículos en el blog.	Administrador	Media	3	Sprint 4 – Panel de Gestión	Terminado
US12	Como visitante, quiero ver un botón de WhatsApp.	Visitante	Media	2	Sprint 4 – Panel de Gestión	Terminado
US13	Como usuario, quiero leer artículos informativos.	Usuario	Media	3	Sprint 4 – Panel de Gestión	Terminado
US14	Como productor, quiero revisar la web con mis productos	Productor	Media	2	Sprint 5 – Pruebas y Optimización	Terminado

	cargados.					
--	------------------	--	--	--	--	--

Imágen 1

Flujo de las historias



Nota: La imagen fue creada en la plataforma Miro.

1.7 Viabilidad del Proyecto

La viabilidad del proyecto se evaluó considerando los aspectos técnicos, económicos y operativos, asegurando que el desarrollo de la aplicación web de catálogo virtual sea factible y sostenible.

1.7.1 Viabilidad Técnica

El proyecto es técnicamente viable gracias a:

Uso de tecnologías de código abierto: Se emplearán herramientas como Node.js, Express.js, React.js y PostgreSQL, las cuales son gratuitas, escalables y ampliamente documentadas.

Arquitectura modular: Facilita el mantenimiento, escalabilidad e incorporación de nuevas funcionalidades como filtros avanzados o módulos informativos adicionales.

Compatibilidad multiplataforma: La aplicación será accesible desde navegadores web en computadoras y dispositivos móviles gracias a un diseño responsivo.

1.7.2 Viabilidad Económica

El proyecto es económicamente viable debido a:

Uso de software libre: No hay costos por licencias.

Equipo de desarrollo interno: El proyecto es realizado por los miembros del equipo sin costos adicionales de contratación externa.

Resultado: Se minimizan los gastos y se maximiza la eficiencia de los recursos disponibles.

1.7.3 Viabilidad Operativa

El sistema es operativamente viable porque:

Facilidad de uso: La plataforma está diseñada con una interfaz intuitiva para usuarios con conocimientos técnicos básicos.

Autogestión de productores: Los productores pueden publicar y administrar sus productos sin depender de terceros.

Soporte a largo plazo: La estructura modular permite realizar mantenimiento y agregar mejoras sin afectar el funcionamiento general.

Enfoque informativo y comunitario: La herramienta fomenta la participación de los productores y usuarios sin necesidad de procesos complejos de compra.

1.8 Planificación de los Sprints

La planificación de los sprints se realizó considerando las prioridades del Product Backlog y el tiempo estimado de desarrollo por cada integrante del equipo. Cada sprint abarca un conjunto específico de funcionalidades, asegurando entregables incrementales y validados en cada iteración.

A continuación, se presenta la planificación general de los sprints:

Tabal 4

Planificación de sprints.

Sprint	Fecha de inicio	Fecha de finalización	Duración (días)
Sprint 1 – Configuración y Registro	2025/05/05	2025/05/18	14
Sprint 2 – Catálogo y Productos	2025/05/19	2025/06/01	14
Sprint 3 – Carrito y Pedidos	2025/06/02	2025/06/15	14
Sprint 4 – Panel de Gestión	2025/06/16	2025/07/29	14
Sprint 5 – Pruebas y Optimización	2025/06/30	2025/07/13	14
Sprint 6 – Documentación y Presentación	2025/07/14	2025/07/27	14

Notas:

Revisión de Sprint: Todos los viernes.

Retrospectiva: Al finalizar cada Sprint.

Definition of Done (DoD): Cada entregable cumple con los criterios de aceptación antes de darse por finalizado.

Presentación final del proyecto: 2025/08/08.

1.9 Reuniones de los Sprints

Para asegurar la correcta coordinación del equipo y la entrega oportuna de los módulos, se estableció un esquema de reuniones:

Reunión diaria (Daily Scrum): Se lleva a cabo cada día a las 20:00 horas mediante Microsoft Teams. Durante estas sesiones de 15 minutos, cada integrante informa sobre las tareas completadas, las pendientes y los posibles bloqueos.

Revisión de Sprint (Sprint Review): Al final de cada sprint, se realiza una reunión para presentar los avances al tutor y recoger retroalimentación.

Retrospectiva de Sprint: Se analizan los aspectos positivos y negativos del sprint concluido, identificando oportunidades de mejora para las próximas iteraciones.

Este enfoque garantiza una comunicación continua y una mejora progresiva del proceso de desarrollo.

1.9.1 Definition of Done (DoD)

En el proyecto, el Definition of Done se estableció como el conjunto de criterios que toda funcionalidad debía cumplir antes de ser considerada finalizada. Estos criterios, definidos en conjunto por el equipo y el tutor (Scrum Master), fueron:

Cumplimiento del 100 % de los criterios de aceptación definidos en la historia de usuario correspondiente.

Aprobación del código mediante revisión por pares (code review) en GitHub.

Integración correcta en la rama principal sin conflictos, siguiendo el flujo GitFlow.

Ejecución y aprobación de las pruebas funcionales y de seguridad asociadas.

Validación por parte del tutor en la reunión de revisión de sprint.

Documentación técnica y de usuario actualizada en el repositorio.

Este estándar permitió mantener la calidad de cada entrega y asegurar que todas las funcionalidades estuvieran listas para su uso antes de considerarse completadas.

1.10 Herramientas y Software

El desarrollo del proyecto requiere el uso de diversas herramientas y tecnologías para la gestión, programación y despliegue:

Gestión de proyecto: Jira

Control de versiones: Git (flujo GitFlow)

Backend: Node.js, Express.js, Prisma ORM

Base de datos: PostgreSQL, PgAdmin

Frontend: JavaScript, CSS, Bootstrap

Entorno de desarrollo: Visual Studio Code

Otros: Microsoft Teams (comunicación), Postman (pruebas de API)

Estas herramientas fueron seleccionadas por su versatilidad, disponibilidad y compatibilidad con proyectos de tipo SaaS.

