



PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR

Unidad Académica de Formación Técnica y Tecnológica – PUCE TEC

**Sistema de inventario de productos de uso básicos en una tienda pequeña local
basado en una aplicación web**

**Proyecto de titulación previo a la obtención del título de: Tecnología en
Desarrollo de Software**

Autores: Barcia Barcia Willians Andrés

Adriana Abigail Rogel Pucha

Tutor: Chilingua Tutachá Andrés Mauricio

Quito, Ecuador

2025

Índice

Agradecimientos.....	10
Introducción	1
Antecedentes.....	2
Planteamiento del Problema	2
Justificación	3
Objetivo General.....	3
Objetivos específicos.....	3
Metodología Desarrollo de Software Scrum	4
Capítulo I.....	6
Procesos actuales	6
Infraestructura actual.....	9
Metodología Scrum	10
<i>Actores que intervienen en la metodología</i>	10
<i>Fases o Ceremonias de la Metodología Scrum</i>	10
Planificación de trabajo.....	11
Capítulo II	18
Construcción del Sistema.....	18
<i>Análisis y diseño</i>	18
Mapa de Navegación	20
Frontend	20

Viabilidad Técnica.....	27
Viabilidad económica.....	27
Casos de uso.....	29
Ingresar al Sistema	29
Gestionar Productos	30
<i>Gestionar entradas de productos</i>	31
<i>Gestionar salida de productos</i>	32
Consultar inventarios	33
Gestionar roles de usuario	34
<i>Alertas de stock bajo</i>	35
Capítulo III.....	37
Requerimientos del sistema	37
Requerimientos Funcionales.....	37
1. <i>Gestión de usuarios:</i>	37
2. <i>Gestión de productos:</i>	37
3. <i>Gestión de inventario:</i>	38
4. <i>Gestión de ventas:</i>	38
5. <i>Alertas de stock bajo:</i>	38
6. <i>Generación de reportes:</i>	38
7. <i>Accesibilidad y usabilidad:</i>	39
Requerimientos No Funcionales.....	39

1. Rendimiento:	39
2. Seguridad:	39
3. Escalabilidad:	39
4. Fiabilidad y disponibilidad:	40
5. Compatibilidad:	40
6. Mantenimiento:	40
7. Usabilidad:	40
Cronograma	41
Diagrama de clases	43
Diagrama de secuencia.....	45
Ingreso de productos	45
Construcción.....	46
Modelo BDD.....	47
Diccionario de datos.....	48
Arquitectura	51
Cliente Web:	51
Funcionalidad:	51
Tecnología:	52
Servidor web:.....	52
Capitulo IV.....	53
Pruebas y resultado.....	53

Conclusiones.....	59
Recomendaciones	60
Referencias bibliográficas.....	61

Lista de tablas

Tabla 1.....	6
Tabla 2.....	9
Tabla 3.....	12
Tabla 4.....	12
Tabla 5.....	13
Tabla 6.....	13
Tabla 7.....	14
Tabla 8.....	14
Tabla 9.....	15
Tabla 10.....	15
Tabla 11.....	15
Tabla 12.....	16
Tabla 13.....	16
Tabla 14.....	17
Tabla 15.....	23
Tabla 16.....	27
Tabla 17.....	28
Tabla 18.....	28
Tabla 19.....	28
Tabla 20.....	29
Tabla 21.....	30
Tabla 22.....	31
Tabla 23.....	33
Tabla 24.....	33

Tabla 25.....	34
Tabla 26.....	35
Tabla 27.....	41
Tabla 28.....	42
Tabla 29.....	48
Tabla 30.....	49
Tabla 31.....	49
Tabla 32.....	50
Tabla 33.....	53
Tabla 34.....	54
Tabla 35.....	56
Tabla 36.....	57

Lista de figuras

Figura 1	20
Figura 2	21
Figura 3	22
Figura 4	22
Figura 5	23
Figura 6	29
Figura 7	30
Figura 8	31
Figura 9	32
Figura 10	33
Figura 11	34
Figura 12	35
Figura 13	44
Figura 14	45
Figura 15	46
Figura 16	47
Figura 17	48
Figura 18	52

DECLARACIÓN y AUTORIZACIÓN

Nosotros, Willians Andrés Barcia Barcia y Adriana Abigail Rogel Pucha con C.I. 3050594013 y C.I. 1106101676 respectivamente, autores del trabajo de titulación intitulado: Sistema de inventario de productos de uso básicos en una tienda pequeña local basado en una aplicación web, previa a la obtención del título de **Tecnología En Desarrollo De Software** en la Unidad Académica de Formación Técnica y Tecnológica PUCE TEC:

1.- Declaramos 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.- Autorizamos 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, 15 de febrero 2025

Willians Andrés Barcia Barcia

C.I. 3050594013



Adriana Abigail Rogel Pucha

C.I. 1106101676

Agradecimientos

Primeramente, le doy agradecimientos a Dios por darme las circunstancias y las herramientas para poder realizar mis estudios en esta etapa de mi vida, seguidamente agradezco a mi esposa que ha sido un pilar fundamental y me ha impulsado, motivado y apoyado para comenzar la carrera y continuar con los retos que se han presentado en las materias estos años, sin su apoyo no hubiese podido llegar hasta este punto (Willians).

Agradezco a mi madre y a mi hermana quienes, a pesar de la distancia me han brindado su apoyo de manera incondicional y aún más durante mi etapa de formación académica, en los momentos más difíciles me han demostrado su confianza que me ha servido de motivación y me ha llenado de determinación y perseverancia para entregar mi máximo esfuerzo y lograr aprobar las materias cada año (Adriana).

Introducción

La implementación de una aplicación web para la gestión del inventario básica es importante, ya que contribuirá a comprender las necesidades y desafíos de la tienda “Agua de maíz”, nos permitirá adquirir experiencia práctica al desarrollar el proyecto y de esta manera solucionar problemas reales de un pequeño negocio.

Al implementar esta solución se resolverán los problemas inmediatos de la gestión de los productos, además también fortalecerá la capacidad del negocio brindándole una herramienta con la que pueda tener una visión global del estado del inventario existente, para que de esta manera pueda enfrentar desafíos y aprovechar nuevas oportunidades de negocio pudiendo invertir en nuevos productos y descartar otros.

Como alcance se desea realizar un sistema de inventario para manejar mínimos y máximos de productos sin tener integración con sistemas de ventas o facturación, con esto establecer umbrales de excesos y faltas de stock, realizando una toma física como método inicial y provisional con una pistola/escáner de códigos de barras/QR para escanear los códigos de los productos y así poder realizar el ingreso a través de un módulo y el egreso al momento de que salga un producto y al pasarlo con el escáner se realizará la reducción del producto saliente, ya sea automáticamente por unidad o registrando la cantidad por pantalla de ser más de un solo producto.

La administración del sistema se realizará con un usuario administrador y para el uso normal otro usuario sin privilegios, esto para manejar la seguridad del sistema y evitar que personas sin autorización manipulen la data.

Para el manejo de conceptos de inventarios, nos basaremos en documentación en la web, libros de la biblioteca de la Universidad y videos relacionados a manejo de inventarios.

Antecedentes

Es muy común que en las pequeñas tiendas se realice la toma de inventario de manera manual, sin embargo, esto no exime que se cometan errores humanos en escribir o en hacer cálculos de los productos sin contar que afecta en tiempo al negocio y satisfacción completa del negocio y es por estas mismas circunstancias que se plantea aumentar una herramienta tecnológica que simplifique las tareas diarias del dueño del negocio para que de esta manera se logre una mejor eficiencia en su labor.

Planteamiento del Problema

En la tienda de barrio de venta de productos de consumo llamada Tienda “Agua de Maíz”, se ha observado un problema recurrente en la gestión manual en papel del inventario de productos, ya que no se tiene un registro exacto de existencias. La falta de un sistema adecuado no solo consume mucho tiempo, sino que también está propenso a errores humanos, como el mal registro de productos, inventarios incorrectos y pérdida de ventas por desabastecimiento de productos. Este tipo de establecimientos son esenciales para el suministro de productos básicos ya que en el área donde se encuentra el acceso a grandes supermercados es limitado. La falta de un sistema de inventario eficiente puede llevar a la pérdida de clientes que optan por tiendas mejor gestionadas, afectando así los ingresos y la viabilidad del negocio de la tienda.

Con la implementación de un sistema de inventario automatizado se mantendrá una mejor planificación de compras, evitando el desabastecimiento y el exceso de stock. Este sistema facilitará la gestión del inventario, proporcionando datos precisos para que el propietario del negocio tome decisiones en tiempo real con los productos. Además, contribuirá a que el uso de una herramienta tecnológica sea vista como un medio para modernizar y profesionalizar el negocio, mejorando la percepción y la competitividad de la tienda en el mercado local logrando la fidelidad de los clientes al negocio.

Justificación

El implementar la aplicación web facilitará en gran medida al tiempo, eficiencia y experiencia positiva al propietario del negocio aportando un mayor control sobre los ingresos, egresos y stock actualizado de los productos, con la implementación de la pistola que marca los precios le ahorrará tiempo y le permitirá agilizar los procesos que mantiene en la actualidad y con el panel de administración se le brindará al usuario seguridad del control de sus datos y la fiabilidad de sus datos en el sistema.

Objetivo General

Desarrollar e implementar una aplicación web de inventario para la tienda Agua de Maíz en la ciudad de Caracas que permita una gestión eficiente y precisa de los productos de consumo diario, mejorando la operatividad y eficiencia del negocio como también la satisfacción del cliente.

Objetivos específicos

Analizar los procesos actuales de gestión de inventario en la tienda Agua de Maíz para encontrar las novedades de mayor riesgo y mejorar el funcionamiento en el negocio.

Diseñar una base de datos usando PostgreSQL estructurándola de manera organizada para mantener un control de la información del inventario, incluyendo detalles como cantidad, fecha de ingreso, fecha de caducidad y roles de la tienda (en este caso rol de administrador y usuario).

Se desarrollará una interfaz de usuario que sea amigable y de fácil uso que le permitirá al personal de la tienda acceder, registrar ingreso, egresos y consultar información del inventario de manera inmediata, colocando limitantes de acceso en la aplicación mediante roles de usuario y administrador.

Desarrollar una API REST que permitirá generar la comunicación de la base de datos y la interfaz creada permitiendo implementar los CRUDS principales (crear, leer, actualizar y eliminar) los productos del inventario.

Implementar funciones automatizadas de alerta y reporte, que notifiquen al usuario sobre productos con bajo stock y generar reportes del inventario actual para mantener el inventario actualizado en la tienda.

Realizar pruebas de funcionamiento del producto garantizando la entrega de un software funcional al cliente y así mismo que la aplicación es estable.

Metodología Desarrollo de Software Scrum

La metodología usada en el proyecto es la Scrum, que “es un marco de gestión de proyectos ágil que ayuda a los equipos a estructurar y gestionar el trabajo a través de un conjunto de valores, principios y prácticas” (Schwaber & Sutherland, 2020). La metodología Scrum consiste en hacer entregas parciales y regulares. Los participantes tienen que estar informados acerca de las asignaciones y las responsabilidades que deben cumplir. Es ideal para aplicar en aquellos contextos donde se maneja una carga elevada de cambios o en grupos de gran tamaño, ya que es una metodología ágil con un alto nivel de flexibilidad. De esta manera, permite ajustar sus resultados y responder a las exigencias reales y exactas del cliente. (Atlassian, s. f.)

Para poder definir de manera correcta se usarán dailys en días establecidos previamente (reuniones de no más de 15 minutos para alinear los avances o bloqueos en las tareas) y para definir cada tarea se usará el product backlog (sirve para definir las tareas a realizar y asignar cada actividad que se llevará a cabo durante el sprint) (Rubin, 2012).

En scrum un proyecto se trabaja con periodos de tiempo cortos y con duración fija mediante sprints (lapsos de tiempo de 1 semana, dependiendo del proyecto y del equipo pueden ser de 3 a 4 semanas, incluyendo el feedback y análisis del producto). Las iteraciones

realizadas deben mostrar un resultado completo y una versión mejorada del producto final que sea funcional, además de ser entregado en el tiempo establecido por el cliente (Pichler, 2010).

