



UNIVERSIDAD TÉCNICA DE AMBATO
FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E
INDUSTRIAL
CARRERA DE INGENIERÍA EN SISTEMAS
COMPUTACIONALES E INFORMÁTICOS

TEMA:

IMPLEMENTACIÓN DE UN MECANISMO DE GEOREFERENCIACIÓN
PARA LA CADENA DE DISTRIBUCIÓN DE UNA INICIATIVA DE
COMERCIO ELECTRÓNICO UTILIZANDO SOFTWARE LIBRE

Trabajo de Titulación. Modalidad: Proyecto de Investigación, presentado previo
a la obtención del título de Ingeniero en Sistemas Computacionales e
Informáticos

ÁREA: Software

LÍNEA DE INVESTIGACIÓN: Desarrollo de software

AUTOR: Sebastián Ricardo Mayorga Naranjo

TUTOR: Ing. PhD. Félix Oscar Fernández Peña, MsC.

Ambato - Ecuador

marzo, 2023

APROBACIÓN DEL TUTOR

En calidad de tutor del Trabajo de Titulación con el Tema: IMPLEMENTACIÓN DE UN MECANISMO DE GEOREFERENCIACIÓN PARA LA CADENA DE DISTRIBUCIÓN DE UNA INICIATIVA DE COMERCIO ELECTRÓNICO UTILIZANDO SOFTWARE LIBRE, desarrollado bajo la modalidad Proyecto de Investigación por el señor Sebastián Ricardo Mayorga Naranjo, estudiante de la Carrera de Ingeniería en Sistemas Computacionales e Informáticos, de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, de la Universidad Técnica de Ambato, me permito indicar que el estudiante ha sido tutorado durante todo el desarrollo del trabajo hasta su conclusión, de acuerdo a lo dispuesto en el Artículo 15 del Reglamento para obtener el Título de Tercer Nivel, de Grado de la Universidad Técnica de Ambato, y el numeral 7.4 del respectivo instructivo.

Ambato, marzo 2023

Ing. PhD. Félix Oscar Fernández Peña, MsC.

TUTOR

AUTORÍA

El presente Proyecto de Investigación titulado: IMPLEMENTACIÓN DE UN MECANISMO DE GEOREFERENCIACIÓN PARA LA CADENA DE DISTRIBUCIÓN DE UNA INICIATIVA DE COMERCIO ELECTRÓNICO UTILIZANDO SOFTWARE LIBRE es absolutamente original, auténtico y personal. En tal virtud, el contenido, efectos legales y académicos que se desprenden del mismo son de exclusiva responsabilidad del autor.

Ambato, marzo 2023

Sebastián Ricardo Mayorga Naranjo

CC: 1804629226

AUTOR

DERECHOS DE AUTOR

Autorizo a la Universidad Técnica de Ambato, para que haga uso de este Trabajo de Titulación como un documento disponible para la lectura, consulta y procesos de investigación.

Cedo los derechos de mi Trabajo de Titulación, con fines de difusión pública. Autorizo además su reproducción dentro de las regulaciones de la Universidad Técnica de Ambato.

Ambato, marzo 2023

Sebastián Ricardo Mayorga Naranjo

CC: 1804629226

AUTOR

APROBACIÓN DEL TRIBUNAL DE GRADO

En calidad de par calificador del Informe Final del Trabajo de Titulación presentado por el señor Sebastián Ricardo Mayorga Naranjo, estudiante de la Carrera de Ingeniería en Sistemas, Electrónica e Industrial, bajo la Modalidad Proyecto de Investigación, titulado IMPLEMENTACIÓN DE UN MECANISMO DE GEOREFERENCIACIÓN PARA LA CADENA DE DISTRIBUCIÓN DE UNA INICIATIVA DE COMERCIO ELECTRÓNICO UTILIZANDO SOFTWARE LIBRE, nos permitimos informar que el trabajo ha sido revisado y calificado de acuerdo al Artículo 17 del Reglamento para obtener el Título de Tercer Nivel, de Grado de la Universidad Técnica de Ambato, y al numeral 7.6 del respectivo instructivo. Para cuya constancia suscribimos, conjuntamente con la señora Presidenta del Tribunal.

Ambato, marzo de 2023

Ing. Pilar Urrutia, Mg.

PRESIDENTE DEL TRIBUNAL

Ing. Rubén Nogales, Mg.
PROFESOR CALIFICADOR

Ing. David Guevara, Mg.
PROFESOR CALIFICADOR

DEDICATORIA

El presente trabajo de investigación lo dedico en primer lugar a mis padres Elsa Naranjo y Mario Mayorga quienes han sido participes en mi formación como una persona responsable, además me han brindado el apoyo necesario durante mi vida estudiantil y profesional.

A mis hermanas Carolina y Beatriz que me han brindado apoyo incondicional durante todo este proceso que me ha servido como motivación para seguir adelante.

Por último a mis compañeros y amigos quienes han sido parte de este proceso y han compartido gratas experiencias conmigo.

Sebastián Ricardo Mayorga Naranjo

AGRADECIMIENTO

A toda mi familia quien ha estado constantemente apoyándome en lo que he necesitado y siempre han estado presentes en mis buenos y malos momentos.

Al Ing. Félix Fernández por los conocimientos brindados y el tiempo dedicado como tutor en el desarrollo del presente proyecto de investigación.

A todos los docentes que me han inculcado los conocimientos necesarios para poder salir adelante en el ámbito personal y profesional.

Finalmente, a mis amigos por aquellos trabajos realizados en conjunto y los buenos momentos vividos en el transcurso de la universidad.

Sebastián Ricardo Mayorga Naranjo

ÍNDICE

APROBACIÓN DEL TUTOR	ii
AUTORÍA	iii
DERECHOS DE AUTOR	iv
APROBACIÓN COMISIÓN CALIFICADORA	v
Dedicatoria	vi
Agradecimiento	vii
CAPÍTULO I MARCO TEÓRICO	1
1.1 Tema de investigación	1
1.2 Antecedentes investigativos	1
1.2.1 Contextualización del problema	2
1.2.1.1 Fundamentación teórica	4
1.3 Objetivos	14
1.3.1 Objetivo general	14
1.3.2 Objetivos específicos	14
CAPÍTULO II METODOLOGÍA	15
2.1 Materiales	15
2.1.1 Humanos	15
2.1.2 Institucionales	15
2.1.3 Otros	15
2.1.4 Económicos	16
2.2 Métodos	16
2.2.1 Modalidad de la investigación	16
2.2.2 Población y muestra	17
2.2.3 Recolección de la información	17
2.2.4 Metodología seleccionada	18

CAPÍTULO III RESULTADOS Y DISCUSIÓN	19
3.1 Análisis y discusión de resultados	19
3.1.1 Validación de encuesta	19
3.1.2 Resultados de encuesta	20
3.1.3 Determinación del framework	25
3.1.4 Determinación del gestor de contenido	27
3.1.5 Determinación del mecanismo de georeferenciación	28
3.2 Desarrollo de la propuesta	30
3.2.1 Fase 1: Planificación	30
3.2.1.1 Levantamiento de la información	30
3.2.1.2 Definición de roles	30
3.2.1.3 Historias de usuario	31
3.2.1.4 Tareas	37
3.2.1.5 Valoración de historias de usuario	46
3.2.1.6 Estimación de historias de usuario	46
3.2.1.7 Plan de entrega	49
3.2.2 Fase 2: Diseño	49
3.2.2.1 Estructura del proyecto	49
3.2.2.2 Herramientas utilizadas	50
3.2.2.3 Esquema del proyecto	51
3.2.2.4 Diseño de la base de datos	52
3.2.2.5 Tarjetas CRC	54
3.2.2.6 Prototipos de interfaces	55
3.2.2.7 Diseño sistema web	60
3.2.3 Fase 3: Codificación	70
3.2.3.1 Comandos para crear el proyecto	70
3.2.3.2 Modificación de interfaz en wordpress	70
3.2.3.3 Conexión a base de datos	71
3.2.4 Fase 4: Pruebas	73
3.2.4.1 Pruebas de aceptación	73
3.2.4.2 Evaluación heurística con los principios de Jakob Nielsen	79
CAPÍTULO IV CONCLUSIONES Y RECOMENDACIONES	81
4.1 Conclusiones	81
4.2 Recomendaciones	81
Bibliografía	83