1.10.1 Flujo de trabajo en Git

Para la gestión del control de versiones se utilizó GitHub aplicando la estrategia GitFlow, lo que permitió organizar el trabajo del equipo y evitar conflictos en el código. El flujo aplicado fue el siguiente:

Rama principal (main): Contení la versión estable lista para producción.

Rama de desarrollo (develop): Integraba las funcionalidades completadas antes de pasar a producción.

Ramas de funcionalidad (feature): Cada nueva característica se desarrolló en una rama independiente creada a partir de develop, con el formato feature/nombre-funcionalidad.

Ramas de corrección (hotfix): Usadas para solucionar errores críticos detectados en producción.

Ramas de liberación (release): Preparaban la versión final, con pruebas y ajustes previos a su integración en main.

Integración: Todas las ramas pasaban por revisión de código (pull request) antes de fusionarse.

Este flujo permitió que varios desarrolladores trabajaran en paralelo sin afectar la estabilidad del producto final.

1.11 Hardware Requerido

Para el desarrollo y las pruebas del sistema, se definieron los siguientes requisitos mínimos de hardware:

Procesador: Intel Core o superior.

Memoria RAM: 2 GB (preferible 4 GB para un rendimiento óptimo).

Almacenamiento: 120 GB de disco duro.

Sistema operativo: Ubuntu 20.04 LTS o superior.

Conexión a Internet estable para el uso de herramientas colaborativas.

Estos requerimientos son accesibles para los miembros del equipo y compatibles con las tecnologías elegidas.

CAPÍTULO II

CONSTRUCCIÓN DEL SISTEMA

2.1 Alcance

El sistema desarrollado tiene como objetivo principal proporcionar una plataforma web que funcione como catálogo virtual informativo para la Feria Agroecológica La Floresta.

Objetivos específicos del alcance:

Permitir que los productores publiquen, editen y eliminen sus productos.

Mostrar un catálogo público con información completa de cada producto, incluyendo imágenes, descripciones y beneficios.

Habilitar un módulo de artículos informativos para fomentar la educación alimentaria.

Incorporar un sistema de aprobación de productos gestionado por el administrador.

Facilitar el contacto entre usuarios o visitantes y productores a través de la plataforma.

2.2 Análisis de Diseño

El análisis de diseño definió la estructura y componentes principales del sistema para garantizar su modularidad, escalabilidad y facilidad de mantenimiento.

2.2.1 Arquitectura de software

Frontend: Desarrollado en React.js para ofrecer una interfaz de usuario dinámica y responsiva.

Backend: Implementado con Node.js y Express.js, con controladores y servicios organizados según el patrón MVC.

Base de datos: PostgreSQL administrado mediante Prisma ORM para garantizar consultas seguras y escalables.

API REST: Comunicación entre frontend y backend basada en endpoints seguros.

Contenedores: Uso de Docker para encapsular los servicios y facilitar a futuro el despliegue.

Imagen 2

Arquitectura de Software



Nota: Imagen creada en la plataforma Canva.

2.3 Mapa de Navegación

El sistema fue diseñado con una navegación simple e intuitiva:

Inicio: Página principal con información general de la feria.

Catálogo de productos: Visualización de productos con opción de buscar por productos.

Detalle de producto: Información completa del producto, beneficios e información de contacto del productor.

Blog informativo: Artículos educativos sobre alimentación saludable y prácticas agroecológicas.

Panel de productor: Área privada para gestionar productos.

Panel de administrador: Área privada para aprobar productos, gestionar usuarios y publicar artículos.

Inicio de sesión: Pantalla de autenticación para productores .

2.4 Lógica de Negocio

Actualmente, la Feria Agroecológica La Floresta se desarrolla de manera presencial los días jueves, en un horario de 08:00h a 16:00h. Sin embargo, los procesos de difusión de productos y gestión de información se realizan de forma manual, lo que limita la capacidad de los productores para visibilizar su oferta y llegar a un público más amplio durante toda la semana.

La propuesta del sistema busca digitalizar y optimizar estos procesos mediante una plataforma web de catálogo virtual, que permitirá a los productores publicar información de sus productos y a los usuarios acceder a contenidos informativos en cualquier momento, sin depender del espacio físico de la feria.

Procesos principales de la lógica de negocio

Gestión de Inventario Digital (Tenant – Product):

Cada productor (Tenant) podrá registrar sus productos (Product) con información detallada: nombre, descripción, beneficios e imágenes.

Los productores podrán editar o eliminar productos según sea necesario.

Este módulo reemplaza el control manual y mejora la visibilidad de los productos, sin incluir precios ni procesos de venta.

Administración de Usuarios y Roles (Tenant – User):

Se podrán asignar roles específicos según las responsabilidades: publicar productos, editar información o aprobar contenido (en el caso de los administradores).

Este mecanismo asegura una gestión ordenada sin funciones de ventas o pagos.

Difusión de Productos y Contenido Informativo (BlogPost – Category – Tag):

El sistema contará con un módulo de publicaciones (BlogPost) para compartir artículos educativos, recetas y noticias sobre la feria.

Este componente contribuye al posicionamiento digital de la feria y fomenta la educación alimentaria.

Beneficios de la lógica de negocio

Permite a los productores aumentar su visibilidad.

Optimiza los recursos y extiende el alcance de la feria más allá del espacio físico y el horario tradicional.

Brinda a los usuarios una plataforma clara y accesible para informarse sobre productos agroecológicos y contenidos educativos.