El (product owner) prioriza los objetivos haciendo un análisis del valor que le aporta cada funcionalidad al esfuerzo y al costo de su equipo para organizar los sprints con la entrega de las tareas hacia cada miembro. (Definición de Hecho, 2018)

Capítulo I

Procesos actuales

Luego de la reunión mantenida con el propietario de la tienda, se analizaron los procesos que se llevan a diario con el inventario, los cuales se detallan a continuación:

Tabla 1

Tabla de procesos

Proceso	Responsable	Actores	Descripción del proceso	Problema identificado
Compras	Propietario	1	Comunicarse con el proveedor mediante llamada telefónica para solicitar abastecimiento de productos.	Al no tener un control total del inventario se realizar una inspección visual y de las anotaciones para saber que productos se deben reabastecer.
Recepción de compras	Ayudante de bodega y propietario	2	Registrar en un cuaderno físico cada producto entrante incluyendo la fecha de expiración.	Al tener que transcribir el detalle de los productos se suelen cometer errores al anotar la fecha real que expira un producto.

Registro de productos	Ayudante de bodega	1	Registrar cada producto en un cuaderno físico con la fecha de llegada, cantidad y fecha de caducidad.	Toma tiempo registrar el detalle de cada producto sin contar que se debe de ser muy cuidadoso para no cometer errores.
Organización manual de bodega	Ayudante de bodega y propietario	2	Organizar los productos en la tienda usando el método FIFO (first in, first out)	Se debe escribir en cada producto las etiquetas de fechas y precios
Venta de productos	Propietario y ayudante de bodega	2	Registrar la transacción en un cuaderno físico con la fecha de salida, producto vendido y la cantidad del producto saliente.	Al ser un cuaderno físico toma tiempo escribir de manera manual y en ocasiones se ha deteriorado por no estar en el lugar adecuado.
Actualización semanal de productos	Propietario	1	Realiza el cálculo de entradas y salidas comparando con las hojas que mantiene para estar al día con el stock.	Falta de precisión en el total del stock y deben volver a contar los productos para asegurarse que los datos son correctos.
Actualización diaria de productos	Ayudante de bodega y propietario	2	Conteo físico semanal de cada producto asegurando el stock actual	Diariamente se lleva el registro de los productos y a la semana se debe volver a contar los productos para corroborar los datos.

Revisión de fechas de caducidad de productos perecederos	Propietario y ayudante de bodega	2	Registran en un cuaderno físico la fecha que ingresa un producto y se realiza revisión periódica de cada producto para la venta antes de su fecha de caducidad.	Por las múltiples ocupaciones se han omitido las revisiones causando pérdida en los productos caducados.
--	----------------------------------	---	---	--

Nota: Cada proceso es la manera actual del inventario en la tienda.

Infraestructura actual

En la reunión mantenida con el propietario se identifica que posee una infraestructura tecnológica básica, los detalles de los equipos disponibles se describen a continuación:

Tabla 2

Infraestructura del negocio

Área	Cantidad	Dispositivo	Características
Despacho	1	Computadora de escritorio	Procesador Intel Core i5 de cuarta generación.
			SSD de 256 GB de almacenamiento
			8 GB de memoria RAM.
			Bluetooth.
Ethernet de 1gbps.			
Cajón de equipos de conexión	1	Router	Internet de 30mbps
Despacho	1	Impresora multifuncional	Marca HP
			Inyección de tinta a color profesional.
Despacho	1	UPS	De 750va a 500w

Nota: Equipos que mantienen en el negocio

Para desarrollar el proyecto empleamos la metodología Scrum, que está orientado a organizar las tareas y efectivizar el tiempo en cada una de ellas:

Metodología Scrum

“Scrum es un proceso de gestión que reduce la complejidad en el desarrollo de productos para satisfacer las necesidades de los clientes. La gerencia y los equipos de Scrum trabajan juntos alrededor de requisitos y tecnologías para entregar productos funcionando de manera incremental” (Schwaber & Sutherland, 2020). Scrum promueve la colaboración entre los equipos facilitando el desarrollo de productos complejos (Atlassian, s. f.).

Actores que intervienen en la metodología

Product Owner: “Representa los intereses de las partes interesadas y es responsable de gestionar el backlog del producto y definir las prioridades del trabajo.” (Schwaber & Sutherland, 2020).

Scrum Master: Se encarga de distribuir las tareas, hacer seguimiento diario de bloqueos en el desarrollo de las mismas y de los avances en cada proceso llevando un control del equipo y debe gestionar el apoyo en otras áreas si es necesario con el fin de que se puedan cumplir las tareas en el tiempo asignado. (Atlassian, s.f.)

Equipo de Desarrollo: Son profesionales con las habilidades necesarias para ofrecer un incremento de producto que pueda lanzarse. Son responsables del verdadero trabajo de creación del producto (Schwaber & Sutherland, 2020).

Fases o Ceremonias de la Metodología Scrum

Sprint Planning: “Esta es la fase inicial de cada sprint, donde el equipo decide qué trabajo se va a realizar y cómo se va a llevar a cabo. Se identifican las tareas que se extraerán del Product Backlog y se priorizan” (Schwaber & Sutherland, 2020).

Sprint Execution: “Durante esta fase, los desarrolladores trabajan en las tareas definidas, implementando la funcionalidad. Esto se realiza en un ciclo iterativo con un enfoque en la colaboración continua y la entrega de valor” (Cohn, 2009).

Sprint Review: “Al final del sprint, se presenta el trabajo realizado a las partes interesadas para recibir retroalimentación. Se analiza lo que se completó en el sprint y se realiza una demostración del incremento de producto” (Schwaber & Sutherland, 2020).

Sprint Retrospective: “Después de la Revisión, el equipo se reúne para reflexionar sobre el sprint que acaba de terminar, identificando lo que salió bien y lo que podría mejorarse en el siguiente sprint” (Schwaber & Sutherland, 2020).

Planificación de trabajo

A continuación, se detalla las tareas que se usaron durante cada sprint para el desarrollo del backend y frontend de la aplicación:

Sprints Backend

Tabla 3

SPRINT 1 (Del 23 al 27 de septiembre)

Nombre	Tarea	Prioridad	Responsable	Tiempo
Análisis y Diseño de BDD	Revisión de requerimientos y funcionalidades del sistema.	Alta	Willians Barcia	1 día
	Definición de entidades y relaciones que se usarán en la base de datos.	Alta	Willians Barcia	1 día
	Diseñar el modelado de la base de datos.	Media	Willians Barcia	1 día
Backend	Diseño de la estructura de la API REST con rutas, modelos y controladores.	Media	Willians Barcia	2 días

Nota: Análisis y diseño

Tabla 4

SPRINT 2 (Del 30 de septiembre al 4 de octubre)

Nombre	Tarea	Prioridad	Responsable	Tiempo
BDD	Crear las principales tablas para la base de datos en PostgreSQL.	Alta	Willians Barcia	1 día
Backend	Configurar el entorno de desarrollo con Node.js, Express y PostgreSQL.	Media	Willians Barcia	1 día
	Configurar las conexiones hacia la base de datos desde el backend mediante el archivo .env que contiene las variables de entorno.	Media	Willians Barcia	1 día
	Implementar la autenticación básica de usuarios (registro, inicio de sesión).	Alta	Willians Barcia	1 día

	Probar las configuraciones realizadas hacia la base de datos y autenticación.	Media	Willians Barcia	1 día
--	---	-------	-----------------	-------

Nota: Configuración inicial y base de datos

Tabla 5

SPRINT 3 (Del 7 al 11 de octubre)

Nombre	Tarea	Prioridad	Responsable	Tiempo
Backend	Crear las rutas y controladores para CRUD de productos (crear, leer, actualizar, eliminar).	Alta	Willians Barcia	2 días
	Implementar validaciones de entrada de datos.	Alta	Willians Barcia	1 día
	Corrección de las configuraciones de la base de datos y pruebas de autenticación.	Alta	Willians Barcia	1 día
Base de datos	Revisión de script, relaciones y restricciones para corregir errores	Media	Willians Barcia	1 día

Nota: Módulo de productos

Tabla 6

SPRINT 4 (del 14 al 18 de octubre)

Nombre	Tarea	Prioridad	Responsable	Tiempo
Backend	Crear las rutas y controladores para entrada y salidas de productos.	Alta	Willians Barcia	2 días
	Implementar la lógica para actualizar el stock en cada entrada/salida.	Alta	Willians Barcia	1 día
	Implementar la funcionalidad para alertas de productos por caducar.	Media	Willians Barcia	1 día

	Pruebas de configuración e integración del módulo de inventario.	Media	Willians Barcia	1 día
--	--	-------	-----------------	-------

Nota: Módulo de inventario

Tabla 7

SPRINT 5 (Del 21 al 25 de octubre)

Nombre	Tarea	Prioridad	Responsable	Tiempo
Backend	Crear las rutas y controladores para el CRUD de usuarios (crear, editar, eliminar).	Alta	Willians Barcia	2 días
	Implementar los roles y permisos para los usuarios administrador y usuario normal.	Media	Willians Barcia	1 día
	Proteger las rutas según los privilegios de los usuarios.	Alta	Willians Barcia	1 día
	Pruebas de configuración e integración del sistema de usuarios.	Media	Willians Barcia	1 día

Nota: Manejo de usuarios y roles

Tabla 8

SPRINT 6 (Del 28 de octubre al 1 de noviembre)

Nombre	Tarea	Prioridad	Responsable	Tiempo
Backend	Revisión de funcionalidad del programa de inventario junto con el frontend.	Alta	Willians Barcia	1 día
	Realizar pruebas de conexiones finales con plataforma supabase.	Media	Willians Barcia	1 día
	Configuración y preparación para el despliegue en servidor en la nube Azure.	Alta	Willians Barcia	1 día
	Despliegue y pruebas en el entorno de producción.	Media	Willians Barcia	1 día
	Redactar la documentación final del proyecto. (Base de Datos, Backend, Despliegue).	Alta	Willians Barcia	1 día

Nota: Pruebas, despliegue y documentación

Sprints Frontend

Tabla 9

Sprint 1 (Del 23 al 29 de septiembre)

Nombre	Tarea	Prioridad	Responsable	Tiempo
Análisis y Diseño	Definir la estructura y diseño de la interfaz del usuario.	Alta	Adriana Rogel	1 día
	Seleccionar el framework para la interfaz (Tailwind).	Alta	Adriana Rogel	1 día
	Crear los esquemas iniciales de vistas principales (login, productos, inventario).	Media	Adriana Rogel	1 día

Nota: Análisis y diseño

Tabla 10

Sprint 2 (Del 30 de septiembre al 06 de octubre)

Nombre	Tarea	Prioridad	Responsable	Tiempo
Frontend	Configurar el entorno de desarrollo con react y vite.	Alta	Adriana Rogel	1 día
	Configurar rutas iniciales con react-router-dom.	Alta	Adriana Rogel	1 día
	Implementar la autenticación y protección de rutas (registro, login).	Alta	Adriana Rogel	1 día
	Pruebas de configuración del entorno y autenticación.	Alta	Adriana Rogel	1 día

Nota: Configuración inicial del frontend.