ÍNDICE DE TABLAS

2.1	Recursos económicos	16
2.2	Recolección de información	17
3.1	Rangos de confiabilidad para el coeficiente alfa de Cronbach	20
3.2	Resultados de encuesta - Pregunta 1	20
3.3	Resultados de encuesta - Pregunta 2	21
3.4	Resultados de encuesta - Pregunta 3	21
3.5	Resultados de encuesta - Pregunta 4	21
3.6	Resultados de encuesta - Pregunta 5	22
3.7	Resultados de encuesta - Pregunta 6	23
3.8	Resultados de encuesta - Pregunta 7	23
3.9	Resultados de encuesta - Pregunta 8	23
3.10	Resultados de encuesta - Pregunta 9	24
3.11	Resultados de encuesta - Pregunta 10	24
3.12	Resultados de encuesta - Pregunta 11	25
3.13	Comparación de frameworks	26
3.14	Comparación gestores de contenido	28
3.15	Comparación mecanismos de georeferenciación	29
3.16	Definición de roles	30
3.17	Plantilla de historia de usuario	31
3.18	Historia de usuario: Definir la estructura del sistema	32
3.19	Historia de usuario: Comercio electrónico	32
3.20	Historia de usuario: Estructura de la base de datos	33
3.21	Historia de usuario: Productos e información georeferenciada	33
3.22	Historia de usuario: Registro en el sistema	34
3.23	Historia de usuario: Acceso al sistema	34
3.24	Historia de usuario: Administrar usuarios	35
3.25	Historia de usuario: Gestión de puntos de distribución	35
3.26	Historia de usuario: Gestión de personal por producto	36
3.27	Historia de usuario: Asignar ubicaciones por producto	36
3.28	Historia de usuario: Validar ubicaciones	37

3.29	Historia de usuario: Designar puntos de distribución	37
3.30	Plantilla de tarea de historia de usuario	38
3.31	Tarea: Análisis del sistema	39
3.32	Tarea: Creación del comercio electrónico	39
3.33	Tarea: Análisis de la base de datos	39
3.34	Tarea: Diseño de interfaz de inicio del sistema	40
3.35	Tarea: Diseño de interfaz del detalle de un producto seleccionado .	40
3.36	Tarea: Elaboración de métodos para obtener los productos disponibles	40
3.37	Tarea: Diseño de la interfaz de registro de usuario	41
3.38	Tarea: Elaboración de métodos para registrar usuarios	41
3.39	Tarea: Diseño de la interfaz de inicio de sesión	41
3.40	Tarea: Elaboración de métodos para el inicio de sesión	42
3.41	Tarea: Diseño de interfaz para la gestión de usuarios	42
3.42	Tarea: Elaboración de métodos para gestionar usuarios registrados en el sistema	42
3.43	Tarea: Diseño de interfaz para gestionar puntos de distribución . .	43
3.44	Tarea: Elaboración de métodos para gestionar puntos de distribución	43
3.45	Tarea: Diseño de interfaz para gestionar personal por producto . .	43
3.46	Tarea: Elaboración de métodos para gestionar personal por producto	44
3.47	Tarea: Diseño de interfaz para gestionar ubicaciones de un producto	44
3.48	Tarea: Elaboración de métodos para gestionar ubicaciones de un producto	44
3.49	Tarea: Diseño de interfaz para validar ubicaciones de productos .	45
3.50	Tarea: Elaboración de métodos para validar ubicaciones de productos	45
3.51	Tarea: Diseño de la interfaz para gestionar puntos de distribución de un producto	46
3.52	Tarea: Elaboración de métodos para designar puntos de distribu- ción de un producto	46
3.53	Estimación de historias de usuario	47
3.54	Resumen iteración 1	47
3.55	Resumen iteración 2	48
3.56	Resumen iteración 3	48
3.57	Resumen iteración 1	48
3.58	Plan de entrega	49
3.59	Tarjeta CRC - Usuario	54
3.60	Tarjeta CRC - Producto	55

3.61 Tarjeta CRC - Sucursal	55
3.62 Tarjeta CRC - Ubicación	55
3.63 Prueba de aceptación 1	73
3.64 Prueba de aceptación 2	74
3.65 Prueba de aceptación 3	75
3.66 Prueba de aceptación 4	75
3.67 Prueba de aceptación 5	76
3.68 Prueba de aceptación 6	77
3.69 Prueba de aceptación 7	77
3.70 Prueba de aceptación 8	78
3.71 Prueba de aceptación 9	78
3.72 Prueba de aceptación 10	79

ÍNDICE DE FIGURAS

1.1	Arquitectura de 3 niveles [1].	6
1.2	Metodología scrum [2].	8
1.3	Fases de la metodología XP [2].	9
1.4	Fases de la metodología kanban [3].	10
1.5	Representación MVC Angular [4].	11
1.6	Diseño de API REST [5].	13
3.1	Arquitectura del proyecto woomap	50
3.2	Estructura del frontend	51
3.3	Estructura del backend	52
3.4	Esquema de base de datos del sistema web.	53
3.5	Tablas de base de datos de la página web.	54
3.6	Prototipo de interfaz de login	56
3.7	Prototipo de interfaz de registro de usuario	56
3.8	Prototipo de interfaz de inicio	57
3.9	Prototipo de interfaz de gestión de personal	58
3.10	Prototipo de interfaz para administrar puntos de distribución . . .	58
3.11	Prototipo de interfaz de gestión de usuarios	59
3.12	Prototipo de interfaz para administrar puntos de distribución por producto	59
3.13	Prototipo de interfaz para validar ubicaciones	60
3.14	Prototipo de interfaz para asignar ubicaciones a un producto . . .	60
3.15	Pantalla de inicio de tienda virtual	61
3.16	Pantalla portafolio de productos	61
3.17	Pantalla detalle de producto	62
3.18	Pantalla de administración de wordpress	62
3.19	Pantalla de administración de wordpress	63
3.20	Pantalla de inicio del sistema web	64
3.21	Pantalla de registro de nuevo usuario	64
3.22	Pantalla de ingreso al sistema	65
3.23	Pantalla para administrar usuarios	65

3.24	Pantalla para administrar los puntos de distribución	66
3.25	Pantalla para administrar el personal por producto	67
3.26	Pantalla para asignar ubicaciones de un producto	68
3.27	Pantalla para validar ubicaciones registradas por un proveedor . .	69
3.28	Pantalla para designar puntos de distribución a un producto . . .	69
3.29	Comando para crear el proyecto angular	70
3.30	Comando para crear una aplicación de nodejs	70
3.31	Comando para instalar módulos en la aplicación de node js	70
3.32	Código para agregar un botón a la interfaz de un tema de wordpress	70
3.33	Conexión a base de datos de woomap	71
3.34	Conexión a base de datos de wordpress	71
3.35	Método de inicio de sesión	72
3.36	Resultados de la evaluación heurística	80
1.1	Método para autenticar un usuario	88
1.2	Método para registrar un nuevo usuario	89
1.3	Método para verificar token de inicio de sesión	90
1.4	Método para obtener los usuarios del sistema	90
1.5	Método para actualizar un usuario del sistema	91
1.6	Método para eliminar un usuario del sistema	92
1.7	Método para obtener los usuarios del sistema	92
1.8	Método para buscar un producto	93
1.9	Método para crear una sucursal	94
1.10	Método para asignar una sucursal a un producto	95
1.11	Método para insertar ubicaciones del recorrido de un proveedor .	96
1.12	Método para obtener las ubicaciones validadas que ha recorrido un proveedor	97
1.13	Instanciación de nodemailer	97
1.14	Método para enviar correo de gmail	98
1.15	Métodos para guardar y obtener el token de inicio de sesión . . .	98
1.16	Métodos para guardar y obtener el usuario logueado	99
1.17	Guard que redirecciona a la ventana de login	99
1.18	Guard que redirecciona a la ventana de home	99
1.19	Métodos de inicio de sesión y registro de un usuario nuevo	100
1.20	Métodos para abrir un mensaje emergente de aprobación y error .	100
1.21	Declaración de contenedor para insertar mapa	101
1.22	Métodos para instanciar un mapa de leaflet	101
1.23	Instanciación de librería antpath y uso en un mapa leaflet instanciado	102

1.24	Método bindPopup	102
1.25	Métodos para insertar una o varias ubicaciones	103
1.26	Método que obtiene ubicaciones registradas	104
1.27	Método invocado para abrir ventana flotante de edición	105
1.28	Método que actualiza el estado de una ubicación registrada	105
1.29	Declaración componentes angular material	106
1.30	Comprobación de inicio de sesión de un usuario	106

RESUMEN EJECUTIVO

La información georeferenciada que posee un sistema web de comercio electrónico el cual ha sido implementado a través de un CMS como es el caso de wordpress, se limita al uso de plugins creados por usuarios externos, estos plugins tienen múltiples características y la mayoría de ellos brindan todo su potencial solo al pagar una suscripción, es por ello que el presente trabajo de investigación titulado “Implementación de un mecanismo de georeferenciación para la cadena de distribución de una iniciativa de comercio electrónico utilizando software libre”, tiene la finalidad de demostrar que a través de un sistema web externo creado como prototipo se puede exponer información georeferenciada de un comercio electrónico que este implementado utilizando un CMS.

El proyecto ha sido desarrollado utilizando el framework angular v.13.3.1, que permite construir aplicaciones de tipo SPA (Single Page Application), adicional a esto para el backend que proporcionará los servicios necesarios para el sistema web se ha utilizado la tecnología de node.js v.16.14.2 en conjunto de sequelize que es un ORM (Object Relational Mapping), el cual a través de métodos ya creados permite modelar una base de datos y utilizar los métodos select, create, delete y update sin necesidad de implementar sentencias sql; para la creación de la tienda virtual se utilizó wordpress en conjunto con el plugin de woocommerce que permite implementar una tienda virtual de una manera profesional.