2.5 Modelo de Base de Datos

El modelo de datos del sistema se construyó utilizando Prisma ORM sobre una base de datos relacional en PostgreSQL. Se diseñó bajo una arquitectura multitenant, lo que permite que múltiples productores trabajen de forma independiente sin interferir con la información de otros.

Este modelo contempla tanto las entidades requeridas para el catálogo virtual informativo como aquellas que se han previsto para futuras implementaciones.

2.5.1 Entidades principales

Tenant (Productor)

Representa a cada productor registrado en la plataforma.

Atributos:

id: Identificador único del productor.

name: Nombre del productor.

createdAt: Fecha de creación del registro.

User (Usuario)

Representa a los usuarios vinculados a un productor (administradores y colaboradores).

Atributos:

id: Identificador único.

tenantId: Relación con el productor.

name: Nombre del usuario.

email: Correo electrónico único.

phone: Número de teléfono.

password: Contraseña cifrada.

role: Rol del usuario (administrador, productor).

createdAt: Fecha de creación del registro.

Product (Producto)

Representa los productos publicados en el catálogo informativo.

Atributos:

id: Identificador único.

tenantId: Relación con el productor.

userId: Usuario que creó el producto.

name: Nombre del producto.

description: Descripción detallada.

photoUrl: Imagen del producto.

contactNumber: Teléfono de contacto.

quantityAvailable: Cantidad disponible.

status: Estado de aprobación.

createdAt: Fecha de creación.

BlogPost (Publicación)

Artículos informativos relacionados con productos y prácticas agroecológicas.

Atributos:

id: Identificador único.

tenantId: Relación con el productor.

userId: Usuario que creó la publicación.

title: Título del artículo.

description: Contenido.

imageUrl: Imagen asociada.

createdAt: Fecha de publicación.

Category (Categoría)

Clasifica productos y artículos.

Atributos:

id: Identificador único.

name: Nombre de la categoría.

Tag (Etiqueta)

Etiquetas asociadas a artículos para mejorar su clasificación.

Atributos:

id: Identificador único.

name: Nombre de la etiqueta.

PostCategories

Relación muchos a muchos entre publicaciones y categorías.

Atributos:

p: ID de la publicación.

i: ID de la categoría.

PostTags

Relación muchos a muchos entre publicaciones y etiquetas.

Atributos:

p: ID de la publicación.

i: ID de la etiqueta.

Customer (Futuro)

Tabla reservada para futuras funcionalidades de gestión de clientes.

Atributos:

id: Identificador único.

tenantId: Relación con el productor.

name: Nombre.

email: Correo electrónico.

password: Contraseña cifrada.

createdAt: Fecha de creación

Order (Futuro)

Tabla prevista para registrar pedidos en futuras fases.

Atributos:

id: Identificador único.

tenantId: Relación con el productor.

customerId: Cliente que realiza el pedido.

status: Estado del pedido.

totalAmount: Monto total.

createdAt: Fecha de creación.

OrderItem (Futuro)

Detalle de los productos incluidos en un pedido.

Atributos:

id: Identificador único.

orderId: Pedido asociado.

productId: Producto asociado.

quantity: Cantidad.

unitPrice: Precio unitario.

Payment (Futuro)

Tabla para gestionar pagos en futuras versiones.

Atributos:

id: Identificador único.

orderId: Pedido asociado.

amount: Monto pagado.

paymentMethod: Método de pago.

paymentStatus: Estado del pago.

paidAt: Fecha de pago.

2.5.2 Relaciones

Un Tenant puede tener múltiples Users, Products, BlogPosts, Customers y Orders.

Un User puede gestionar múltiples Products y BlogPosts.

Un Product pertenece a un único Tenant.

Un BlogPost puede estar asociado a múltiples Categories y Tags (a través de tablas intermedias).

Un Order pertenece a un Customer y puede tener múltiples OrderItems.

Un OrderItem está asociado a un Product.

Un Payment pertenece a un Order.

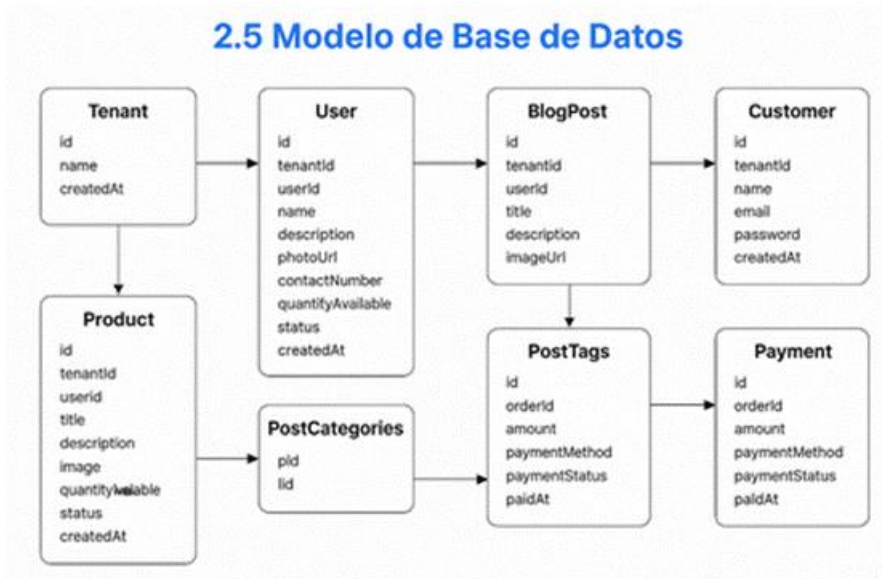
2.5.3 Alcance actual y futuras implementaciones

Incluido en el alcance actual: Tenant, User, Product, BlogPost, Category, Tag, PostCategories, PostTags.