Tabla 11

Sprint 3 (Del 07 al 13 de octubre)

Nombre	Tarea	Prioridad	Responsable	Tiempo
Frontend	Crear vistas para la gestión de productos (listado, formulario, CRUDS).	Alta	Adriana Rogel	1 día
	Implementar validaciones en los campos de productos.	Alta	Adriana Rogel	1 día

	Integrar las vistas de productos con el backend mediante la API.	Alta	Adriana Rogel	1 día
	Prueba de vistas e integración y la integración con la API.	Alta	Adriana Rogel	1 día

Nota: Módulo de productos

Tabla 12

Sprint 4 (Del 14 al 20 de octubre)

Nombre	Tarea	Prioridad	Responsable	Tiempo
Frontend	Crear vistas para gestionar las entradas y salidas de los productos.	Alta	Adriana Rogel	1 día
	Implementar la interfaz de visualización para el stock de productos con ayudas visuales.	Alta	Adriana Rogel	1 día
	Integrar la lógica de entradas/salidas con la API del backend.	Alta	Adriana Rogel	1 día
	Pruebas de integración con la interfaz del inventario.	Alta	Adriana Rogel	1 día

Nota: Módulo de inventario

Tabla 13

Sprint 5 (Del 21 al 27 de octubre)

Nombre	Tarea	Prioridad	Responsable	Tiempo
Frontend	Crear vistas para el CRUD de usuarios (listado, formulario).	Alta	Adriana Rogel	1 día
	Implementar la visualización de roles y permisos en las vistas.	Alta	Adriana Rogel	1 día
	Integrar con la API de usuarios y gestionar las autenticaciones.	Alta	Adriana Rogel	1 día

	Pruebas de las vistas y la integración de roles.	Alta	Adriana Rogel	1 día
--	--	------	---------------	-------

Nota: Gestión de usuarios y roles

Tabla 14

Sprint 6 (Del 28 de octubre al 03 de noviembre)

Nombre	Tarea	Prioridad	Responsable	Tiempo
Frontend	Pruebas de funcionalidad de la aplicación de inventario junto al backend.	Alta	Adriana Rogel	1 día
	Realizar pruebas de interfaz y experiencia del usuario.	Alta	Adriana Rogel	1 día
	Configuración y preparación para el despliegue en servidor en la nube Azure.	Alta	Adriana Rogel	1 día
	Despliegue de la aplicación en el servidor y pruebas en producción.	Alta	Adriana Rogel	1 día
	Documentar la configuración del frontend y manual del usuario.	Alta	Adriana Rogel	1 día

Nota: Pruebas, despliegue y documentación

Capítulo II

Construcción del Sistema

Análisis y diseño

Se realizó un análisis de la necesidad para el funcionamiento del negocio de la tienda de productos de consumo diario "Agua de Maíz", donde se identificó la necesidad de agilizar los procesos de inventario para garantizar la continuidad operativa del negocio.

Se tomó la decisión de implementar una aplicación web para la tienda debido a los problemas derivados de la toma manual del inventario, lo cual afectaba la eficiencia operativa y, por ende, la experiencia del cliente. Además, se consideró fundamental no depender de la instalación en un equipo único, ya que este podría fallar, lo que obligaría a implementar una nueva instalación y afectaría la continuidad de la gestión.

Los problemas diarios que enfrenta la tienda con el inventario manual incluyen la pérdida de tiempo que podría destinarse a otras actividades, además de la propensión a cometer errores en el registro de datos. A su vez, la falta de información actualizada y precisa puede resultar en la escasez de productos necesarios para la venta.

Se diseñará una aplicación gestionada por roles de usuario, con la que se administrarán los productos a través de atributos como nombre, código de barras, cantidad, descripción, entre otros. Esto permitirá una gestión y verificación adecuada de los productos y toda la información se almacenará en una base de datos relacional.

El frontend se desarrollará en javascript con la librería react, el backend con el framework Express.js y con la base de datos PostgreSQL para mejorar los procesos actuales, automatizando el ingreso y salida de productos, lo que reducirá significativamente los errores humanos. Además, se implementarán alertas para notificar sobre niveles bajos de stock. El sistema también permitirá conectar una pistola lectora de código de barras, lo que facilitará el

escaneo del producto y poder realizar descuento automático de los productos que salgan del inventario.

Con la implementación de esta aplicación, el propietario del negocio tendrá mayor control en el manejo de sus datos de inventario y accederá al sistema mediante su aplicación web para poder gestionar el estado de los productos, llevando un control de los ingresos y egresos, así como obtener reportes de los productos en existencia, lo que facilitará la organización y la realización oportuna de pedidos a los proveedores.

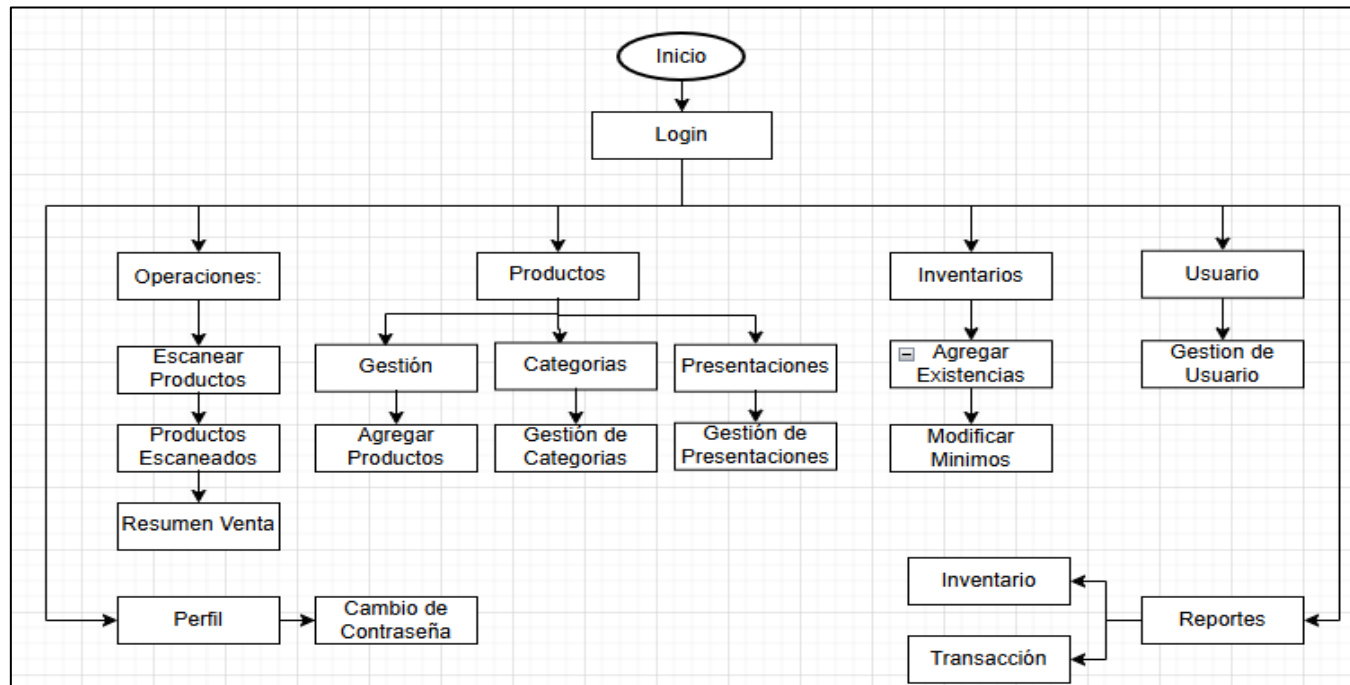
Mapa de Navegación

Frontend

Luego de revisar las necesidades con el propietario del negocio se ha pensado en un diseño el cual tendrá la siguiente navegación.

Figura 1

Figura del mapa de navegación de la aplicación



Nota: Figura donde se representa la navegación propuesta del aplicativo .

A continuación, se presentan cada pantalla y como se encuentran conectadas dentro de la app, el usuario podrá navegar entre cada una de ellas de manera interactiva:

1. Dirigirse hacia la pantalla de inicio de sesión, colocar el correo electrónico con la contraseña dentro de cada campo respectivo, dar clic en iniciar sesión e ingresará a la pantalla principal de la aplicación.

Figura 2

Interfaz Inicial

Bienvenido a SuperExpress



Inicio de sesión

Correo electrónico

Contraseña

Iniciar Sesión

2. Al estar dentro de la aplicación podrá escoger algunas opciones, entre ellas la interfaz de productos:

Figura 3

Interfaz de productos

Si seleccionamos la opción de agregar un nuevo producto

Producto	Cantidad	Descripción	Código
Coca Cola	50	Gaseosa 1.25L	
Ruffles	60	Ruffles de limón	
Alesol	100	Aceite de 1LT	

Figura 4

Ingreso de Productos

Se desplegará la siguiente pantalla con el formulario para registrar los nuevos

Ingreso de productos

Ingrese el nombre del nuevo producto

Ingrese la cantidad

Ingrese una breve descripción

Código del producto

Precio del producto

Guardar Cancelar

- Al darle clic sobre la opción de gestión de usuarios, se despliega la siguiente pantalla en la que le permitirá al usuario administrador asignar el rol que desee:

Figura 5

Interfaz de Usuarios

	Id	Nombre	Correo	Contraseña	Rol
Productos Gestión de usuarios	1	Willians Barcia	willians.barcia@tienda.com	*****	Administrador
	2	Adriana Rogel	adriana.rogel@tienda.com	*****	Usuario▼
	3	Carlos Usuario	carlos.usuario@tienda.com	*****	Usuario▼ Administrador

El administrador puede asignar el tipo de rol

Tabla 15

Tabla de relación de las API

Método	Ruta	Controlador	Middleware	Descripción
POST	/login	loginUsuario	None	Inicia sesión de usuario
GET	/usuario	obtenerUsuarioActual	authMiddleware	Obtiene los datos del usuario actual

GET	/categorias	obtenerCategorias	authMiddleware	Obtiene todas las categorías
GET	/categorias/:id	obtenerCategoriaPorId	authMiddleware	Obtiene una categoría por ID
POST	/categorias	crearCategoria	authMiddleware	Crea una nueva categoría
PUT	/categorias/:id	actualizarCategoria	authMiddleware	Actualiza una categoría por ID
DELETE	/categorias/:id	eliminarCategoria	authMiddleware	Elimina una categoría por ID
POST	/entradas	registrarEntrada	authMiddleware	Registrar el ingreso de productos al inventario
POST	/salidas	registrarSalida	authMiddleware	Registrar un producto saliente del inventario

GET	/productos	obtenerProductos	authMiddleware	Nos muestra los productos buscados
GET	/productos/:id	obtenerProductoPorId	authMiddleware	Obtención de un producto buscado por ID
POST	/productos	crearProducto	authMiddleware	Creación de un producto nuevo
PUT	/productos/:id	actualizarProducto	authMiddleware	Actualiza el producto a través del ID
DELETE	/productos/:id	eliminarProducto	authMiddleware	Elimina productos según su ID
GET	/roles	obtenerRoles	authMiddleware	Obtiene todos los roles de usuario
GET	/roles/:id	obtenerRolPorId	authMiddleware	Obtiene un rol de usuario por ID

POST	/roles	crearRol	authMiddleware	Crea un nuevo rol
PUT	/roles/:id	actualizarRol	authMiddleware	Actualiza un rol por ID
DELETE	/roles/:id	eliminarRol	authMiddleware	Elimina un rol por ID
GET	/usuarios	obtenerUsuarios	authMiddleware	Obtiene todos los usuarios
GET	/usuarios/:id	obtenerUsuarioPorId	authMiddleware	Obtiene un usuario por ID
POST	/usuarios	crearUsuario	authMiddleware	Crea un nuevo usuario
PUT	/usuarios/:id	actualizarUsuario	authMiddleware	Actualiza un usuario por ID
DELETE	/usuarios/:id	eliminarUsuario	authMiddleware	Elimina un usuario por ID
PUT	/usuarios/:id/password	cambiarContraseña	authMiddleware	Cambia la contraseña de un usuario

POST	/ventas	registrarVenta	authMiddleware	Registra una venta
------	---------	----------------	----------------	--------------------

Nota: Tabla de navegación de funciones de las API.

Viabilidad Técnica

De acuerdo con las necesidades de la empresa se detallan los requerimientos necesarios en recursos para poder desarrollar el proyecto.

Tabla 16

Viabilidad técnica

Área	Cantidad	Dispositivo	Características
Inventario	1	Computador de escritorio.	8 GB de RAM Procesador Intel Core, disco duro 250GB.
Caja	1	Host	Compatibilidad con SO Windows. Backup automático, detección de intrusos.
Ventas	1	Pistola etiquetadora de precios	Impresión térmica. Recargable, dimensiones estándar.
Bodega	1	Impresora	Inyección a tinta. Impresión de precios y código de barras.
Caja	1	Sistema operativo	Windows o Linux.
Venta	1	Wifi	Señal de internet, conexión estable.

Nota: Detalle de recursos necesarios para la implementación de la aplicación.

Luego de la revisión de viabilidad técnica del negocio se confirma que el proyecto si es viable por lo que se puede continuar con el desarrollo e implementación.

Viabilidad económica

Para desarrollar el proyecto se debe verificar gastos básicos como el personal, gastos de mantenimiento y otros gastos generales, los mismos detallados a continuación:

Tabla 17*Gastos de mantenimiento de la app*

Cantidad	Equipo / Dispositivo	Periodicidad / Licencia	Costo
1	Servidor web	Mensual	\$15.00
1	Computadora de escritorio	Existente	\$0.00
1	Internet	Mensual	\$20.00
1	Suministros de tinta y etiquetas	Mensual	\$10.00
Total			\$45.00

Nota: Detalle de gastos básicos para el mantenimiento de la aplicación en línea.