La metodología utilizada para el desarrollo del presente proyecto fue la metodología XP (Xtreme Programming), la cual es una metodología de desarrollo ágil que permite el desarrollo de aplicaciones de manera rápida a través de historias de usuario que son ejecutadas en diferentes iteraciones permitiendo así la fácil implementación de nuevas funcionalidades y correcciones al sistema.

Palabras clave: Wordpress, Angular, Node.js, Typescript, Woocommerce, Sequelize, ORM, MySQL, georeferenciación, leaflet.

ABSTRACT

The georeferenced information that an electronic commerce web system has, which has been implemented through a CMS such as wordpress, is limited to the use of plugins created by external users, these plugins have multiple features and most of them provide its full potential only by paying a subscription, which is why this research work entitled "Implementation of a georeferencing mechanism for the distribution chain of an electronic commerce initiative using free software", has the purpose of demonstrating that through From an external web system created as a prototype, georeferenced information can be exposed from an electronic commerce that is implemented using a CMS.

The project has been developed using the angular v.13.3.1 framework, which allows building SPA (Single Page Application) type applications, in addition to this, for the backend that will provide the necessary services for the web system, node technology has been used. .js v.16.14.2 together with sequelize, which is an ORM (Object Relational Mapping), which through already created methods allows modeling a database and using the select, create, delete and update methods without the need to implement sql statements; For the creation of the virtual store, wordpress was used in conjunction with the woocommerce plugin that allows to implement a virtual store in a professional way.

The methodology used for the development of this project was the XP (Xtreme Programming) methodology, which is an agile development methodology that allows the development of applications quickly through user stories that are executed in different iterations, thus allowing the easy implementation of new features and fixes to the system.

Keywords: Wordpress, Angular, Node.js, Typescript, Woocommerce, Sequelize, ORM, MySql, georeferencing, leaflet.

CAPÍTULO I

MARCO TEÓRICO

1.1. Tema de investigación

IMPLEMENTACIÓN DE UN MECANISMO DE GEOREFERENCIACIÓN PARA LA CADENA DE DISTRIBUCIÓN DE UNA INICIATIVA DE COMERCIO ELECTRÓNICO UTILIZANDO SOFTWARE LIBRE

1.2. Antecedentes investigativos

Revisando la información bibliográfica de diferentes repositorios de universidades del Ecuador se encontró los siguientes trabajos referentes al tema de investigación que se realizó, trabajos que nos sirvieron de apoyo para obtener información relevante para la investigación en desarrollo.

Según [6] indica que la creación de la aplicación móvil multiplataforma cumple con la estructura de tabs, mientras que su funcionamiento se basa en el modelo cliente servidor ya que para la gestión de la información la app hace uso de un API REST usando NodeJS con la versión 12.18.2 LTS, la cual se encarga de proporcionar los servicios a la aplicación conectándose a la base de datos de MongoDB. Adicional a esto añade que para la autenticación de usuarios implementa firebase y para la visualización de mapas 3D y 2D hace uso de los servicios y librerías ofrecidos por Mapbox y Leaflet respectivamente.

En [7] señala que para el desarrollo del proyecto se utiliza la metodología XP de desarrollo ágil, cumpliendo el modelo de cliente-servidor para su funcionamiento, separando el front-end del back-end y desarrollándolos en distintas tecnologías, tales como Ionic usando su framework y firebase, respectivamente. Utiliza los lenguajes de programación TypeScript y Angular aplicando la arquitectura MVC (Modelo, Vista, Controlador) mientras que para la implementación de mapas offline utiliza el SDK de Mapbox y el plugin de geolocalización y network de ionic.

Según [8] menciona que después del análisis de las distintas metodologías y tecnologías de desarrollo, la metodología seleccionada fue XP mientras que las tecno-

logías utilizadas fueron: Ionic con la utilización de su framework o SDK, firebase, tanto para el módulo de inicio de sesión como para la administración de datos con el servicio de firestore y el SDK de la biblioteca de Mapbox para la implementación de mapas offline además de contar con varios estilos, fuentes y marcadores personalizados que se representan en un mapa. Adicional a esto añade que la aplicación está basada en el modelo cliente-servidor siguiendo la estructura MVC y desarrollada en el lenguaje de programación TypeScript con NodeJS versión 14.17.1.

En [9] indica que Android Studio tiene fácil conectividad con el servidor de base de datos firebase, lo que da como resultado un menor consumo de datos durante las peticiones, además de poseer gran compatibilidad con las APIs Google Maps y Geofire, para la incorporación de mapas reales y la realización de acciones relacionadas a cambios de posición sobre los mapas. Razón por la cual implementó estas tecnologías para el desarrollo de la aplicación móvil utilizando el lenguaje java, además de desarrollar el proyecto con la metodología Mobile-D, ya que permite el desarrollo ágil de aplicaciones móviles en dispositivos pequeños y la detección rápida de bugs o fallas, garantizando el producto final.

1.2.1. Contextualización del problema

A nivel mundial la economía se ha visto severamente afectada por el Covid-19, en algunos países se ha interrumpido la producción, ha decrecido la demanda de bienes y servicios, ha puesto en suspenso las cadenas de suministro, obligando a las empresas a suspender o reducir sus operaciones [10].

Según el Instituto Latinoamericano de Comercio Electrónico (ILCE), Ecuador tiene un gran potencial para aumentar las transacciones a través de los sistemas de negocios electrónicos [11].

En Ecuador, de acuerdo a un estudio realizado por la Cámara Ecuatoriana de Comercio Electrónico (CECE) mediante una encuesta en línea en mayo del 2020 a 126 empresas, se pudo determinar que el 56 % de estas indicaron que sus ventas disminuyeron[12], además en un estudio realizado por el ministerio de producción, comercio exterior, inversiones y pesca se indica que las empresas han tenido una reducción del 42 % en las ventas[13], esto se debe a la aparición de la pandemia y a que la mayoría de microempresas cuentan con una aplicación web que no abastece con las necesidades de los clientes, lo que produce que dichas microem-

presas dedicadas a la venta de artefactos no puedan generar suficientes ingresos a lo largo de un periodo de tiempo, siendo esto desfavorable para los mismos. En cambio, las empresas que generan más ganancias y obtienen mayor rentabilidad, por lo general poseen un correcto sistema de ventas en línea, lo que les permite ofrecer sus productos a través del internet, mismo que engloba una gran cantidad de clientes potenciales interesados por diferentes categorías de productos, y que constantemente se encuentran buscando nuevas promociones y páginas en donde puedan adquirir un producto o servicio a un precio razonable [14].

En la actualidad tanto empresas como negocios familiares han buscado formas innovadoras de llegar a sus clientes y facilitar las compras, haciendo que estas sean ágiles, cómodas y rápidas, desde cualquier dispositivo electrónico en cualquier parte del mundo [15].

En la zona central del País principalmente en la provincia de Tungurahua la publicidad se mantiene en constante crecimiento, por esta razón los medios de comunicación social han ido evolucionando y son los encargados de influir en las personas a través de la información brindada. Las estrategias de publicidad que utilizan medios de comunicación masivos poseen una poderosa fuerza de persuasión, permitiendo modelar actitudes y comportamientos de la gente, por lo que en la provincia de Tungurahua a través de la publicidad se ha incrementado el nivel de producción y ventas, logrando de esta manera que el sector industrial y comercial crezca constantemente [16].

En la provincia de Tungurahua el 72.9% de un grupo de 339 PYMES implementaron marketing digital durante la pandemia. Un total del 97.40% de las PYMES optaron por nuevos canales de comercialización para vender en línea [17]. Por medio de estas acciones se promueve la interacción de sus productos con los clientes a través de catálogos en línea además de la implementación de pagos por plataformas digitales. Estos medios ayudan en el proceso de comunicación, registro, optimización de recursos entre otros, para llegar a sus clientes [18].

En la Provincia de Tungurahua y en la ciudad de Ambato, las empresas generalmente olvidan que el mercadeo es esencialmente la permanente búsqueda y uso de acciones estratégicas para incentivar y/o facilitar la demanda de un producto o servicio (a través de cualquiera de las herramientas mercadotécnicas; reposicionamiento, nuevas presentaciones, cambio de empaque, promociones, etc.), lo que

lleva frecuentemente a acomodarse con un determinado volumen de negocios o a conformarse con el crecimiento mentiroso en dólares resultante de la inflación [19].

La ciudad de Ambato en los últimos años ha tenido un alto crecimiento económico, lo que ha obligado a las empresas comercializadoras a buscar nuevos métodos para brindar sus servicios y exponer sus productos a la comunidad; es por esto que adoptan el uso de la tecnología para su uso cotidiano.

El comercio es una actividad ancestral que con el tiempo ha evolucionado para adaptarse a las necesidades de los clientes en un mundo cada vez más globalizado [20].

La integración del sistema de comercio electrónico con el sistema de gestión comercial es la mejor forma para poder aprovechar los beneficios de aumento de productividad y reducción de costos. Esta integración evita malgastar tiempo, recursos y los errores que se pueden derivar de estas duplicaciones [21].