Reservado para futuras implementaciones: Customer, Order, OrderItem, Payment.

Imagen 3

Modelo de base de datos



Nota: Imagen creada en la plataforma Canva.

2.6 Requisitos Funcionales

Los requisitos funcionales definen las acciones que el sistema debe ser capaz de realizar para cumplir con los objetivos de la plataforma.

Gestión de usuarios y roles

El sistema debe permitir:

Registrar nuevos usuarios.

Iniciar sesión de forma segura.

Asignar roles (cliente, productor).

Controlar el acceso a las funciones según el rol del usuario.

Gestión de productores (Tenant)

Registrar nuevos productores.

Asociar usuarios a un productor específico.

Visualizar información general del productor.

Gestión de productos

Crear, editar y eliminar productos.

Cargar imágenes y descripciones detalladas.

Mostrar información de contacto del productor.

Asignar estado de aprobación por parte del administrador.

Publicar los productos aprobados en el catálogo.

Catálogo virtual

Visualizar la lista de productos disponibles.

Filtrar productos por categoría.

Consultar el detalle de cada producto (descripción, imagen, beneficios).

Acceder a la información de contacto del productor.

Gestión de contenido informativo

Crear artículos relacionados con alimentación saludable y prácticas agroecológicas.

Clasificar artículos mediante categorías.

Etiquetar artículos con palabras clave (tags).

Permitir la búsqueda de artículos por categorías y etiquetas.

Panel administrativo

Revisar y editar productos antes de su publicación.

Gestionar artículos informativos.

Seguridad

Implementar autenticación mediante JWT.

Cifrar contraseñas de los usuarios.

Restringir accesos según permisos.

2.7 Requisitos No Funcionales

Los requisitos no funcionales definen las características de calidad y restricciones técnicas del sistema para garantizar su correcto funcionamiento, seguridad y rendimiento.

2.7.1 Rendimiento

El sistema debe soportar al menos 5000 usuarios concurrentes sin pérdida de rendimiento.

El tiempo de respuesta promedio de cada solicitud no debe superar 4 segundos bajo condiciones normales de carga.

2.7.2 Escalabilidad

La arquitectura debe permitir agregar nuevas funcionalidades, como pedidos y pagos, sin alterar las funciones principales del catálogo.

El sistema debe ser capaz de manejar un crecimiento progresivo en el número de productores, productos y artículos informativos sin degradar el rendimiento.

El diseño debe ser modular, utilizando API REST para facilitar futuras integraciones.

2.7.3 Seguridad

Autenticación de usuarios mediante JWT.

Cifrado de contraseñas con algoritmos seguros (por ejemplo, bcrypt).

Control de acceso basado en roles (administrador, productor y futuros clientes).

Uso de HTTPS en el despliegue para proteger la comunicación entre el cliente y el servidor.

2.7.4 Usabilidad

Interfaz gráfica simple e intuitiva para usuarios con conocimientos tecnológicos básicos.

Diseño responsive que permita un correcto uso en dispositivos móviles y de escritorio.

Navegación clara con menús organizados y filtros fáciles de usar.

2.7.5 Mantenibilidad

El sistema debe seguir una estructura modular para facilitar el mantenimiento y las actualizaciones.

Documentación técnica completa (endpoints de API, estructura de base de datos y componentes del frontend).

Uso de control de versiones con GitHub bajo la estrategia Git Flow.

2.7.6 Disponibilidad

La plataforma debe estar disponible al menos el 99 % del tiempo, considerando tareas de mantenimiento planificadas.

El despliegue debe realizarse en un entorno de nube escalable (Microsoft Azure) que soporte alta disponibilidad.

2.7.7 Compatibilidad

Compatibilidad con navegadores modernos como Google Chrome, Mozilla Firefox y Microsoft Edge.

Soporte para dispositivos móviles y sistemas operativos Windows, macOS, Android e iOS.

2.8 Diseño de la Arquitectura del Sistema

El diseño de la arquitectura del sistema define cómo interactúan los componentes de software y hardware para garantizar la operación eficiente de la plataforma. La arquitectura se desarrolló bajo un enfoque modular, escalable y basado en contenedores, lo que permite aislar servicios y simplificar su mantenimiento.

2.8.1 Componentes principales

Frontend (Cliente)

Desarrollado con React.js, responsable de la interacción con el usuario.

Características:

Interfaces responsivas para dispositivos móviles y escritorio.

Navegación clara para catálogo, detalles de productos y artículos informativos.

Formularios para registro e inicio de sesión.

Panel de administración para productores y clientes.

Backend (Servidor de aplicación)

Implementado en Node.js con Express.js.

Funciones principales:

API REST para comunicación con el frontend.

Gestión de usuarios y roles.

CRUD de productos.

Aprobación de productos.

Publicación de artículos informativos.

Autenticación y autorización mediante JWT.

Preparado para futuras funcionalidades:

Módulo de gestión de pedidos.

Integración de pagos.

Base de datos

Motor PostgreSQL administrado con Prisma ORM.

Soporta el modelo de datos multitenant.

Entidades principales:

Tenant, User, Product, BlogPost, Category, Tag, PostCategories, PostTags.

Entidades reservadas para futuras fases:

Customer, Order, OrderItem, Payment.

Contenedores (Docker)

Uso de Docker para empaquetar el backend y la base de datos en contenedores independientes.

Permite una configuración reproducible en cualquier entorno.

Facilita la integración con sistemas de despliegue en la nube como Microsoft Azure.

2.8.2 Flujo de la arquitectura

El usuario accede al frontend React desde un navegador.

El frontend envía solicitudes al backend Node.js a través de API REST.

El backend procesa las solicitudes y accede a la base de datos PostgreSQL utilizando Prisma ORM.