Tabla 18*Gastos del desarrollador*

Cantidad	Cargo	Periodicidad	Valor mensual	Total
1	Desarrolladores frontend y backend	4 meses	\$ 0.00	\$ 0.00
1	Alimentación	4 meses	\$ 40.00	\$160.00
1	Servicios básicos	4 meses	\$50.00	\$200.00
1	Transporte	1 mes	\$25.00	\$25.00
Total				\$385.00

Nota: Detalle de gastos del desarrollador en el proyecto.

Tabla 19*Resumen de costos*

Descripción de gastos	Valor
Gastos de la app	\$45.00
Gastos usados en el desarrollador	\$385.00
Total	\$430.00

Nota: Costos totales en el desarrollo del proyecto.

Luego de la revisión de viabilidad económica del negocio se confirma que el proyecto si es viable por lo que se puede continuar con el desarrollo e implementación.

Casos de uso

Los casos de uso son herramientas utilizadas para describir cómo un sistema interactúa con los usuarios de manera eficiente, detallando las funcionalidades y procesos desde la perspectiva del usuario (Sommerville, 2011).

Ingreso al Sistema

Este caso de uso “Ingresar al sistema” permite a los usuarios registrados acceder al sistema, proporcionando sus credenciales (usuario y contraseña). Si las credenciales son correctas, se inicia una sesión donde podrá hacer uso de las funciones de la aplicación.

Figura 6

Diagrama de caso de uso 1.1

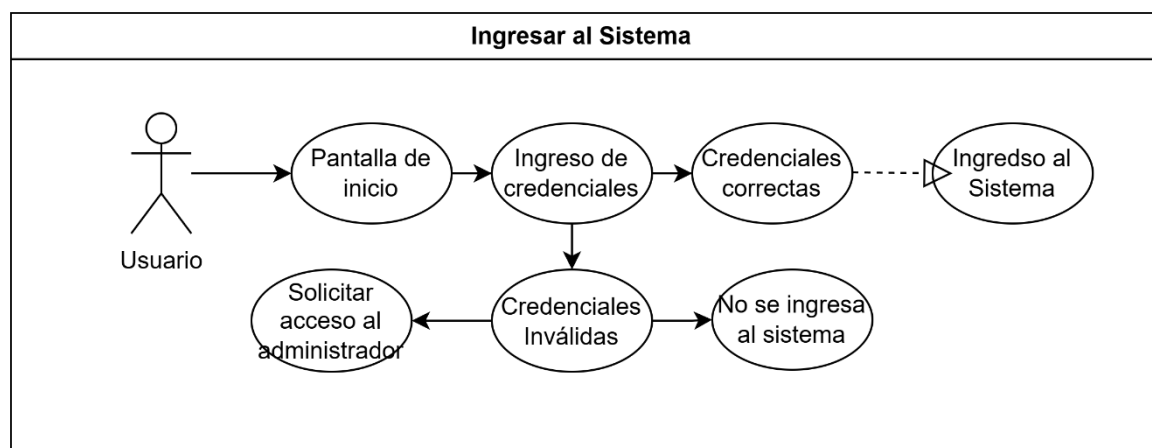


Tabla 20

1.1 Ingresar al Sistema (Autenticación de usuarios)

Nombre:	1.1 Ingresar al sistema
Descripción:	El usuario debe ingresar su correo electrónico con su contraseña para poder acceder hacia el contenido del sistema.
Actores:	Usuarios (Administrador y Empleado)
Precondiciones:	Usuario ya registrado en el sistema.
Requisitos no Funcionales:	Validación de datos ingresados

Flujo de Eventos:

1. El usuario ingresa sus credenciales.
2. El sistema realiza la validación de las credenciales ingresadas.
3. Si son correctas, el usuario accede al contenido del sistema de acuerdo al rol (Administrador o Empleado).
4. Si son incorrectas, el sistema le mostrará un mensaje de error.

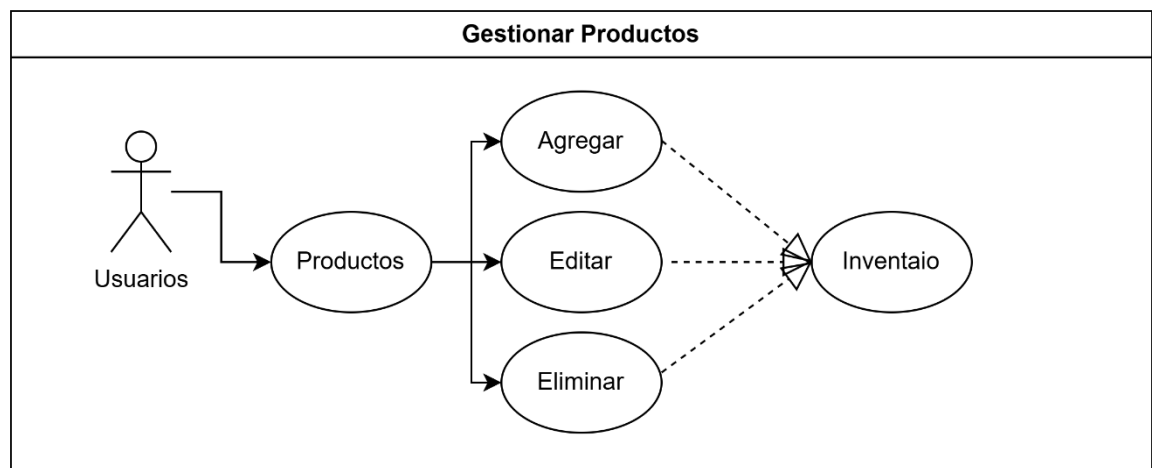
Flujo Alternativo:

- 3.1. Si el usuario no se encuentra registrado en el sistema, deberá solicitar al administrador la creación de una cuenta para el acceso.

Nota: Casos de uso detallados con los procesos de autenticación en sistema.

Gestionar Productos

El caso de uso “Gestionar productos” permite a los usuarios administradores que ingresan al sistema, poder realizar la gestión de productos como agregar, editar o eliminar producto del inventario.

Figura 7*Diagrama de caso de uso 1.2***Tabla 21***1.2 Gestionar productos (Administrador)*

Nombre:	1.2 Gestionar productos (Administrador)
Descripción:	El administrador puede agregar, editar o eliminar productos del inventario.
Actores:	Administrador

Precondiciones: Usuario con rol de administrador.

Requisitos no Funcionales: Validación de los datos ingresados

Flujo de Eventos:

1. El administrador accederá hacia la interfaz de gestión de productos.
2. El administrador visualizará la lista del inventario con los productos existentes.
3. El administrador agregará nuevos productos, editando los detalles como nombre, código de barras, precio, stock y categoría.
4. El administrador puede eliminar productos del inventario.
5. El sistema actualiza el inventario en tiempo real.

Flujo Alterno:

- 3.1. Si el administrador agrega o edita un producto sin completar todos los campos (como nombre, precio, etc.), el sistema le mostrará un mensaje de error con la notificación que debe llenar todos los campos faltantes.

Nota: Casos de uso detallados con los procesos de gestión de productos por un administrador.

Gestionar entradas de productos

En el caso de uso “Gestionar entradas de productos” los usuarios pueden ingresar nuevos productos en el inventario.

Figura 8

Diagrama de caso de uso 1.3

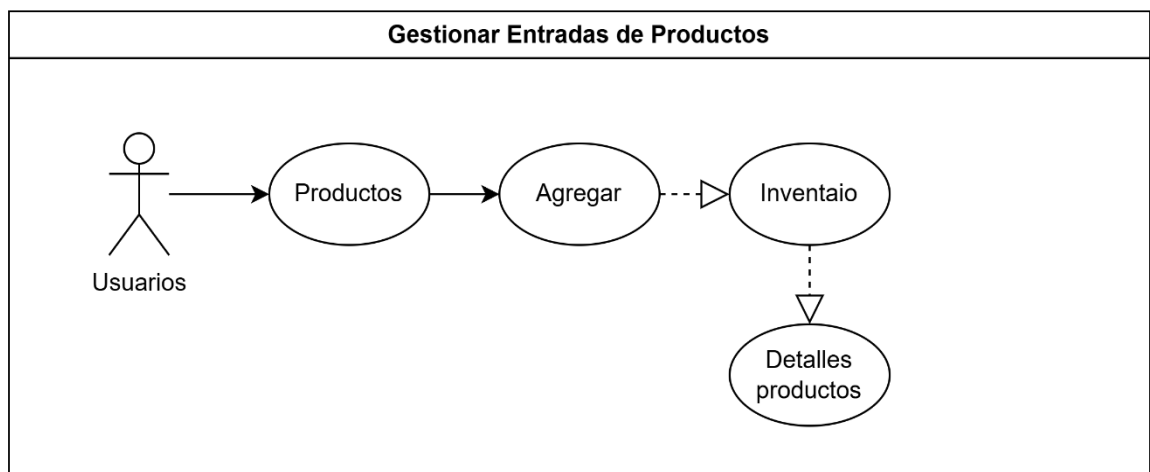


Tabla 22

1.3 Gestionar entradas de productos.

Nombre:	1.3 Gestionar entradas de productos
----------------	-------------------------------------

Descripción: Registrar las entradas de productos al inventario.
Actores: Administrador o Empleado
Precondiciones: El producto ya debe estar previamente registrado en el sistema.
Requisitos no Funcionales: Validación de datos ingresados
Flujo de Eventos: <ol style="list-style-type: none"> 1. El usuario (Administrador o Empleado) ingresa a la sección de entradas del inventario. 2. El usuario selecciona el producto y registra la cantidad que entra en el inventario. 3. El sistema actualiza el stock disponible de ese producto. 4. El sistema guarda la información con la fecha, usuario y observaciones.
Flujo Alternativo: <ol style="list-style-type: none"> 2.1. Si la cantidad ingresada es negativa le reflejará un mensaje de error y se le solicitará que se ingrese una cantidad válida. 2.2. Si el producto no está registrado en el sistema, el sistema solicita que se registre el producto.

Nota: Casos de uso detallados con los procesos de gestión de entrada de productos.

Gestionar salida de productos

El caso de uso “Gestionar salida de productos” permite a los usuarios poder realizar la salida de productos del inventario.

Figura 9

Diagrama de caso de uso 1.4

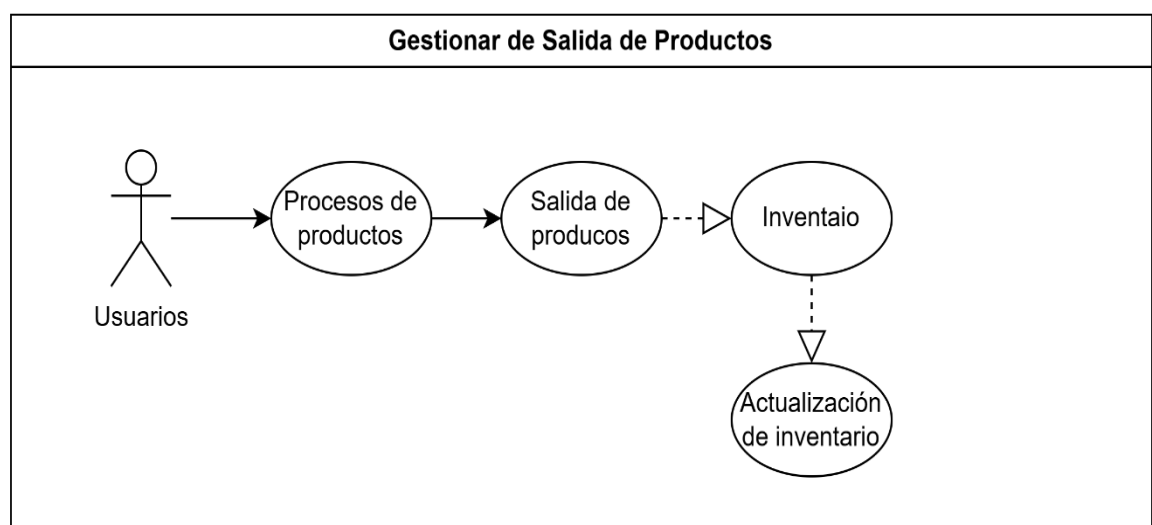


Tabla 23

1.4 Gestionar salida de productos (Empleado)

Nombre:	1.4 Gestionar salida de productos
Descripción:	Registrar las salidas de productos del inventario.
Actores:	Administrador o Empleado
Precondiciones:	Producto ya registrado en el sistema y tener stock disponible.
Requisitos no Funcionales:	Validación de datos ingresados
Flujo de Eventos:	<ol style="list-style-type: none">1. El usuario (Administrador o Empleado) ingresará a la sección de salidas del inventario.2. El usuario selecciona el producto y registra la cantidad que sale del inventario.3. El sistema actualiza el stock disponible de ese producto.4. El sistema guarda la información con la fecha, usuario y observaciones.