Las Pymes que no tienen tiendas virtuales se ven afectadas económicamente, denotando problemas como reducción en ventas, mercadería sobrante en bodega, un desbalance económico, reducción de presupuesto para importación de mercadería.

1.2.1.1. Fundamentación teórica

Sistema gestor de contenido (CMS) Conocidos en inglés como Content Management System, es un sistema que sirve para la publicación rápida y sencilla de sitios web, se caracteriza por separar la interfaz gráfica de usuario del contenido (base de datos), muchos de estos sistemas son de libre distribución y cuentan con secciones especializadas en noticias, foros, descargas, preguntas frecuentes, encuestas, galerías de imágenes y control de usuarios por medio del uso de privilegios, contraseñas, manejo de grupos y permisos de lectura/escritura. Los CMS en su mayoría son desarrollados con software libre utilizando herramientas como PHP, MYSQL y Apache, de igual forma existe una gran cantidad de desarrolladores de CMS entre los más importantes se pueden mencionar a Mambo, PosNuke, PHPNuke y ASPNuke que es una alternativa para plataformas de software comercial [22].

Georeferenciación Es la técnica de posicionamiento espacial que posee una entidad en una localización geográfica única y definida en un sistema de

coordenadas. Es una operación habitual dentro de los Sistemas de Información Geográfica (SIG) tanto para imágenes de mapa de píxeles como para objetos vectoriales, además la georeferenciación es algo esencial en el análisis de datos geoespaciales, debido a que es la base para obtener la correcta localización de información en el mapa y por ende de la adecuada fusión y comparación de datos procedentes de sensores ubicados en diferentes locaciones tanto espaciales como temporales [23].

Aplicación web Son herramientas que utilizan los usuarios para acceder a un servidor web a través de la red y mediante un navegador determinado, es decir una aplicación web se puede definir como una aplicación que se accede mediante la web a través de una red que este conectada por internet o localmente, esta aplicación no necesita instalarse en el ordenador ya que solo se utiliza un navegador para ser ejecutada, utiliza por defecto el lenguaje de marcas de hipertexto o también conocido por HTML [24].

Ventajas [1]:

- No necesita instalar software especial.
- Bajo coste de actualizar de equipos con una nueva versión.
- Acceso a la última versión de la aplicación.
- La información se encuentra centralizada por lo cual facilita el acceso a la misma.
- Facilidad para establecer y llevar controles de una política de copias de seguridad centralizada.
- Movilidad.

Las aplicaciones web se basan en la arquitectura cliente-servidor donde las tareas y capacidad de proceso se reparten entre proveedores de servicios (servidores) y los demandantes (clientes), lo cual brinda ventajas como centralización del control, escalabilidad, fácil mantenimiento y tecnologías. A su vez, las aplicaciones web también pueden implementar la arquitectura de tres niveles, en la cual existe un nivel intermedio dando como resultado una arquitectura compartida por cliente, servidor de aplicaciones y servidor de datos permitiendo mayor y mejor grado de flexibilidad, seguridad y rendimiento [24].

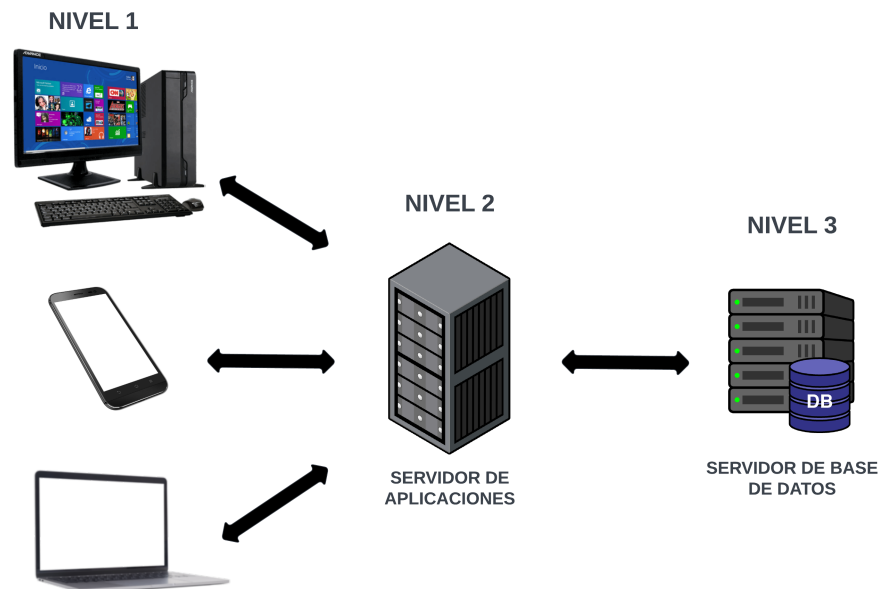


Figura 1.1: Arquitectura de 3 niveles [1].
Fuente: Elaborado por el investigador

Metodologías de desarrollo de software La metodología de desarrollo puede definirse como una organización sistemática para el ciclo de vida del sistema y cada una de sus partes o también como un conjunto de técnicas, procedimientos y herramientas implementadas para controlar y planificar el desarrollo de un nuevo software por lo cual su objetivo principal es organizar todos los procesos para el desarrollo del software por medio de un método sistemático, el cual contempla el establecimiento de requisitos, tiempo, costo, cambios y errores del sistema con la finalidad de proveer un producto de calidad [2].

Una metodología de desarrollo debe cubrir varios aspectos tales como [2]:

- Proceso de ciclo de vida completo.
- Conjunto de conceptos y modelos.
- Descripción completa del software a desarrollar.
- Conjunto de técnicas aprobadas.
- Identificación de roles organizacionales.

- Guías para la gestión de proyecto y aseguramiento de la calidad.

Clasificación:

1. Metodologías tradicionales o pesadas

Son aquellas que se caracterizan por centrarse en la planificación y control del proyecto, así como en la especificación de requisitos y modelado, este tipo de metodología exige abundante y exhaustiva documentación de la que posteriormente se obtienen resultados esperados debido a que esta se basa en la planificación detallada desde la fase inicial del proyecto [25].

2. Metodologías ágiles

Este tipo de metodología se dirige a equipo de trabajo pequeños y se centra en el aspecto humano relacionados con el trabajo en equipo y la participación del cliente durante el proceso de desarrollo, se denominan ágiles ya que están orientadas a la generación de código con ciclos cortos de desarrollo.

Las metodologías ágiles se caracterizan por su código fuente que es adaptable y orientado a los usuarios basando así su éxito en el equipo de trabajo, por lo cual permiten afrontar el problema de los requerimientos inciertos o variables de las entregas tempranas que tienen por objetivo satisfacer los requerimientos básicos del cliente [25].

Principales modelos de metodologías ágiles:

Metodología scrum

El modelo Scrum se fundamenta en el ciclo de vida iterativo e incremental en el que se va liberando el producto por partes de forma periódica e iterativamente donde cada entrega es un incremento de funcionalidad respecto a la anterior. Se caracteriza por definir roles dentro de la creación de equipos auto organizadas y practicas que aportan dinamismo a la ejecución del proyecto [26].

Scrum se organiza por sprints, los cuales corresponden a un periodo de tiempo de cuatro semanas o un mes durante y contienen los requisitos de funcionalidades o historial de usuarios, en cada sprint se realizan avances del sistema y reuniones donde se tratan temas como la planificación del sprint, scrum diarios, revisiones y retrospectivas [26].

Roles de Scrum [26]:

- **Scrum master:** líder de equipo que tiene como función primordial velar por la integración del equipo creando un ambiente adecuado para completar el desarrollo del producto de forma satisfactoria.

- **Product Owner:** es el encargado de integrar los requisitos funcionales y no funcionales e internos o externos del cliente por medio de la comunicación clara con el equipo Scrum.
- **Equipo Scrum:** son los encargados del desarrollo del software en base a los requerimientos del negocio.

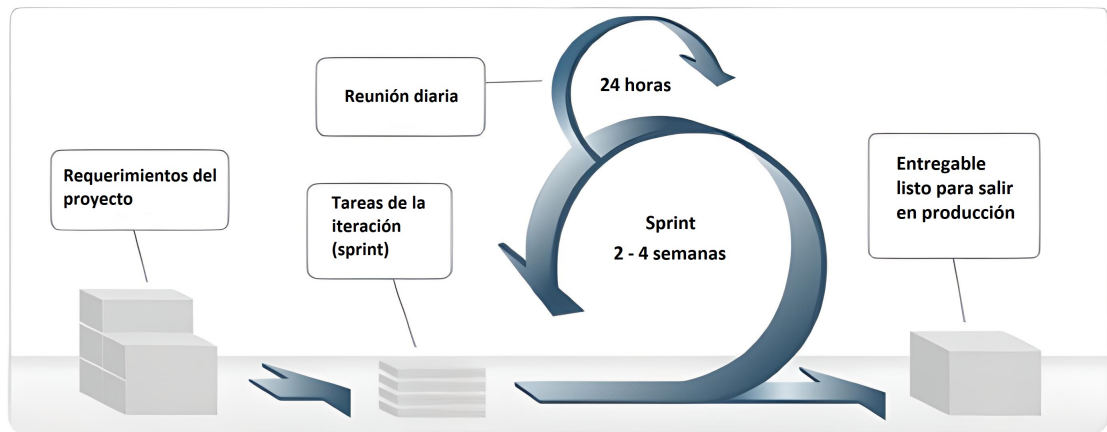


Figura 1.2: Metodología scrum [2].