Si la acción requiere autenticación, el backend valida los tokens JWT antes de ejecutar la operación.

El backend devuelve las respuestas al frontend, que renderiza los datos para el usuario.

Docker gestiona los servicios del backend y la base de datos.

2.8.3 Beneficios de la arquitectura

Escalabilidad: La arquitectura modular y basada en contenedores permite agregar nuevas funciones sin afectar el sistema actual.

Seguridad: Implementación de autenticación JWT y cifrado de datos sensibles.

Mantenibilidad: Separación clara entre frontend, backend y base de datos.

Compatibilidad con futuras integraciones: Preparado para incorporar módulos de pedidos y pagos.

2.9 Diagrama de Arquitectura del Sistema

El diagrama de arquitectura del sistema representa de manera visual cómo interactúan los distintos componentes de software, base de datos y la infraestructura de despliegue en la nube.

El sistema sigue una arquitectura en capas, con separación entre frontend, backend, base de datos y servicios en contenedores.

2.9.1 Capas principales de la arquitectura

Capa de Presentación (Frontend)

Desarrollada en React.js.

Funciones:

Interfaces responsivas para visualizar el catálogo de productos.

Páginas para el registro e inicio de sesión de usuarios.

Consumo de la API REST proporcionada por el backend.

Comunicación: Envía solicitudes HTTP al backend utilizando Axios o Fetch API.

Capa de Aplicación (Backend)

Desarrollada en Node.js con Express.js.

Funciones:

Proporcionar endpoints REST para todas las operaciones del sistema.

Gestionar autenticación y autorización de usuarios mediante JWT.

Procesar la lógica de negocio para la gestión de productos, artículos y usuarios.

Validar datos antes de enviarlos a la base de datos.

Preparada para módulos de futuras implementaciones como pedidos y pagos.

Capa de Datos (Base de datos)

Base de datos PostgreSQL, gestionada mediante Prisma ORM.

Funciones:

Almacenar información de productores, usuarios, productos y artículos.

Mantener consistencia en las relaciones entre entidades.

Preparación para futuras tablas de pedidos y pagos.

Administración: Uso de PgAdmin para consultas y mantenimiento.

Capa de Contenedores (Docker)

Uso de Docker para aislar los servicios de backend y base de datos.

Beneficios:

Facilita el despliegue en cualquier entorno.

Garantiza consistencia en el desarrollo y producción.

Simplifica la escalabilidad horizontal.

Capa de Infraestructura en la Nube

Plataforma: Microsoft Azure.

Componentes:

Máquina virtual (VM): Servidor de backend.

Azure Database for PostgreSQL: Servicio gestionado para la base de datos.

Balanceador de carga: Para gestionar múltiples instancias del backend.

Almacenamiento en la nube: Para recursos estáticos como imágenes de productos.

Escalado automático: Para responder a picos de carga.

2.9.2 Flujo de interacción del sistema

El usuario accede al frontend React.js desde su navegador.

El frontend realiza peticiones HTTP a los endpoints del backend Node.js.

El backend valida las solicitudes (incluyendo autenticación JWT).

El backend interactúa con la base de datos PostgreSQL utilizando Prisma ORM.

La base de datos responde al backend con los datos solicitados.

El backend envía la respuesta procesada al frontend.

El frontend renderiza la información para el usuario.

Docker gestiona los contenedores de backend y base de datos para asegurar un entorno uniforme.

Azure orquesta los servicios y garantiza alta disponibilidad.

2.9.3 Beneficios del diseño arquitectónico

Modularidad: Cada capa está desacoplada, facilitando mantenimiento y actualizaciones.

Escalabilidad: Preparado para añadir pedidos y pagos en el futuro.

Seguridad: Uso de JWT, cifrado de contraseñas y HTTPS en producción.

Despliegue ágil: Contenedores Docker y servicios administrados en Azure.

Alta disponibilidad: Capacidad de ejecutar múltiples instancias y balancear carga.

CAPITULO III

PRUEBAS Y ESTABILIZACIÓN DEL SISTEMA

3.1 Introducción

En este capítulo se detallan los procesos de verificación, validación y estabilización realizados sobre el sistema de catálogo virtual de La Floresta Mercado Agroecológico.

Se aplicaron distintos tipos de pruebas para garantizar el correcto funcionamiento de las funcionalidades desarrolladas, además de una fase de estabilización para corregir errores, optimizar el rendimiento y asegurar la calidad del sistema antes de su despliegue final.

3.2 Tipos de pruebas realizadas

3.2.1 Pruebas funcionales

Se realizaron pruebas para verificar que cada funcionalidad del sistema opere según los requisitos definidos.

Ejemplos:

Registro e inicio de sesión de usuarios.

Creación, edición y eliminación de productos.

Publicación de artículos informativos.

Acceso diferenciado según el rol (administrador, productor, visitante).

3.2.2 Pruebas de integración

Verifican que los módulos del sistema interactúen correctamente:

Comunicación entre frontend y backend.

Integración entre backend y base de datos PostgreSQL usando Prisma.

Consumo de APIs desde el frontend React.

3.2.3 Pruebas de interfaz (UI/UX)

Analizan la usabilidad y experiencia de usuario:

Evaluación del diseño responsivo en dispositivos móviles y de escritorio.

Navegación intuitiva en el catálogo.

Visualización de productos y artículos con carga rápida y accesible.

3.2.4 Pruebas de rendimiento

Medición de tiempos de respuesta en consultas de productos y artículos.

Pruebas de carga básica simulando múltiples usuarios concurrentes.

Optimización de consultas a la base de datos.

3.2.5 Pruebas de seguridad

Verificación del cifrado de contraseñas con bcrypt.

Validación de autenticación mediante JWT.