Nota: Casos de uso detallados con los procesos de gestión de salida de productos.

Consultar inventarios

El caso de uso “Consultar inventarios” permite a los usuarios ver o verificar visualmente el inventario actual existente.

Figura 10

Diagrama de caso de uso 1.5

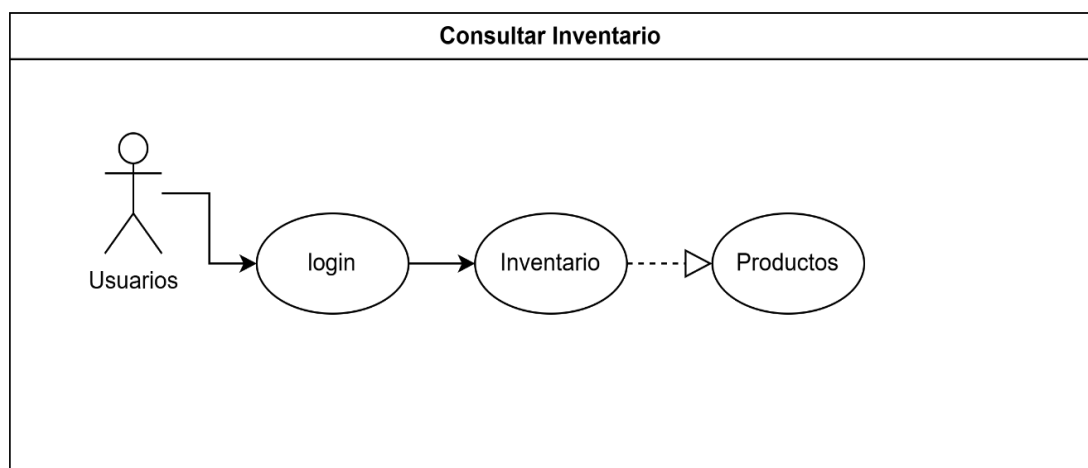


Tabla 24

1.5 Consultar inventario

Nombre:	1.5 Consultar inventarios
Descripción:	Consultar el estado del inventario de productos, incluyendo la cantidad en stock, la categoría y la información de caducidad si aplica.
Actores:	Administrador o Empleado
Precondiciones:	Ingresar al sistema
Requisitos no Funcionales:	Validación de datos ingresados
Flujo de Eventos:	<ol style="list-style-type: none"> 1. El usuario accederá hacia la sección de consulta de inventario. 2. El sistema le mostrará la lista de los productos con sus detalles actualizados (cantidad en stock, fecha de caducidad, etc.). 3. El usuario puede filtrar o buscar productos por nombre, categoría o código de barras.

Nota: Casos de uso detallados con los procesos de consulta de inventario de productos.

Gestionar roles de los usuarios

El caso de uso “Gestionar roles de usuario” le permite al usuario administrador editar, asignar roles de administrador o usuario a los usuarios registrados dentro del sistema.

Figura 11

Diagrama de caso de uso 1.6

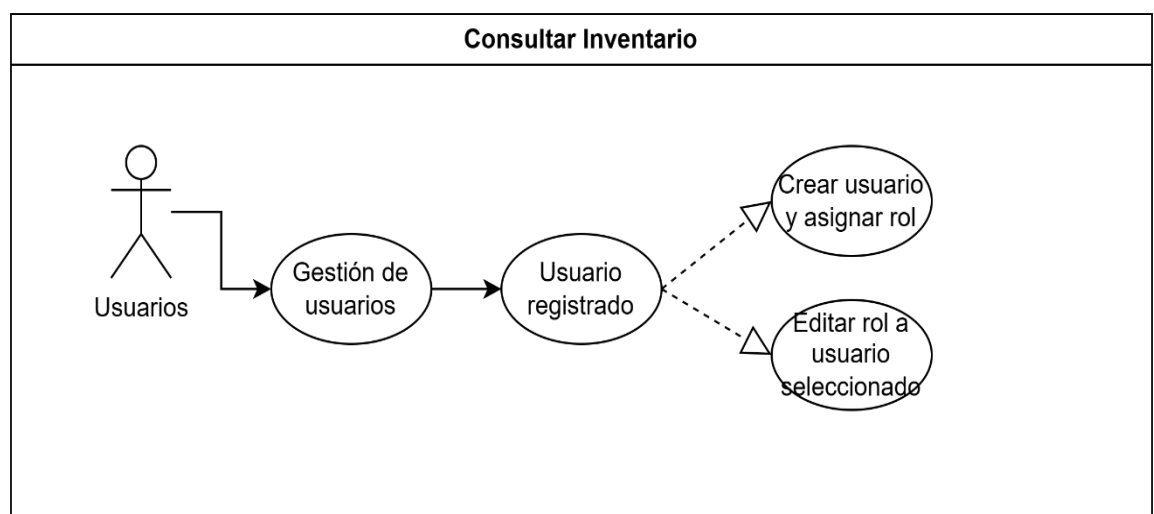


Tabla 25

1.6 Gestionar roles de usuario

Nombre:	1.6 Gestionar roles de usuario
Descripción:	El administrador puede asignar y modificar los roles de los usuarios (Administrador, Empleado).
Actores:	Administrador
Precondiciones:	Usuario con rol de administrador.
Requisitos no Funcionales:	Validación de los datos ingresados
Flujo de Eventos:	<ol style="list-style-type: none"> 1. El administrador debe dirigirse hacia la sección de gestión de usuarios. 2. El administrador puede agregar usuarios, asignarles roles y editar los roles existentes. 3. El sistema actualiza los roles de acceso y permisos según lo configurado.
Flujo Alternativo:	<ol style="list-style-type: none"> 2.1. Si el usuario esta creado muestra un mensaje que el usuario ya existe.

Nota: Casos de uso detallados con los procesos de gestión de roles de usuario.

Alertas de stock bajo

El caso de uso “Alertas de stock bajo” permite al usuario administrador poder visualizar notificaciones de stock bajo de los productos para así poder realizar acciones necesarias como reponer inventario.

Figura 12

Diagrama de caso de uso 1.7

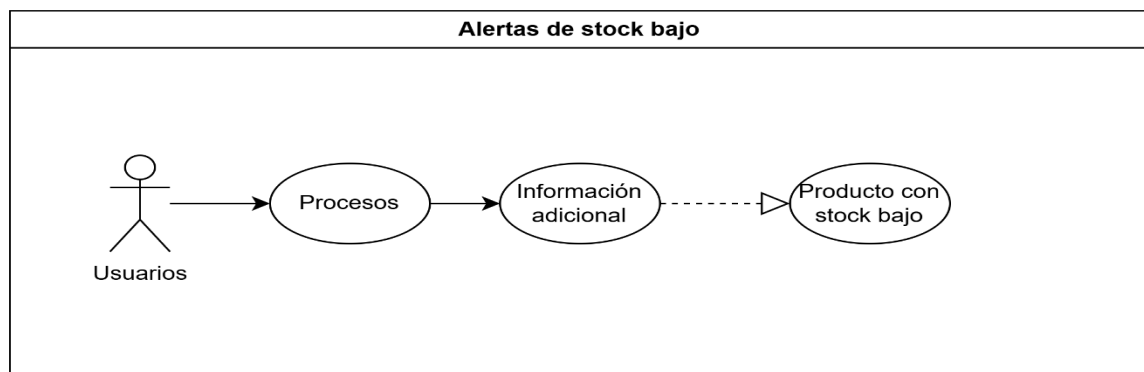


Tabla 26

1.7 Alertas de Stock bajo

Nombre:	1.7 Alertas de Stock bajo
Descripción:	El sistema deberá generar alertas si un producto se encuentra en un nivel inferior bajo sobre el stock definido.
Actores:	Administrador
Precondiciones:	El producto debe tener un nivel mínimo de stock configurado.
Requisitos no Funcionales:	Validación de datos ingresados
Flujo de Eventos:	<ol style="list-style-type: none">1. El sistema verifica el stock de cada producto.2. Cuando un producto alcanza el nivel mínimo, el sistema genera una alerta.3. El administrador recibe la alerta para proceder con la reposición del producto.
Flujo Alternativo:	<ol style="list-style-type: none">2.1. Si no hay productos con stock bajo, no se muestra ninguna alerta.

Nota: Casos de uso detallados con los procesos de gestión de alertas por stock bajo.

Capítulo III

Requerimientos del sistema

Luego de revisado el flujo de trabajo y casos de usos para la implementación, podemos listar los requerimientos del sistema, tanto los funcionales como los no funcionales.

Los requerimientos funcionales son las partes del programa con la cual el usuario interactuará en el sistema y los requerimientos no funcionales son aquellas funciones del programa que no son visibles a simple vista como, por ejemplo, la implementación de la autenticación de usuarios. A continuación, se detallan:

Requerimientos Funcionales

1. *Gestión de usuarios:*

- El sistema permitirá la creación, edición y eliminación de los usuarios con sus roles diferenciados (Administrador, Empleado).
- El sistema debe permitir asignar roles específicos a los usuarios (Administrador o Empleado), con permisos asociados para cada uno.
- El sistema le debe permitir al usuario la autenticación a través de su nombre de usuario y su contraseña registrada.

2. *Gestión de productos:*

- El sistema le debe permitir a cada administrador editar, agregar y eliminar los productos en el inventario.
- Los productos deben incluir información detallada como: nombre, descripción, código de barras, precio, cantidad en stock, y fecha de caducidad.
- El sistema debe permitir gestionar categorías de productos (por ejemplo, Lácteos, Bebidas, etc.).

- El sistema debe permitir la búsqueda de productos por nombre, código de barras o categoría.

3. *Gestión de inventario:*

- El sistema debe aceptar la entrada de los productos nuevos hacia el inventario.
- El sistema debe permitir que se puedan registrar la salida de los productos del inventario, actualizando el stock en tiempo real.
- El sistema debe almacenar en un historial las entradas y salidas de los productos con la información de fecha, cantidad, y usuario que registró la operación.

4. *Gestión de ventas:*

- El sistema le debe permitir a cada empleado registrar la venta de los productos salientes.
- En el sistema al registrar una venta el inventario se actualizará de manera automática.

5. *Alertas de stock bajo:*

- El sistema debe generar una alerta cuando el nivel del stock de un producto baje al mínimo que se defina en el producto.
- Las alertas deben mostrarse como notificación para que el administrador pueda tomar acciones.

6. *Generación de reportes:*

- El administrador podrá generar reportes sobre las ventas realizadas en el sistema con los detalles sobre los productos que se vendieron, cantidad, ingresos y las fechas de los mismos.
- El sistema debe permitir exportar los reportes en formato PDF.

7. Accesibilidad y usabilidad:

- El sistema debe ser accesible a través de una aplicación web desde cualquier dispositivo con conexión a Internet.
- Para los usuarios sin experiencia previa contarán con una interfaz que será amigable, intuitiva y de fácil uso.

Requerimientos No Funcionales

1. Rendimiento:

- El sistema debe de manejar al menos de 25 transacciones por minuto y no debe de experimentar lentitud o tiempos de respuesta inadecuados.
- El tiempo de respuesta de cada operación en el sistema no debe exceder de 2 segundos bajo condiciones normales de carga.

2. Seguridad:

- Las contraseñas de los usuarios deben ser encriptadas utilizando un algoritmo seguro como bcrypt.
- El sistema debe implementar un control de acceso que estará enfocado principalmente en cada rol y con esto, se podrá gestionar los accesos hacia las funcionalidades según el tipo de usuario (Administrador o Empleado).
- El sistema debe mantener los registros de auditoría de las acciones que realicen los usuarios dentro del sistema, como la creación, edición o eliminación de productos.