Metodología XP

Extreme Programming o Programación Extrema esta organizada por fases y se basa en la simplicidad, los cambios incrementales, la calidad y el feedback ya que su objetivo es responder a las necesidades del cliente y comprometer al equipo por medio de roles y fases [26].

Roles de XP [26]:

- Programador.
- Cliente.
- Responsable de las pruebas o tester.
- Encargado del seguimiento o tracker.
- Entrenador o coach.
- Consultor y gestor o the big boss.

Fases de XP [26]:

- Fase de exploración: comprende un par de semanas y se establece el alcance general del proyecto, el cliente define los requerimientos por medio de historias de usuario y los programadores evalúan el tiempo de desarrollo.

- Fase de planificación: escenario de corta duración donde cliente y el equipo de trabajo definen el orden en el que se debe desarrollar y entregas de las historias de usuario.
- Fase de iteraciones: fase de mayor importancia ya que se desarrollan las funcionalidades y se realiza un entregable funcional en base a las historias de usuario asignadas a la iteración.
- Fase de puesta en producción: entrega de módulos funcionales y sin errores.

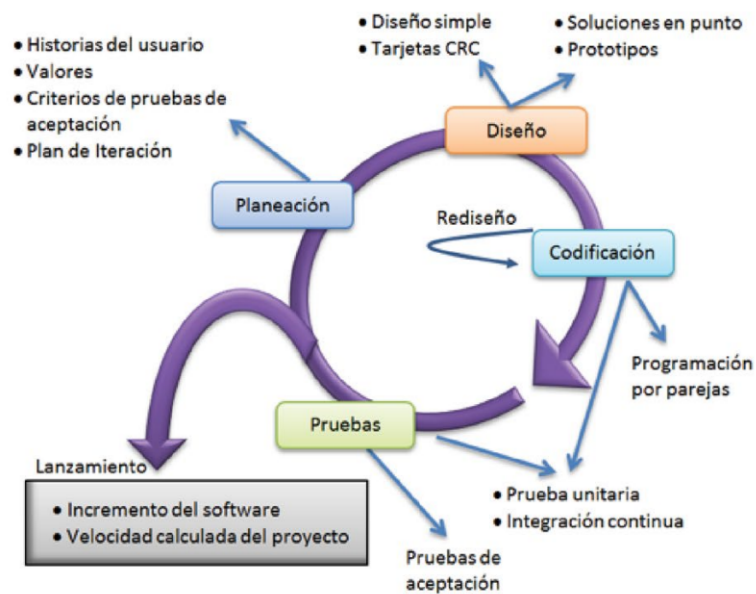


Figura 1.3: Fases de la metodología XP [2].

Kanban

Esta metodología se organiza por medio de un tablero de tareas denominado Kanban que permite a cada integrante del equipo visualizar el estado de implementación del software [26].



Figura 1.4: Fases de la metodología kanban [3].

Angular Angular es un framework de desarrollo que permite desarrollar aplicaciones web en la sección del cliente implementando HTML y Typescript. Permite la creación de aplicaciones web de una sola página con la carga de datos de forma asincrónica, lo cual mejora el rendimiento de las aplicaciones web. Se basa en la implementación de componentes con lo que se puede encapsular la funcionalidad optimizando el mantenimiento de las aplicaciones. Angular se caracteriza por implementar el patrón MVC, ser orientado a objetos y permitir el uso de TypeScript [4].

Principales elementos de Angular [4]:

- Modules.
- Components.
- Templates.
- Metadata.
- Data binding.
- Directives.
- Services.
- Dependency Injection

Arquitectura MVC [4]:

- Vistas: componentes.
- Capa de control: router.
- Backend: servicios.

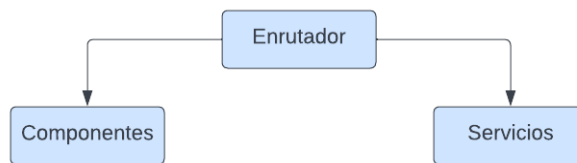


Figura 1.5: Representación MVC Angular [4].
Fuente: Elaborado por el investigador

Node js Es una plataforma basada en javascript desarrollado por Google se caracteriza por ser ligero y eficiente. Utiliza el modelo de E/S no bloqueante dirigido por eventos, su objetivo principal es facilitar la construcción de aplicaciones de red escalables. Node proporciona un entorno de ejecución para un determinado lenguaje de programación y un conjunto de librerías básicas. Por las características que posee node se puede decir que encaja perfecto cuando se requiere realizar lo siguiente [27]:

- Interfaces ligeras REST/JSON.
- Aplicaciones monopágina.
- Reutilizar herramientas de Unix.
- Datos por streaming.
- Comunicación.

API REST Se puede definir a una API REST como una interfaz programable que implementa el protocolo HTTP para la comunicación mientras cumple los principios de arquitectura REST (Representational State Transfer) [28].
Arquitectura REST [28]:

- Cliente-servidor.
- Stateless.

- Cache.
- Interface Uniforme.
- Sistema de capas.
- Código-en-demanda.

Métodos en REST

Los métodos de REST coinciden con los verbos de HTTP y son [5]:

- Post: creación.
- Get: lectura.
- Put: actualización.
- Delete: borrado.

También existen métodos secundarios tales como [5]:

- HEAD.
- OPTIONS.
- PATCH.

Limitaciones REST [29]:

- Protocolo cliente-servidor sin estado.
- Objetos en rest se gestionan a partir de la URI
- Interfaz uniforme
- Sistema de capas
- Uso de hipermedios

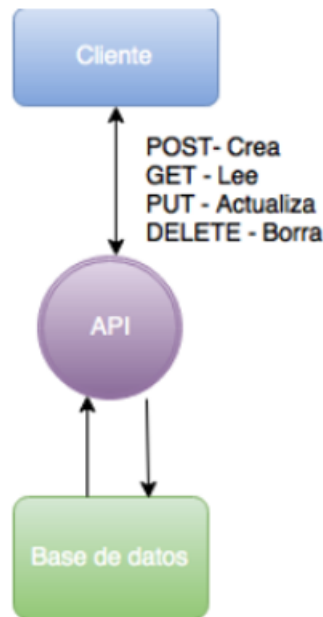


Figura 1.6: Diseño de API REST [5].

Object relational mapping (ORM) Es un mecanismo que permite la dirección, acceso y gestión de objetos sin tener que considerar como se relacionan dichos objetos con sus datos fuentes, además permite mantener una vista consistente de los objetos.

Una de las funciones de ORM es administrar los detalles de mapeo entre un conjunto de objetos y base de datos relacionales subyacentes, repositorios XML u otra fuentes y sumideros de datos mientras oculta los detalles cambiantes de las interfaces relacionadas de los desarrolladores y código que crean [30].

ORM es una técnica de programación para convertir datos entre sistemas de tipos incompatibles por medio de un lenguaje de programación orientado a objetos [30]. Los ORMs son una forma de acceder y manipular las bases de datos sin escribir consultas SQL ya que generan consultas en función del modelo de datos de JavaScript con lo que se reducen los errores de SQL y acelera el desarrollo [31].

Sequelize ORM Es un ORM o framework de mapeo objeto-relacion basado en promesas para Node.js y io.js, compatible con MSSQL, MariaDB, SQLite, MySQL y PostgreSQL; se caracteriza por contar con soporte de transacciones sólidas, relaciones y replicación de lectura [32].

1.3. Objetivos

1.3.1. Objetivo general

Definir un mecanismo de georeferenciación para la cadena de distribución de una iniciativa de comercio electrónico utilizando software libre para determinar el incremento de ventas realizadas a través del internet.

1.3.2. Objetivos específicos

- Identificar mecanismos de georeferenciación válidos para experiencias de comercio electrónico utilizando software libre.
- Implementar un mecanismo de georeferenciación integrado a un sistema de comercio electrónico para PYMEs, seleccionando herramientas de desarrollo web basadas en software libre.
- Validar el mecanismo implementado utilizando criterio de usuarios potenciales y expertos.

CAPÍTULO II

METODOLOGÍA

2.1. Materiales

2.1.1. Humanos

- Docente tutor de la Universidad Técnica de Ambato.
- Investigador.

2.1.2. Institucionales

- Facultad de Ingeniería en Sistemas, Electrónica e Industrial.
- Unidad Operativa de la Dirección de Investigación y Desarrollo de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial.
- Repositorios.
- Biblioteca virtual.

2.1.3. Otros

- Internet.
- Computador portátil.
- Libros y artículos científicos.