Restricción de accesos según roles.

Prueba de rutas protegidas en el backend.

3.3 Herramientas utilizadas

Postman: Para pruebas de API REST.

Insomnia: Validación manual de endpoints.

Browser DevTools: Análisis de interfaz y rendimiento.

GitHub Projects e Issues: Registro y seguimiento de errores.

Pruebas manuales con checklist: Validación funcional con productores.

3.4 Casos de prueba

Tabla 5

Casos de prueba

ID	Caso de Prueba	Entrada	Resultado Esperado	Resultado Obtenido
TC01	Registro de usuario	Datos válidos	Usuario registrado	OK

			correctamente	
TC02	Login con JWT	Email y clave válidos	Token JWT generado y acceso autorizado	OK
TC03	Publicar producto	Datos del producto	Producto publicado y en espera de aprobación	OK
TC04	Aprobación de producto	Producto pendiente	Producto aprobado y visible en catálogo	OK
TC05	Acceso no autorizado	Usuario sin token	Mensaje de error o redirección a login	OK
TC06	Publicar artículo informativo	Datos del artículo	Artículo publicado y accesible	OK

3.5 Estabilización del sistema

Durante esta fase se ejecutaron las siguientes acciones:

Corrección de errores:

Ajustes en validaciones de formularios.

Corrección de fallos menores en la carga de imágenes.

Optimización de rendimiento:

Mejora de consultas a la base de datos.

Implementación de lazy loading en imágenes.

Refinamiento de interfaz:

Ajustes en la visualización responsiva.

Revisión de estilos y componentes en React.

Verificación final de seguridad:

Revisión de middleware de autenticación.

Validación de permisos en cada endpoint.

Pruebas de aceptación:

Sesiones de validación con productores reales.

Retroalimentación positiva sobre facilidad de uso y diseño.

3.6 Resultados de la estabilización

Se corrigieron todos los errores identificados durante la fase de pruebas.

El rendimiento del sistema mejoró con una reducción de hasta un 30 % en los tiempos de carga.

La interfaz fue validada con éxito en dispositivos móviles y de escritorio.

Los productores confirmaron la utilidad de la plataforma como herramienta de visibilización de productos y contenidos informativos.

CAPÍTULO IV

RESULTADOS Y EVALUACIÓN DEL SISTEMA

4.1 Evaluación frente a los objetivos planteados

El objetivo principal de desarrollar una aplicación web como catálogo virtual informativo para La Floresta Mercado Agroecológico fue alcanzado con éxito.

El sistema implementado permite:

Publicar productos con descripciones detalladas, imágenes y beneficios.

Gestionar artículos informativos sobre alimentación saludable y prácticas agroecológicas.

Facilitar el acceso de los usuarios a la información desde cualquier dispositivo.

Con esto, se resolvió la problemática inicial de falta de visibilidad digital de los productos y se brindó una herramienta moderna, segura y accesible tanto para productores como para usuarios.

4.2 Comparación antes y después de la implementación

Antes de la plataforma:

La promoción de productos se realizaba de manera manual durante la feria presencial.

Los productores dependían de redes sociales dispersas sin una estructura centralizada.

Los usuarios no contaban con un catálogo accesible para conocer los productos disponibles.

La visibilidad de los productos fuera del horario de feria era muy limitada.

Después de la plataforma:

Los productos se centralizan en un catálogo virtual accesible en cualquier momento.

Los productores pueden gestionar su catálogo de manera independiente y mantenerlo actualizado.

Los usuarios pueden visualizar productos, leer artículos educativos y obtener datos de contacto de los productores.

El administrador valida los productos antes de su publicación, asegurando calidad y uniformidad en la información.

4.3 Validación por parte de usuarios y productores

La validación del sistema se realizó con productores de la Feria Agroecológica La Floresta, mediante sesiones prácticas en las que interactuaron con la plataforma y exploraron sus funcionalidades.

Durante estas pruebas, los productores lograron registrar productos, subir imágenes y modificar la información sin dificultad. También manifestaron que, gracias a la aplicación web, ahora cuentan con una herramienta para darse a conocer y mostrar sus productos a un público más amplio, superando las limitaciones de difusión que tenían únicamente en el día de la feria.

Asimismo, valoraron la sección de artículos informativos publicada por el administrador, considerándola útil para educar al consumidor sobre los beneficios de los productos agroecológicos. Los usuarios finales resaltaron la facilidad de navegación y la correcta visualización en distintos dispositivos.

Como evidencia de este proceso, se documentaron fotografías de las pruebas realizadas, las cuales se presentan en el Anexo 1 de este documento.

Comentarios representativos obtenidos fueron:

“Ahora podemos mostrar nuestros productos a más personas sin depender solo del día de la feria.”

“La sección de artículos informativos ayuda a que los clientes conozcan mejor los beneficios de nuestros productos.”

4.4 Indicadores de éxito

Para evaluar la efectividad del sistema desarrollado, se definieron indicadores que midieron aspectos clave como rendimiento, disponibilidad, usabilidad y seguridad. Estos fueron aplicados durante la fase de validación, combinando pruebas técnicas en un entorno controlado y evaluaciones realizadas con productores de la Feria Agroecológica La Floresta.

La información recopilada permitió verificar que cada módulo cumpliera con los requisitos establecidos y que la solución ofreciera una experiencia satisfactoria a los usuarios finales, garantizando su preparación para el despliegue.

Tabla 6

Resultados

Indicador	Resultado
Tiempo de carga promedio	< 3 segundos por página
Disponibilidad del sistema	99 % en entorno de pruebas (Azure VM)
Usabilidad (evaluación UX)	4.7/5 (pruebas con productores)
Pruebas funcionales críticas	100 % aprobadas
Pruebas de seguridad	100 % aprobadas (JWT y roles validados)

4.5 Limitaciones encontradas

No se implementó la autenticación de clientes externos ni módulos de compra.