3. Escalabilidad:

- El sistema debe ser escalable para soportar un crecimiento futuro del negocio, permitiendo aumentar el número de productos, usuarios y de las transacciones sin que esto afecte al rendimiento de la aplicación.

4. *Fiabilidad y disponibilidad:*

- El sistema debe contar con un 99.9% de disponibilidad durante el horario de operación (excepto durante las horas de mantenimiento planificado).

5. *Compatibilidad:*

- El sistema deberá ser compatible con los distintos navegadores web actuales y que sean los más utilizados (Google Chrome, Mozilla Firefox, Safari).
- El sistema debe ser responsivo, es decir, debe adaptarse a diferentes tamaños de pantalla (desde computadoras de escritorio hasta dispositivos móviles tipo tablets).

6. *Mantenimiento:*

- El sistema debe permitirle al usuario actualizar los productos, entradas y salidas de inventario sin interrumpir las operaciones.

7. *Usabilidad:*

- El sistema contará con su interfaz de usuario clara, sencilla, de fácil navegación y que permita la realización de tareas y consulta de información para los usuarios sin conocimientos técnicos.

Cronograma

Para poder realizar todas estas labores, nos basaremos en un cronograma por el cual nos regiremos para poder realizar las diferentes tareas de desarrollo e implementación del programa en su backend y frontend.

Tabla 27

Cronograma de trabajo

Módulo	Iteración	Día de inicio	Requerimiento	Prioridad	Tarea	Responsable	Duración (días)
Recolección de información	1	1	Analizar los requisitos del cliente	Alta	Investigación de la necesidad del usuario en la tienda	Willians Barcia	5
Diseño de BDD	2	5	Diseñar la base de datos	Alta	Diseño de la base de datos en la página de supabase	Willians Barcia	7
Configuración	3	12	Configuración inicial del sistema	Alta	Configuración del entorno de desarrollo del aplicativo	Willians Barcia	7
Diseño del backend	4	19	Diseño de la interfaz del aplicativo con su respectiva base de datos.	Alta	Diseñar el código necesario para el funcionamiento de la aplicación	Willians Barcia	7

Implementación	5	26	Implementación de funcionalidades del backend con el frontend	Alta	Desarrollo de las conexiones del al frontend	Willians Barcia	7
Pruebas	6	33	Pruebas de funcionamiento	Alta	Pruebas iniciales en un host local junto con la base de datos remota	Willians Barcia	7

Nota: Cronograma de trabajo aplicado al desarrollo del backend.

Tabla 28

Cronograma de la app frontend

Módulo	Iteración	Día de inicio	Requerimiento	Prioridad	Tarea	Responsable	Duración (días)
Recopilación de la información	1	1	Análisis de requisitos	Alta	Investigación de acuerdo a la necesidad del usuario	Adriana Rogel	5
Diseño de interfaz	1	12	Diseño de la interfaz del aplicativo	Alta	Diseño de mockups para la aplicación	Adriana Rogel	10

Configuración	1	21	Configuración inicial del sistema	Alta	Configuración del entorno de desarrollo	Adriana Rogel	9
Implementación	1	27	Implementación de las funcionalidades básicas	Alta	Desarrollo del frontend y conexiones con el backend	Adriana Rogel	24
Pruebas	1	48	Pruebas de funcionamiento	Alta	Pruebas iniciales de la app en un ambiente local	Adriana Rogel	20

Nota: Cronograma de tareas desarrolladas del frontend.

Diagrama de clases

Para proceder con la creación del código del backend, definiremos como debería ser la estructura e interacción de nuestras clases y las estableceremos de acuerdo al siguiente gráfico:

Figura 13

Diagrama UML de clases del software

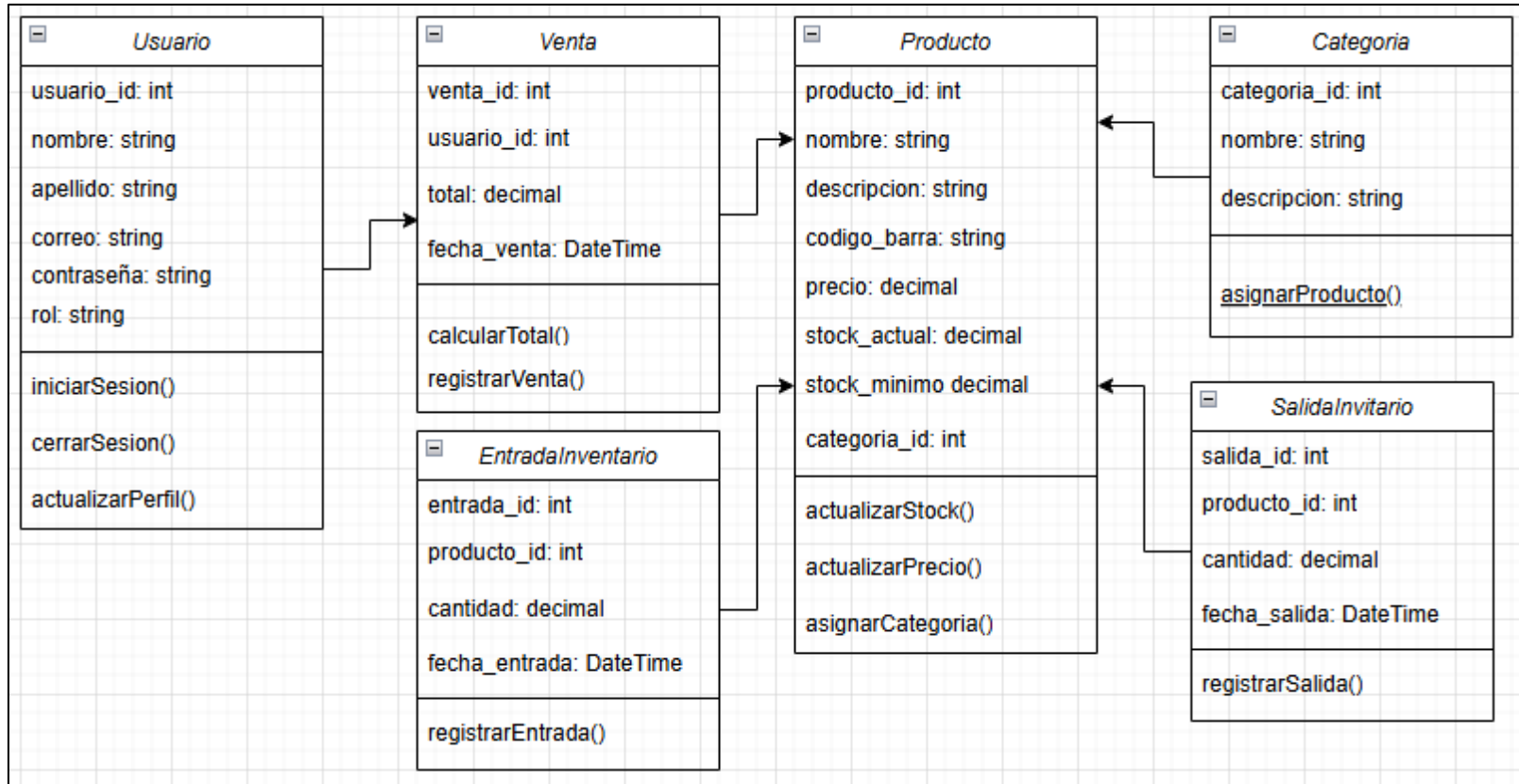


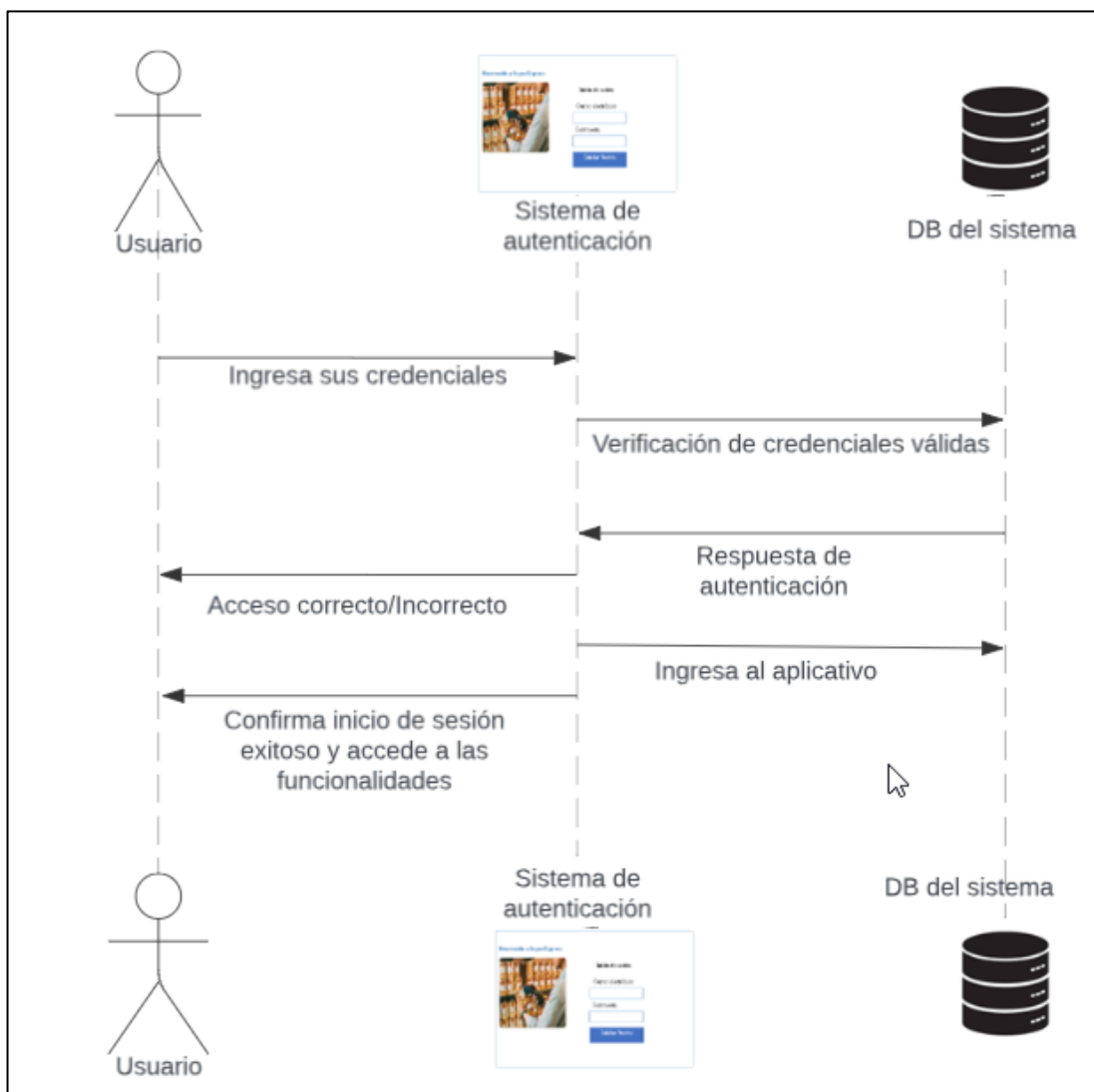
Diagrama de secuencia

Nos permite verificar el proceso en el cual el usuario ingresa hacia el sistema y realiza el proceso con cada elemento en la aplicación.