2.1.4. Económicos

Tabla 2.1: Recursos económicos

N°	Detalles	Unidad	Cantidad	Valor Unitario	Valor Total
1	Internet	h	340	\$0.60	\$204.00
2	Laptop	c/u	1	\$1430.00	\$1430.00
3	Hosting	c/u	1	\$224.21	\$224.21
4	Dominio	c/u	1	\$1.00	\$1.00
5	Impresiones	c/u	30	\$0.20	\$6.00
6	Luz eléctrica	Kw/h	310	\$0.01	\$3.10
7	Esferos	c/u	3	\$0.50	\$1.50
8	Lapiz	c/u	3	\$0.40	\$1.20
				Subtotal	\$1871.01
				Imprevistos (10%)	\$187.10
				Total	\$2058.11

Fuente: Elaborado por el investigador

2.2. Métodos

2.2.1. Modalidad de la investigación

Las modalidades de investigación que fueron utilizadas en el presente trabajo son:

Investigación bibliográfica

La presente investigación fue bibliográfica porque se utilizaron como puntos de referencia libros, trabajos de titulación alojadas en repositorios de diferentes universidades tanto a nivel nacional como internacional, revistas, artículos, entre otros que fueron sustentos para la elaboración del marco teórico.

Investigación descriptiva

La presente investigación fue descriptiva debido a que se realizó un análisis para determinar los requerimientos que son necesarios para realizar el aplicativo web que sirvió como complemento en la tienda virtual.

Investigación aplicada

La presente investigación además fue aplicada ya que a través de los conocimientos obtenidos durante la carrera se pudo realizar un análisis previo de las necesidades que tienen las pymes con respecto a un sistema de comercio electrónico con georeferenciación.

2.2.2. Población y muestra

Se estableció contacto con 3 empresas, las cuales tienen disponibles a expertos en el área de sistemas, mismos que se consideraron para validar la propuesta de investigación, teniendo en cuenta que Jakob Nielsen uno de los gurús a nivel mundial de la usabilidad concluyó que “se puede encontrar la mayoría de los problemas de usabilidad con un número de evaluadores entre tres y cinco” [33]. Además, en el presente trabajo de investigación al aplicar la fórmula para calcular una muestra de una población infinita [34], se pudo obtener un total de 96 personas, utilizando un nivel de confianza del 95 %, una proporción aproximada de empleados participes en labores de marketing digital del 0.5 y un error máximo admitido del 10 %, mismo que a priori se estableció basándose en el criterio de que serán pocas las empresas que utilizaran el aplicativo web.

$$n = \frac{(1.96)^2 \times 0.5 \times 0.5}{(0.1)^2} = 96.04 \approx 96$$

El resultado calculado de la muestra de personas sirvió como referencia para tener un máximo de participantes para realizar las encuestas y validar el sistema.

2.2.3. Recolección de la información

Tabla 2.2: Recolección de información

Preguntas básicas	Explicación
1. ¿Para qué?	Para cumplir los objetivos de la investigación
2. ¿De qué personas u objetos?	Clientes Empleados Proveedores
3. ¿Sobre qué aspectos?	Georeferenciación aplicada a sistemas de comercio electrónico
4. ¿Quién, quiénes?	Investigador: Sebastian Ricardo Mayorga Naranjo
5. ¿Cuándo?	Primer semestre del 2022
6. ¿Dónde?	Pequeñas y medianas empresas
7. ¿Cuántas veces?	Una
8. ¿Qué técnicas de recolección?	Encuesta Observación
9. ¿Con qué?	Formularios en línea Guía de encuesta

Fuente: Elaborado por el investigador

2.2.4. Metodología seleccionada

Una forma de acelerar el proceso de desarrollo de software es utilizando metodologías ágiles, en el presente proyecto se decidió trabajar con la metodología XP o más conocida como programación extrema, lo que permitió desarrollar un prototipo de software a través de ciclos de desarrollo cortos y pruebas constantes.

La metodología XP esta conformada por las siguientes fases:

- Fase 1: planificación.
- Fase 2: diseño.
- Fase 3: codificación.
- Fase 4: pruebas.

CAPÍTULO III

RESULTADOS Y DISCUSIÓN

3.1. Análisis y discusión de resultados

3.1.1. Validación de encuesta

La presente encuesta realizada fue validada a través del uso del coeficiente alfa de Cronbach, que es una medida de la consistencia interna de un cuestionario o una prueba de rendimiento [35]. Esta medida se utilizó para evaluar la confiabilidad de las preguntas del cuestionario realizado con el propósito de obtener resultados no sesgados de compradores en línea e información que sirvió de apoyo en nuevas definiciones del sistema creado. La fórmula aplicada para calcular el coeficiente de confiabilidad del cuestionario fue:

$$\alpha = \frac{K}{K - 1} \left[1 - \frac{\sum S_i^2}{S_T^2} \right]$$

En donde:

- α es el coeficiente de confiabilidad del cuestionario.
- K es el número de ítems del instrumento
- $\sum_{i=1}^k S_i^2$ es la sumatoria de las varianzas de los ítems.
- S_T^2 es varianza total del instrumento.

Para obtener el coeficiente de confiabilidad se estableció el número de ítems del instrumento en 11, dicho valor se obtuvo a partir del conteo de preguntas en la encuesta realizada, el siguiente valor que se obtuvo fue la sumatoria de las varianzas que tuvieron como resultado cada pregunta, siendo este valor 3.58, finalmente se obtuvo la varianza de la suma de los ítems cuyo valor fue 10.04, al reemplazar los valores previamente obtenidos en la fórmula anteriormente mencionada el valor resultante fue $0.707768924 \approx 0.71$. Cabe mencionar que para medir el nivel de confiabilidad al usar el coeficiente alfa de Cronbach se tiene rangos que a continuación se describen [35]:

Tabla 3.1: Rangos de confiabilidad para el coeficiente alfa de Cronbach

Rango	Confiabilidad
0.53 a menos	Confiabilidad nula
0.54 a 0.59	Confiabilidad baja
0.60 a 0.65	Confiable
0.66 a 0.71	Muy confiable
0.72 a 0.99	Excelente confiabilidad
1	Confiabilidad perfecta

Fuente: Elaborado por el investigador

El valor calculado al encontrarse en el rango de muy confiable demuestra que al aplicar el cuestionario se validó que las preguntas realizadas no están sesgadas y como resultado las respuestas que se obtengan son confiables por parte de los encuestados.

3.1.2. Resultados de encuesta

Al aplicar la encuesta a personas que realizan compras en línea regularmente se obtuvo la siguiente información:

1. ¿Cuál es su sexo?

Tabla 3.2: Resultados de encuesta - Pregunta 1

Opción	Respuestas	Porcentaje
Hombre	11	79 %
Mujer	3	21 %

Fuente: Elaborado por el investigador

Análisis: El 79 % de encuestados mencionan que son hombres, mientras que el 21 % son mujeres.

Interpretación de resultados: En lo que respecta a la población encuestada un total de 11 personas son hombres y 3 son mujeres, lo que se da a entender que la encuesta aplicada le llamó más la atención a un público masculino, mientras que las respuestas capturadas por un público femenino fueron menores.

2. ¿En qué intervalo de edad se encuentra?

Tabla 3.3: Resultados de encuesta - Pregunta 2

Opción	Respuestas	Porcentaje
21 - 25 años	13	93 %
26 - 35 años	1	7 %

Fuente: Elaborado por el investigador

Análisis: El 93 % de las personas encuestadas mencionan que tienen de 21 a 25 años, mientras que el 7 % tienen de 26 a 35 años.

Interpretación de resultados: Del total de los encuestados la mayoría son personas jóvenes, los cuales dominan el uso de tecnología y por ende realizan compras en línea de acuerdo a sus gustos.

3. ¿Realizas compras en línea?

Tabla 3.4: Resultados de encuesta - Pregunta 3

Opción	Respuestas	Porcentaje
Si	12	86 %
No	2	14 %

Fuente: Elaborado por el investigador

Análisis: El 86 % de las personas encuestadas afirman que realizan compras en línea para obtener productos que necesiten, mientras que el 14 % menciona que no realizan compras en línea.

Interpretación de resultados: La mayoría está de acuerdo en que realizan compras en línea para adquirir productos que necesiten, lo que demuestra que en la actualidad el uso del comercio electrónico está presente en la mayoría de gente que tiene internet y lo utiliza diariamente

4. ¿Con qué frecuencia adquiere productos en una tienda en línea?

Tabla 3.5: Resultados de encuesta - Pregunta 4

Opción	Respuestas	Porcentaje
1 o 2 veces al año	7	59 %
1 vez cada temporada	4	33 %
1 vez a la semana	1	8 %

Fuente: Elaborado por el investigador

Análisis: El 59 % de encuestados afirman que realizan compras en línea 1 o 2 veces al año, mientras que el 33 % realizan compras 1 vez cada temporada y el 8 % realiza 1 compra a la semana.

Interpretación de resultados: Demuestra que la mayoría de personas no realizan compras tan frecuentemente en internet, por lo que el uso de comercio electrónico tiende a ser solo informativo, es decir que la gente se centra en visualizar productos de su interés y consulta información del mismo.

5. ¿Qué tipo de productos compras en línea?