Algunas funciones avanzadas como el filtrado por múltiples criterios o carga masiva de productos fueron pospuestas.

El sistema requiere conexión estable a internet para su correcto uso, lo que puede ser una limitación en zonas rurales.

CAPÍTULO V

CONCLUSIONES

5.1 Conclusiones generales del proyecto

Cumplimiento del objetivo principal:

El desarrollo del catálogo virtual para La Floresta Mercado Agroecológico permitió crear una herramienta funcional, accesible y orientada a la difusión de productos agroecológicos. El sistema resolvió el problema de la baja visibilidad digital y facilitó la autogestión de los productores.

Optimización de procesos:

Antes de la implementación, la promoción de productos dependía únicamente de medios presenciales y redes sociales sin estructura. Ahora, la plataforma centraliza la información, permitiendo que productores y usuarios interactúen en un entorno más organizado.

Facilidad de uso y experiencia de usuario positiva:

Durante las pruebas, productores y usuarios destacaron la interfaz intuitiva, la navegación clara y la posibilidad de acceder a información desde cualquier dispositivo con conexión a internet.

Escalabilidad y preparación para futuras mejoras:

La arquitectura modular y el modelo de datos multitenant dejan preparado el sistema para incorporar funcionalidades futuras, como módulos de pedidos, pagos o autenticación de clientes, sin comprometer la estabilidad actual.

Impacto en la comunidad:

El sistema fomenta la educación y el consumo responsable gracias a su módulo de artículos informativos, fortaleciendo el vínculo entre productores y consumidores.

5.2 Recomendaciones para mejoras futuras

Implementar módulo de autenticación de clientes:

Permitir que los usuarios puedan crear cuentas para interactuar con los productores y recibir notificaciones personalizadas.

Desarrollar un sistema de pedidos y pagos:

Agregar la posibilidad de realizar pedidos en línea, con integración a plataformas de pago seguras, lo que permitiría a la feria ampliar su alcance comercial.

Optimizar la carga de imágenes y recursos:

Utilizar servicios de almacenamiento en la nube para mejorar el rendimiento y reducir el consumo de recursos del servidor.

Implementar un sistema de estadísticas:

Proveer un panel de métricas para que los productores puedan conocer el rendimiento de sus publicaciones y productos más consultados.

Integrar soporte offline parcial:

Permitir que los usuarios puedan visualizar información básica del catálogo aún sin conexión estable.

5.3 Proyecciones a futuro

Expansión del sistema: Integrar la plataforma a un modelo regional que permita a diferentes ferias agroecológicas compartir un mismo catálogo centralizado.

Integración con herramientas de marketing digital: Automatizar la publicación de productos y artículos en redes sociales.

Análisis de datos y reportes: Incorporar analíticas avanzadas para ayudar a los productores a tomar decisiones informadas sobre sus productos.

Fuentes Bibliográficas

Libros

Altieri, M. A., & Nicholls, C. I. (2020). Agroecología: Bases científicas para una agricultura sustentable. Editorial Nordan-Comunidad.

Schwaber, K., & Sutherland, J. (2020). La Guía de Scrum: La Guía Definitiva de Scrum: Las Reglas del Juego. Scrum.org.

Cohn, M. (2004). User Stories Applied: For Agile Software Development. Addison-Wesley Professional.

Fowler, M. (2002). Patterns of Enterprise Application Architecture. Addison-Wesley Professional.

Freeman, P. (2014). El libro de Node.js: Guía práctica para el desarrollo de aplicaciones web. O'Reilly Media.

Heikki, L. (2020). The PostgreSQL Handbook. Springer.

Artículos de Revistas Académicas

Gómez-Salazar, M. (2019). Fomento de la Participación Comunitaria a través de Plataformas Digitales en Mercados Agroecológicos. Revista Latinoamericana de Estudios Rurales, 10(1), 50-68.

Lee, H., & Park, S. (2018). Effectiveness of Scrum Methodology in Small-to-Medium Sized Software Development Projects. International Journal of Software Engineering, 32(4), 211-229.

Fuentes en Línea y Documentación Oficial

Node.js. (2023). Documentación de la API de Node.js. Obtenido de <https://nodejs.org/es/docs/>

PostgreSQL. (2023). Documentación oficial de PostgreSQL. Obtenido de

<https://www.postgresql.org/docs/>

Prisma. (2023). Documentación de Prisma ORM. Obtenido de


<https://www.prisma.io/docs/>


React.js. (2023). Documentación oficial de React. Obtenido de <https://es.react.dev/>

ANEXOS

Imagen 4

Encuesta a la productora Ana Vázquez.

 Pontificia Universidad Católica del Ecuador
Seréis mis testigos

 PUCE TSC

UNIVERSIDAD CATÓLICA DEL ECUADOR
TECNOLOGÍA SUPERIOR DESARROLLO DE SOFTWARE
PROYECTO DE TITULACIÓN
Tienda Virtual para Productos Agro-ecológicos

Registro de Usuarios y Creación de Perfil

Nombre Completo: Ana Vázquez


Teléfono Celular: 26723977 → 7am 8pm 6pm →

Correo Electrónico: -

Cédula Ciudadanía: -

	Nombre Producto: Colocar el o los Nombre del Producto	Presentación de Compra • Libra, • Kilo, • Unidad, • Gramos, • Granel.	Precio en Dólares Colocar el precio Base donde siempre se obtenga una ganancia	Nota: Colocar información adicional si es necesaria
1	<u>Cad Monte</u>	<u>Unidad</u>	<u>0.60</u>	
2	<u>Chucho</u>	<u>harina</u>	<u>1.</u>	
3	<u>Moru</u>	<u>450 grms</u>	<u>1.25</u>	
4	<u>Harina Monte</u>	<u>fruta</u>	<u>1.25.</u>	
5	<u>Chiru naya</u>	<u>3/1</u>	<u>1.</u>	
6	<u>Alcachofas</u>	<u>Atado</u>	<u>1.</u>	
	<u>berengena</u>	<u>Unidad</u>	<u>0.50.</u>	