Inicio de sesión. - En el siguiente diagrama se especifica el ingreso del usuario hacia la aplicación, muestra la autenticación del usuario e “inicio de sesión”:

Figura 14

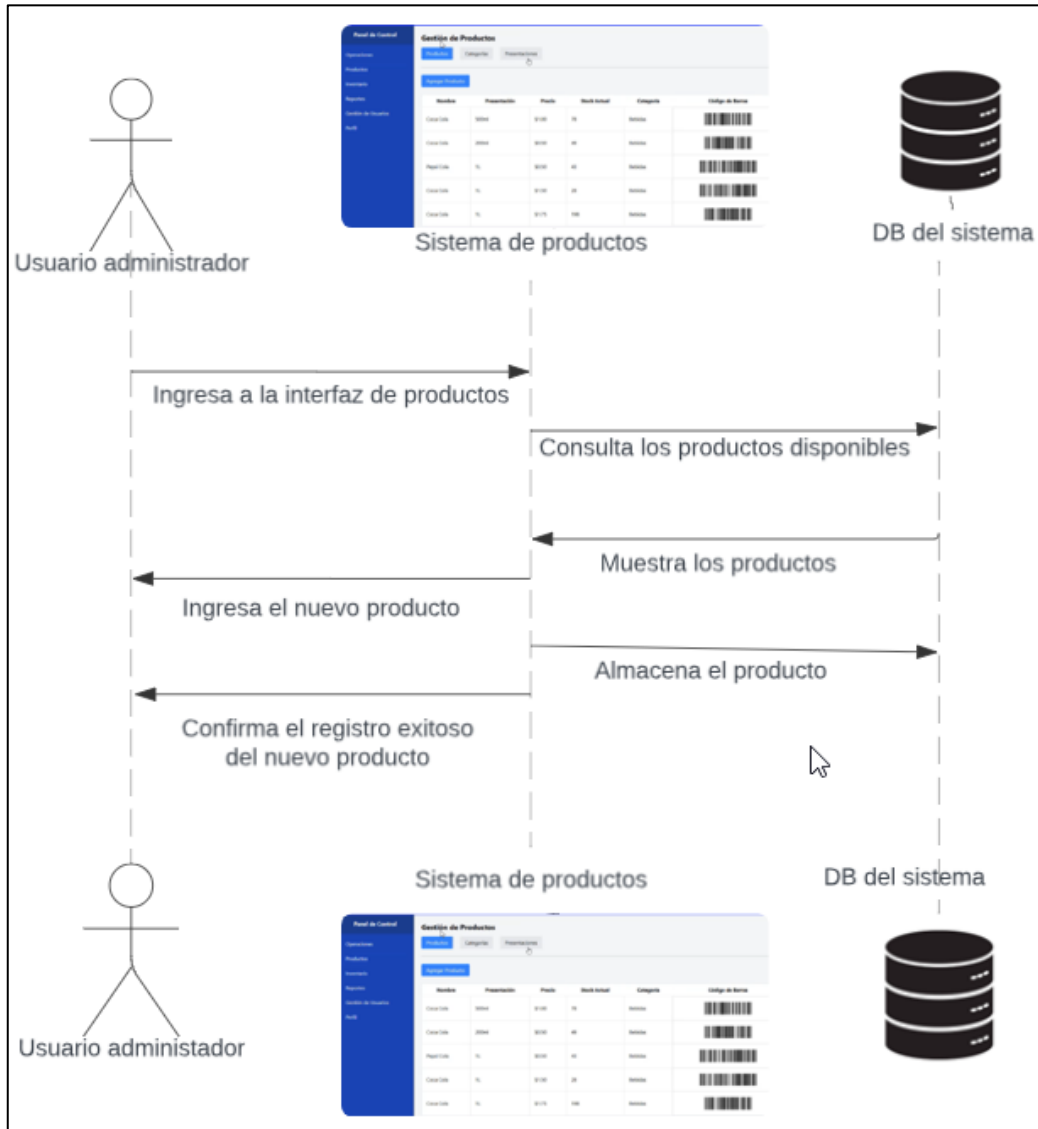
Diagrama de secuencia: Inicio de sesión



Ingresar los productos. - En el siguiente diagrama se visualiza el proceso que realiza el usuario para registrar un nuevo producto en el aplicativo:

Figura 15

Diagrama de secuencia: Ingreso de productos



Construcción

Para la construcción del sistema se está usando lo siguiente:

Javascript: Lenguaje de programación que se usa para implementar la lógica del aplicativo con el backend y el frontend. (Fierro, s.f.)

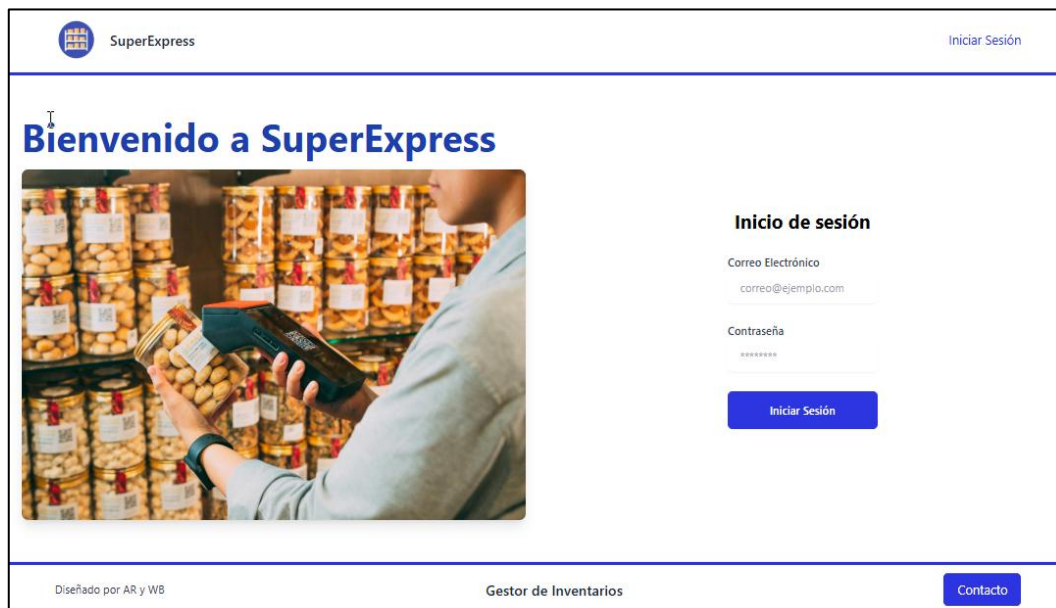
Node.js: Contiene un sistema de módulos que permite tener una mejor organización del código en la aplicación, además nos permite conectarnos a bases de datos y trabajar con directorios. (Fierro, s.f.)

Expres.js: Es el framework usado para crear la estructura en la aplicación. (MDN, s.f.)

Interfaz de inicio: Esta interfaz se diseñó de manera intuitiva con los campos para ingresar el correo y la contraseña, contine el botón de inicio de sesión y los campos contienen validaciones para que el usuario tenga conocimiento si está ingresando la información correcta.

Figura 16

Interfaz visual de inicio de sesión

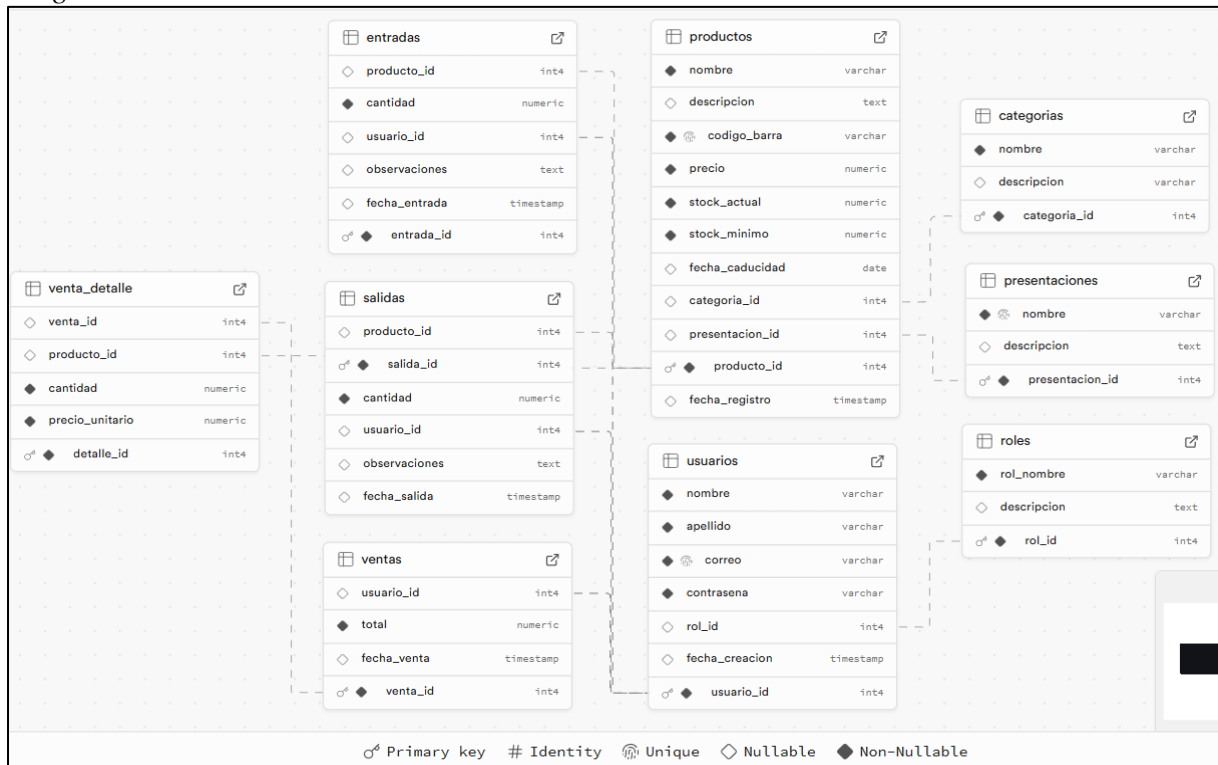


Modelo BDD

Para almacenar y consultar los datos en la aplicación se usó la base de datos PostgreSQL la cual tiene la siguiente estructura.

Figura 17

Diagrama de base de datos



Diccionario de datos

"El diccionario de datos es un conjunto de metadatos que describe los elementos de datos de un sistema de información, incluyendo sus nombres, tipos, valores permitidos y relaciones con otros datos" (Rob & Coronel, 2019, p. 145).

Tabla 29

Tabla de roles

Campo	Tipo de Dato	Restricciones	Descripción
rol_id	SERIAL	PRIMARY KEY	Identificador único
rol_nombre	VARCHAR(50)	NOT NULL	Nombres del rol
descripcion	TEXT		Detalle del rol

Nota: Descripción de los datos de la tabla de roles.

Tabla 30*Tabla de usuarios*

Campo	Tipo de Dato	Restricciones	Descripción
usuario_id	SERIAL	PRIMARY KEY	Identificador único de los usuarios
nombre	VARCHAR(100)	NOT NULL	Denominación del usuario
apellido	VARCHAR(100)	NOT NULL	Apellido de cada usuario
correo	VARCHAR(100)	UNIQUE, NOT NULL	Correo electrónico de cada usuario
contrasena	VARCHAR(255)	NOT NULL	Contraseña de los usuarios
rol_id	INT	FOREIGN KEY (roles)	Relacionado con la tabla de roles
fecha_creacion	TIMESTAMP	DEFAULT NOW()	Fechas de creación del usuario

Nota: Descripción de los datos de la tabla de usuarios.

Tabla 31*Tabla de categorías*

Campo	Tipo de Dato	Restricciones	Descripción
categoria_id	SERIAL	PRIMARY KEY	Identificador único de las categorías

nombre	VARCHAR(100)	NOT NULL	Nombre de la categoría a usar
descripcion	VARCHAR(255)		Detalle de la categoría

Nota: Descripción de los datos en la tabla categorías.

Tabla 32

Tabla de productos

Campo	Tipo de Dato	Restricciones	Descripción
producto_id	SERIAL	PRIMARY KEY	Identificador único de los productos
nombre	VARCHAR(100)	NOT NULL	Nombre del producto
descripcion	TEXT		Descripción del producto
codigo_barra	VARCHAR(100)	UNIQUE, NOT NULL	Código de barras para cada producto
precio	DECIMAL(10,2)	NOT NULL, CHECK (precio >= 0)	Precio asignado al producto
stock_actual	DECIMAL(10,2)	NOT NULL, CHECK (stock_actual >= 0)	Cantidad actual de los productos

stock_minimo	DECIMAL(10,2)	NOT NULL, CHECK (stock_minimo >= 0)	Stock mínimo permitido
fecha_caducidad	DATE	CHECK (fecha_caducidad IS NULL OR fecha_caducidad > CURRENT_DATE)	Fecha de caducidad del producto
categoria_id	INT	FOREIGN KEY (categorias) ON DELETE SET NULL	Relación con la tabla de categorías
fecha_registro	TIMESTAMP	DEFAULT NOW()	Fecha de registro del producto
presentacion_id	INT	FOREIGN KEY (presentaciones) ON DELETE SET NULL	Relación con la tabla de presentaciones

Nota: Descripción de los datos de la tabla de productos.