Tabla 3.6: Resultados de encuesta - Pregunta 5

Opción	Respuestas	Porcentaje
Accesorios tecnológicos Ropa	4	34 %
Accesorios tecnológicos	3	25 %
Ropa	2	17 %
Decoración Bisutería	1	8 %
Bisutería Videojuegos, suscripciones	1	8 %
Accesorios tecnológicos Productos digitales	1	8 %

Fuente: Elaborado por el investigador

Análisis: El 34 % de los encuestados mencionan que les interesa comprar en línea accesorios tecnológicos y ropa, al 25 % solo accesorios tecnológicos, al 17 % ropa, al 8 % decoración y bisutería, al 8 % bisutería, videojuegos y suscripciones y al 8 % restante accesorios tecnológicos y productos digitales.

Interpretación de resultados: Es de suma importancia observar que los accesorios tecnológicos y la ropa es lo que más llama la atención a la hora de realizar compras en línea por parte de una población joven, la bisutería de igual manera la compran pero con menos frecuencia y lo que les llama menos la atención es adquirir suscripciones a plataformas en línea.

6. ¿Las tiendas en línea en las que suele comprar le brindan una forma de obtener información georeferenciada de los productos que tiene?

Tabla 3.7: Resultados de encuesta - Pregunta 6

Opción	Respuestas	Porcentaje
No	9	75 %
Si	3	25 %

Fuente: Elaborado por el investigador

Análisis: El 75 % de los encuestados mencionan que las tiendas en línea no brindan una forma de obtener información georeferenciada, mientras que el 25 % dicen que si poseen información georeferenciada.

Interpretación de resultados: La mayoría esta en desacuerdo que las tiendas en línea no tienen una forma para obtener información georeferenciada, lo que demuestra la necesidad de implementar un mecanismo de georeferenciación en las tiendas virtuales que ofrecen productos a través del internet.

7. ¿Considera útil una aplicación que le permita conocer diferentes rutas por donde paso un producto?

Tabla 3.8: Resultados de encuesta - Pregunta 7

Opción	Respuestas	Porcentaje
Si	10	71 %
No	4	29 %

Fuente: Elaborado por el investigador

Análisis: El 71 % de las personas encuestadas consideran util una aplicación la cual permita conocer diferentes rutas por donde paso un producto, mientras que el 29 % mencionan que no es útil.

Interpretación de resultados: En lo que respecta al funcionamiento de una tienda virtual que tenga una aplicación que muestre las rutas de un producto por donde este haya pasado, la mayoría de los encuestados afirman que sería util esta aplicación, lo que demuestra que la mayoría se ha encontrado con tiendas virtuales que no poseen esta funcionalidad.

8. ¿Considera útil que una aplicación de compras en línea le indique las ubicaciones de los puntos de distribución donde se puede adquirir un producto buscado?

Tabla 3.9: Resultados de encuesta - Pregunta 8

Opción	Respuestas	Porcentaje
Si	13	93 %
No	1	7 %

Fuente: Elaborado por el investigador

Análisis: El 93 % de los encuestados consideran útil una aplicación de compras en línea que indique las ubicaciones de los puntos de distribución donde se puede adquirir un producto deseado, mientras que el 7 % no lo considera útil.

Interpretación de resultados: La mayoría está de acuerdo que sería útil que las aplicaciones de compras en línea indiquen las ubicaciones de los puntos de distribución donde se pueda adquirir un producto, ya que es información necesaria para que el cliente busque un producto y de esta manera se pueda dirigir directamente a una sucursal que tenga disponibilidad del mismo.

9. ¿Utilizaría el servicio de geolocalización de productos si algún local lo tuviese disponible?

Tabla 3.10: Resultados de encuesta - Pregunta 9

Opción	Respuestas	Porcentaje
Si	13	93 %
No	1	7 %

Fuente: Elaborado por el investigador

Análisis: El 93 % de encuestados les interesa utilizar un servicio de geolocalización en una tienda virtual, mientras que el 7 % no lo utilizaría.

Interpretación de resultados: Si una empresa tuviera implementado un servicio de geolocalización en una tienda virtual la mayoría de los encuestados lo utilizarían, ya que este les brindaría información de importancia referente a un producto que deseen.

10. ¿Considera usted que al implementar información georeferenciada en una tienda en línea se pueden incrementar las ventas dentro de una empresa?

Tabla 3.11: Resultados de encuesta - Pregunta 10

Opción	Respuestas	Porcentaje
Si	13	93 %
No	1	7 %

Fuente: Elaborado por el investigador

Análisis: El 93 % de las personas encuestadas consideran que existiría un aumento en las ventas de una empresa al implementar información georeferenciada en una tienda virtual de su disponibilidad, mientras que el 7 % piensan que no aumentarían las ventas con esta implementación.

Interpretación de resultados: La mayoría consideran que si una tienda en

línea implementa información georeferenciada de sus productos se puede incrementar las ventas dentro de una empresa, esto se debe a que la tienda virtual al tener un agregado como es el caso de la información georeferenciada llama más la atención de los clientes cuando desean adquirir un producto.

11. En un rango del 1 al 5, tomando 5 como muy de acuerdo y 1 como desacuerdo, considera usted que el sistema woomap implementado para mostrar información georeferenciada en una tienda virtual ayudaría a incrementar las ventas de la empresa que lo utilice?

Tabla 3.12: Resultados de encuesta - Pregunta 11

Opción	Respuestas	Porcentaje
3	6	43 %
4	6	43 %
2	1	7 %
5	1	7 %

Fuente: Elaborado por el investigador

Análisis: El 43 % de encuestados mencionan que no están de acuerdo ni en desacuerdo en que al implementar el sistema woomap ayudaría a incrementar las ventas en una empresa que lo implemente, el 43 % están de acuerdo, el 7 % están medianamente de acuerdo y el 7 % esta muy de acuerdo.

Interpretación de resultados: La mayor parte de personas encuestadas consideran que al implementar woomap para mostrar información georeferenciada ayudaría a incrementar las ventas que se realicen en una tienda en línea, estos resultados los consideraron ya que probaron el prototipo del sistema y observaron la funcionalidad del mismo.

3.1.3. Determinación del framework

Para determinar el framework a utilizar para la programación del frontend del presente proyecto, se decidió realizar un cuadro comparativo entre los frameworks con mayor popularidad, las características que fueron tomadas en cuenta para esta comparación son el lenguaje de programación, el patrón de diseño, curva de aprendizaje, comunidad y documentación y las principales características que tienen. La tabla 3.13 muestra el cuadro comparativo entre los diferentes frameworks.

Tabla 3.13: Comparación de frameworks

	Ember	Angular	React	VueJS
Lenguaje de programación	JavaScript	Typescript	JavaScript	JavaScript
Patrón de diseño	MVVC(modelo-vista-vista-modelo).	Arquitectura MVC(modelo-vista-controlador).	MVC	MVVM
Curva de aprendizaje	Alta	Media	Media	Baja
Comunidad y documentación	Comunidad activa, documentación en inglés.	Documentación amplia, gratuita y con variedad de ejmplos de código, comunidad activa.	Documentación gratuita, comunidad activa.	Amplia comunidad y documentación organizada.
Características	Utilización de una arquitectura estricta, lenguaje de plantillas Handlebar, poco flexible y orientado a aplicaciones de complejas de varias páginas.	Multiplataforma enlace de datos bidireccional, flexible, estructura de desarrollo modular, fácil mantenimiento, desarrollo web y móviles y orientado a aplicaciones de una página (SPA).	Código estable y seguro, desarrollo web y móvil, enlace de datos unidireccional y orientado a aplicaciones de una página (SPA).	Desarrollo web y móvil, flexible, enlace de datos por v-bind, vue-router, sintaxis e integración sencilla y orientado a aplicaciones de una página (SPA).

Fuente: Elaborado por el investigador

En el análisis realizado en la tabla 3.13 se eligió a angular como framework de desarrollo para el lado del frontend ya que permite desarrollar aplicaciones web de tipo single page application (SPA) a través de componentes que son reutilizables a lo largo del proyecto en ejecución, además el lenguaje de programación que utiliza es typescript el cual tiene varias características adicionales como un tipado estático que permite que las variables declaradas tengan un tipo de dato y los valores que se tenga o se calculen se puedan asignar exclusivamente a variables

del tipo al que correspondan, una aplicación creada utilizando angular puede ser escalable es decir que su código es fácil de entender, modificar y evolucionar permitiendo agregar nuevas funcionalidades o módulos cuando sea necesario, también en base al análisis comparativo se pudo determinar que angular tiene una comunidad activa, aportando de esta manera documentación que sirvió para generar código necesario en diferentes módulos creados.

3.1.4. Determinación del gestor de contenido

Para determinar el gestor de contenido que se va a utilizar en el presente proyecto se consideró realizar una comparación entre los CMS más populares, siendo un total de cinco, la comparación fue realizada en base a las características mas relevantes como son el tipo, curva de aprendizaje, personalización, SEO y las características principales de cada CMS. La tabla 3.14 muestra la comparación entre los distintos gestores de contenidos.