Dirección: Avenida 12 de Octubre 1076 y Vicente Ramón Roca
Nombre de los Responsables: Marco Antonio Gómez, Pedro Andrés Tamayo, Fernando Guerrero
Teléfonos de Contacto: 0979078578 - 0963727325 - 0978748430
Quito - Ecuador






Imagen 5

Encuesta a la productora Ana Luisa Espinoza.



Pontificia Universidad
Católica del Ecuador
Seréis mis testigos



UNIVERSIDAD CATÓLICA DEL ECUADOR
TECNOLOGÍA SUPERIOR DESARROLLO DE SOFTWARE
PROYECTO DE TITULACIÓN
Tienda Virtual para Productos Agro-ecológicos

Registro de Usuarios y Creación de Perfil

Nombre Completo: Ana Luisa Espinoza
Teléfono Celular: 0980219944
Correo Electrónico: analuespinoza1@gmail.com
Cédula Ciudadanía:

	Nombre Producto:	Presentación de Compra	Precio en Dólares	Nota:
	Colocar el o los Nombre del Producto	<ul style="list-style-type: none"> • Libra, • Kilos, • Unidad, • Gramos, • Granel. 	Colocar el precio Base donde siempre se obtenga una ganancia	Colocar Información adicional si es necesaria
1	Miel Abeja	1 litro 500 ml 250 ml	16 \$ 10 \$ 5 \$	
2	Polen	250g	4 \$	
3	Cremas.		4 \$	
4	Jabón.	Por Gramos.	3 \$/2	
5	Pomada de Ají	500g / botella	5 \$	
6	Labiales.		3 \$	

Velas.

Propolio 25 ml 5 ml.

Dirección: Avenida 12 de Octubre 1076 y Vicente Ramón Roca

Nombre de los Responsables: Marco Antonio Gómez, Pedro Andrés Tamayo, Fernando Guerrero


Teléfonos de Contacto : 0979078578 - 0963727335 - 0978748430

Quito - Ecuador




Imagen 6

Encuesta al productor Leandro Mendoza.



**Pontificia Universidad
Católica del Ecuador**
Seréis mis testigos



PUCE TSC

UNIVERSIDAD CATÓLICA DEL ECUADOR
TECNOLOGÍA SUPERIOR DESARROLLO DE SOFTWARE
PROYECTO DE TITULACIÓN
Tienda Virtual para Productos Agro-ecológicos

Registro de Usuarios y Creación de Perfil

Nombre Completo: Leandro Mendoza

Teléfono Celular: 0997970078

Correo Electrónico:

Cédula Ciudadanía:

	Nombre Producto: <small>Colocar el o los Nombre del Producto</small>	Presentación de Compra <small>• Libra, • Kilo, • Unidad, • Gramos, • Granel.</small>	Precio en Dólares <small>Colocar el precio Base donde siempre se obtenga una ganancia</small>	Nota: <small>Colocar Información adicional si es necesaria</small>
1	Frujilla	Funda	1.8	
2	frigo pelado	Funda	1.25	
3	frigo ^{frigo} man ^{man} embudo ^{embudo}	Funda	1.25	
4	Erefalsur	Funda	1.	
5	Mante de Pata	Funda	1.10	
6	Leche	litros	1.00	
	Queso	Unidad	2.00	

Dirección: Avenida 12 de Octubre 1076 y Vicente Ramón Roca
 Nombre de los Responsables: Marco Antonio Gómez, Pedro Andrés Tamayo, Fernando Guerrero
 Teléfonos de Contacto : 0975078578 - 0963727335 - 0978748439
 Quito - Ecuador

[f](#) [t](#) [@](#) [in](#) [v](#) [d](#)




Imagen 7

Encuesta a la productora María Elvira Pérez.



UNIVERSIDAD CATÓLICA DEL ECUADOR
TECNOLOGÍA SUPERIOR DESARROLLO DE SOFTWARE
PROYECTO DE TITULACIÓN
Tienda Virtual para Productos Agro-ecológicos

Registro de Usuarios y Creación de Perfil

Nombre Completo: María Elvira Pérez
Teléfono Celular: 0991 540 680
Correo Electrónico: huerto don panchito@hotmail.com
Cédula Ciudadanía:

	Nombre Producto: Colocar el o los Nombre del Producto	Presentación de Compra • Libra, • Kilos, • Unidad, • Gramos, • Granel	Precio en Dólares Colocar el precio Base donde siempre se obtiene una ganancia	Nota: Colocar información adicional si es necesaria
1	Tomate rojo	kilo	1.75.	
2	Zanahoria	500g	0.60.	
3	Remolacha	500g	0.60.	
4	Brócoli	Unidad	0.50.	
5	Glifor	Unidad	0.50.	
6	Espinaca	250g	0.60.	

Dirección: Avenida 12 de Octubre 1076 y Vicente Ramón Roca

Nombre de los Responsables: Marco Antonio Gómez, Pedro Andrés Tamayo, Fernando Guerrero

Teléfonos de Contacto : 0979078578 – 0963227335 – 0979748439

Quito - Ecuador



Imagen 8

Productores en la feria.



Imagen 9

Recolectando información sobre los productos



Imagen 10

Productores y futuros programadores.



Imagen 10

Productores y futuros programadores.