Arquitectura

Cliente Web:

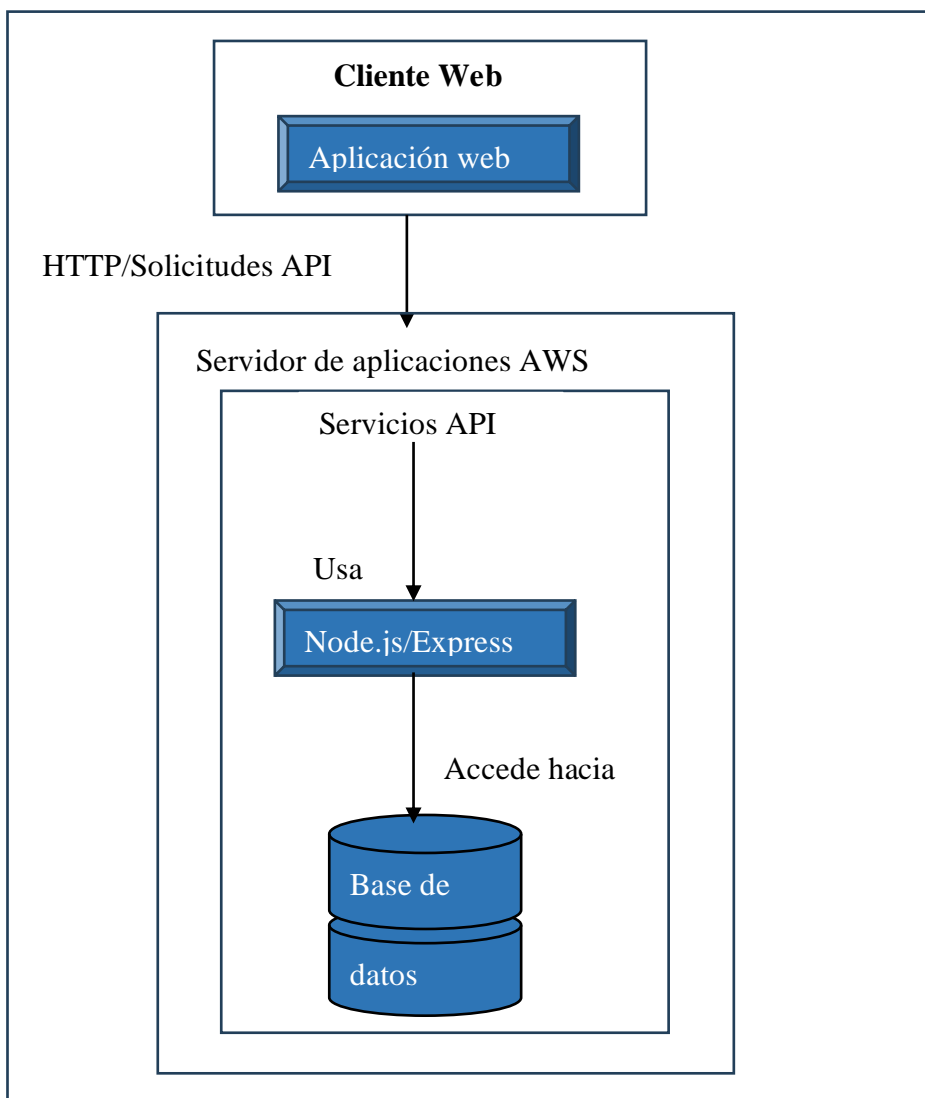
Funcionalidad: La aplicación web proporciona una interfaz para que el usuario tenga adaptabilidad al sistema, cuenta con diseño responsive para que pueda acceder desde su dispositivo sin novedad y está conectada mediante una API REST que permite realizar solicitudes de entrada y salida de información.

Tecnología: Se desarrolló usando javascript con react y node.js, se puede acceder hacia el sitio web con sólo correr los comandos npm run dev desde el IDE Visual Code.

Servidor web: El servidor será desplegado sobre AWS para asegurarnos de que funcione la aplicación en línea, en tiempo real y que las solicitudes desde el Api hacia la aplicación web funcionen correctamente.

Figura 18

Arquitectura de la aplicación



Capítulo IV

Pruebas y resultado

Con el desarrollo de la aplicación SUPEREXPRESS se lograron implementar funcionalidades necesarias para que el usuario final pueda acceder a ellas sin mayor dificultad.

Se logró reducir el tiempo al tener un sistema de inventario de manera centralizado y contar de forma ordenada y mantener controlado el stock de los productos, para asegurar las funcionalidades se realizaron las siguientes pruebas:

Tabla 33

Interfaz al iniciar sesión

Iteración 1.1

Módulo	Iteración	Prerrequisito	Actor	Proceso	Sub-Proceso	Tarea	Resultado alcanzado	Observaciones
Login	1	Interfaz de login creada	Usuario	Creación de página	Obtener información desde la base de datos	Ingresar hacia la aplicación con las credenciales creadas	80%	Se generaba error al encontrar las credenciales en la base de datos

Iteración 1.2

Módulo	Iteración	Prerrequisito	Actor	Proceso	Sub-Proceso	Tarea	Resultado alcanzado	Observaciones
Login	2	Validar el error de inicio y corregirlo	Usuario	Corrección de error en inicio de sesión.	Obtener información almacenada en nuestra base de datos	Corregir el error que permita iniciar sesión con nuestras credenciales ya creadas	100%	Ingreso de manera exitosa.

Nota: Descripción de las iteraciones que se realizaron en la programación de la interfaz de inicio de sesión.

Tabla 34

Interfaz de productos

Iteración 1.1

Módulo	Iteración	Prerrequisito	Actor	Proceso	Sub-Proceso	Tarea	Resultado alcanzado	Observaciones
--------	-----------	---------------	-------	---------	-------------	-------	---------------------	---------------

Productos	1	Página de productos creada	Usuario	Visualizar productos	Mostrar los productos con su QR	Alinear los gráficos y nombres de los productos	90%	Líneas de texto más desiguales en la página.
-----------	---	----------------------------	---------	----------------------	---------------------------------	---	-----	--

Iteración 1.2

Módulo	Iteración	Prerrequisito	Actor	Proceso	Sub-Proceso	Tarea	Resultado alcanzado	Observaciones
Productos	1	Arreglo de página de productos	Usuario	Alineación del contenido de los productos.	Establecer el mismo formato para el contenido	Visualizar los productos en orden en toda la página.	100%	Se visualizan los productos alineados y organizados.

Nota: Descripción de las iteraciones que se realizaron en la programación de la interfaz de productos.

Tabla 35*Interfaz de gestión de usuarios***Iteración 1.1**

Módulo	Iteración	Prerrequisito	Actor	Proceso	Sub-Proceso	Tarea	Resultado alcanzado	Observaciones
Usuarios	1	Roles en el aplicativo	Usuario	Mostrar roles existentes	Visualización de usuario y administrador	Dividir pantallas para mostrar rol de usuario y administrador	70%	No refleja todo el contenido para ambos roles.

Iteración 1.2

Módulo	Iteración	Prerrequisito	Actor	Proceso	Sub-Proceso	Tarea	Resultado alcanzado	Observaciones
---------------	------------------	----------------------	--------------	----------------	--------------------	--------------	----------------------------	----------------------

Usuarios	2	Arreglo de roles en la aplicación	Usuario	Diferenciar los roles en el sistema	Visualización de interfaz para cada rol	Visualización de interfaces de usuario y administrador	100%	Se visualizan las interfaces para ambos usuarios
----------	---	-----------------------------------	---------	-------------------------------------	---	--	------	--

Tabla 36

Interfaz de operaciones

Iteración 1.1

Módulo	Iteración	Prerrequisito	Actor	Proceso	Sub-Proceso	Tarea	Resultado alcanzado	Observaciones
Registro de ventas	1	Página de ventas creada	Usuario	Mostrar productos escogidos	Visualización productos a vender	Mostrar productos escogidos y resumen de venta	60%	Sólo se muestran los productos escogidos al escanear el código de barras.

Iteración 1.2

Módulo	Iteración	Prerrequisito	Actor	Proceso	Sub-Proceso	Tarea	Resultado alcanzado	Observaciones
Registro de ventas	2	Arreglo de visualización de productos vendidos	Usuario	Corregir el resumen de productos vendidos	Mostrar el resumen de venta	Visualización de productos escaneados y resumen de venta de productos	100%	Se visualizan los productos al escanear y también en el resumen de venta con el total a pagar.

Conclusiones

Una vez finalizada el desarrollo de la aplicación se describe las mejoras que se brindará al usuario en su negocio con su sistema de inventario actual:

Control de stock: Al visualizar el stock existente de productos en el negocio el usuario se puede abastecer a tiempo de los productos que estén por terminarse, así como las fechas de los productos que estén próximos a vencer, se mejora en un 80% el control de stock con la implementación de la aplicación.

Optimización del tiempo: Con la creación de la aplicación se facilita una herramienta de fácil navegación y que en el momento que el usuario empiece a usarla le ahorra tiempo haciendo que sea más eficiente y productivo en su negocio, sin duda este aspecto es el más beneficioso ya que se mejora en un 70% hasta adaptarse diariamente al uso de la aplicación y con el uso frecuente este porcentaje subirá hasta 90%.

Reducción de errores: Al tener un sistema automatizado para el inventario de los productos en el negocio, es decir de 10 errores que se tenía actualmente sólo mantienen 5 en la que se evidencia una mejora significativa del 50%.

Control y organización del inventario: Al contar con una base de datos que almacene los egresos e ingresos con sus detalles como las fechas de expiración, unidades de cada producto, etc., falta el punto en la abreviatura el usuario cuenta con información en tiempo real y le permite reducir errores de manera significativa y también para el personal que le apoya en la tienda, el porcentaje de beneficio es de un 80% en control de los productos.

Satisfacción del cliente: Al cliente encontrar todos los productos en el negocio de confianza se asegura la preferencia de que va a acudir con mayor frecuencia hacia la tienda y beneficiándose mutuamente con el propietario del negocio.

Recomendaciones

Se recomienda que la universidad promueva a los estudiantes a desarrollar aplicaciones que sean de beneficio mutuo y sobre todo enfocadas a emprendedores de pequeños y medianos negocios e identificar sus necesidades para el desarrollo de las aplicaciones.

En las ferias tecnológicas que suelen haber en la universidad sería la mejor oportunidad para que los emprendedores tomen en cuenta que los estudiantes de la PUCE estamos capacitados y somos aptos para desarrollar las aplicaciones que necesiten y así mismo nos brinden la oportunidad para desarrollar nuestras prácticas recibidas en clases y con esto nos podemos abrir puertas frente a otras empresas para ser considerados y que se nos brinde oportunidades laborales haciendo quedar en alto a las carreras de la tecnología de PUCETEC.

Referencias bibliográficas

- Schwaber, K., & Sutherland, J. (2020). *The Scrum Guide: The Definitive Guide to Scrum: The Rules of the Game*. Scrum.org.
- Rubin, K. S. (2012). *Essential Scrum: A Practical Guide to the Most Popular Agile Process*. Addison-Wesley.
- Pichler, R. (2010). *Agile Product Management with Scrum: Creating Products that Customers Love*. Addison-Wesley.
- Cohn, M. (2009). *Succeeding with Agile: Software Development Using Scrum*. Addison-Wesley.
- Definición de Hecho. (2018). *Scrum: La Definición de Hecho (DoD)*.
- Mell, P., & Grance, T. (2011). *The NIST Definition of Cloud Computing*. National Institute of Standards and Technology (NIST).
- Sommerville, I. (2011). *Software Engineering* (9th ed.). Pearson Education.
- Rob, P., & Coronel, C. (2019). *Database Systems: Design, Implementation, & Management* (13th ed.). Cengage Learning.
- ¿Qué es Scrum? (s. f.-b). Scrum.org. <https://www.scrum.org/resources/blog/que-es-scrum>
- Fierro, E. (s. f.). *Eduardo Fierro - Web designer - Fullstack Developer*.
<https://eduardofierro.pro/blog/javascript-en-el-backend-todo-lo-que-necesitas-saber-para-empezar/>
- Introducción a Express/Node - Aprende desarrollo web | MDN*. (s. f.). MDN Web Docs. https://developer.mozilla.org/es/docs/Learn_web_development/Extensions/Server-side/Express_Nodejs/Introduction
- Atlassian. (s. f.-b). *What is scrum and how to get started*. <https://www.atlassian.com/agile/scrum>