Tabla 3.14: Comparación gestores de contenido

	WordPress	Drupal	Joomla	Wix
Tipo	Gratuito y pagado	Gratuito	Gratuito	Gratuito y pagado
Curva de aprendizaje	Fácil implementación y usabilidad	Difícil	Intermedio	Fácil
Personalización	Alta	Alta	Media	Media
SEO	Muy bueno	Bueno	Deficiente	Bueno
Seguridad	Buena	Buena	Buena	Buena
Características	<ul style="list-style-type: none"> • Orientado a principiantes o emprendedores. • Adaptable para proyectos web como blogs y portafolios. • Múltiples themes o plantillas. • Amigable al usuario. • Infinidad de plugins. <ul style="list-style-type: none"> • Diseños básicos y agradables 	<ul style="list-style-type: none"> • Orientado a desarrolladores web • Variedad de plugins <ul style="list-style-type: none"> • Interfaz poco amigable • Sitios complejos orientados al mundo profesional 	<ul style="list-style-type: none"> • Orientado a diseñadores web • Diversidad de plugins <ul style="list-style-type: none"> • Sitios complejos orientados al mundo profesional 	<ul style="list-style-type: none"> • Orientado a principiantes o expertos • Múltiples plantillas • Interfaz sencilla e intuitiva

Fuente: Elaborado por el investigador

En base al análisis realizado en la tabla 3.14 se selecciono como CMS a wordpress, que es una plataforma open source que funciona para crear cualquier tipo de sitio web, ofrece características como la implementación de plugins que son pequeños programas que proporcionan funcionalidades extra a la página web y ofrece una extensa documentación que servirá de apoyo en la creación de la página de comercio electrónico.

3.1.5. Determinación del mecanismo de georeferenciación

Para determinar el mecanismo de georeferenciación que se utilizó en el presente proyecto, se decidió realizar un cuadro comparativo entre 6 tecnologías que ofrecen

este servicio, de esta manera se pudo observar sus características en cuanto a la gratuidad, las tecnologías que lo soportan, el tipo de personalización y la implementación de plugins. La tabla 3.15 muestra el cuadro comparativo entre los diferentes mecanismos de georeferenciación.

Tabla 3.15: Comparación mecanismos de georeferenciación

	OpenLayers	Leaflet	Google maps
Tipo	OpenSource	OpenSource	De pago / Versión de prueba
Lenguaje de programación	Javascript	Javascript	Javascript
Tecnologías soportadas	Navegador móvil / Navegador de escritorio	Navegador móvil / Navegador de escritorio	Navegador móvil / Navegador de escritorio
Personalización	<ul style="list-style-type: none"> • Diseño de características: puntos, líneas, polígonos e íconos. • Personalización de apariencia de los controles con css. 	<ul style="list-style-type: none"> • Personalización con ventanas emergentes que utilizan css3. • Fácil re-diseño. <ul style="list-style-type: none"> • Marcadores basados en imagen y html. • Capas de mapas, controles, proyecciones personalizadas. <ul style="list-style-type: none"> • Diseño de características: polígonos. 	<ul style="list-style-type: none"> • Modelo de matiz, saturación y luminosidad de los mapas. • Personalización de sus componentes
Controles e interacciones	Vista general, zoom (control deslizante, botones de encendido/apagado), escala, línea, desplazamiento de mapa y rotación	Arrastre y desplazamiento con inercia, navegación con el teclado, multi-touch, animación Zoom/Panorámica.	Zoom, seleccionar entre tipos de mapa (satélite, mapa, híbrido) o moverse por un mapa pequeño.

Fuente: Elaborado por el investigador

En el análisis realizado en la tabla 3.15 se seleccionó como mecanismo de georeferenciación a leaflet, el cual aportó un valor agregado en la implementación de mapas open source por su gran facilidad de uso y personalización, permitiendo de esta manera utilizarlo en la creación del sistema web.

3.2. Desarrollo de la propuesta

3.2.1. Fase 1: Planificación

En la fase de planificación el cliente comunica los requerimientos necesarios para poder realizar historias de usuario y de esta manera poner en marcha el proyecto.

3.2.1.1. Levantamiento de la información

La Facultad de Ingeniería en Sistemas Computacionales e Informáticos, con el fin de obtener un sistema web que aporte con la ejecución del proyecto de investigación SFFISEI02 titulado “DEFINICION DE UN MODELO DE NEGOCIO PARA EL FORTALECIMIENTO DE INICIATIVAS DEL COMERCIO ELECTRÓNICO EN TUNGURAHUA COMO ALTERNATIVA PARA ENFRENTAR LA CRISIS ECONÓMICA GENERADA POR LA PANDEMIA DEL COVID-19”, ha brindado la información necesaria para realizar el levantamiento de información y de esta manera obtener los requerimientos necesarios.

3.2.1.2. Definición de roles

Los roles en la metodología XP, nos permite establecer una responsabilidad diferente a cada integrante del equipo de trabajo, cabe resaltar que el presente proyecto al ser de un ámbito de investigación no ocupará todos los roles disponibles, sino que se limitará a tres roles:

Tabla 3.16: Definición de roles

Rol	Responsable	Función
Usuario	Ing. PhD. Félix Oscar Fernández Peña	Comprobar y validar las funcionalidades de cada proceso a ser implementado.
Programador	Sebastián Ricardo Mayorga Naranjo	Desarrollar el sistema web con mecanismo de georeferenciación utilizando herramientas open source.
Testers	Ing. Jefferson Zambrano Ing. Carlos Solórzano Ing. Alejandro Morales	Pruebas de aceptación.

Fuente: Elaborado por el investigador

3.2.1.3. Historias de usuario

Las historias de usuario deben representar los requerimientos que debe cumplir el sistema, tienen que estar escritas en un lenguaje común para ser entendidas tanto por los clientes como los desarrolladores.

Tabla 3.17: Plantilla de historia de usuario

Historia de usuario			
Número:		Usuario:	
Nombre:			
Prioridad:		Riesgo:	
Puntos estimados:		Iteración asignada:	
Programador responsable:			
Descripción:			
Observación:			

Fuente: Elaborado por el investigador

La plantilla expuesta esta conformada por los siguientes elementos:

- **Número:** Identificador de una historia de usuario.
- **Nombre:** Descripción general de una historia de usuario.
- **Prioridad:** Nivel de la importancia de la actividad, puede ser alta, media o baja.
- **Puntos estimados:** Cantidad de días estimados para el desarrollo de una historia de usuario.
- **Programador responsable:** Persona encargada de programar una historia de usuario.
- **Descripción:** Información adicional que detallada de mejor manera una historia de usuario.
- **Observación:** Campo opcional que se utiliza para aclarar el requerimiento que se haya descrito en una historia de usuario.
- **Usuario:** Persona o grupo de personas encargadas de una actividad.

- **Riesgo:** Nivel de riesgo que tiene el desarrollador con respecto a la elaboración de una historia de usuario, puede ser alta, media o baja.
- **Iteración asignada:** Número de iteración, en el que se desea implementar una historia de usuario.

Para el desarrollo del presente proyecto se detallan las siguientes historias de usuario:

Tabla 3.18: Historia de usuario: Definir la estructura del sistema

Historia de usuario			
Número:	H01	Usuario:	Desarrollador
Nombre:	Estructura del sistema		
Prioridad:	Alta	Riesgo:	Alto
Puntos estimados:	4	Iteración asignada:	1
Programador responsable:	Sebastián Mayorga		
Descripción:	Conocer la estructura que el proyecto va a tener para que el desarrollo del mismo sea mas simple y de esta manera se facilite el uso de las tecnologías y herramientas de desarrollo del sistema.		
Observación:	Ninguna		

Fuente: Elaborado por el investigador

Tabla 3.19: Historia de usuario: Comercio electrónico

Historia de usuario			
Número:	H02	Usuario:	Todos
Nombre:	Comercio electrónico		
Prioridad:	Alta	Riesgo:	Alto
Puntos estimados:	8	Iteración asignada:	1
Programador responsable:	Sebastián Mayorga		
Descripción:	Elaborar una página web que sea de tipo e-commerce utilizando la herramienta wordpress como CMS.		
Observación:	Ninguna.		

Fuente: Elaborado por el investigador

Tabla 3.20: Historia de usuario: Estructura de la base de datos

Historia de usuario			
Número:	H03	Usuario:	Desarrollador
Nombre:	Diseño de la base de datos		
Prioridad:	Alta	Riesgo:	Alto
Puntos estimados:	3	Iteración asignada:	1
Programador responsable:	Sebastián Mayorga		
Descripción:	Elaborar una base de datos que cumpla con los requerimientos del sistema y permita a los usuarios acceder a la información que necesiten.		
Observación:	Ninguna.		

Fuente: Elaborado por el investigador

Tabla 3.21: Historia de usuario: Productos e información georeferenciada

Historia de usuario			
Número:	H04	Usuario:	Administrador, proveedor, controlador, usuario
Nombre:	Productos e información georeferenciada		
Prioridad:	Alta	Riesgo:	Medio
Puntos estimados:	8	Iteración asignada:	2
Programador responsable:	Sebastián Mayorga		
Descripción:	Los usuarios podrán visualizar los productos que se encuentran disponibles en el e-commerce, además de la información georeferenciada de todos los puntos de distribución que se encuentren registrados, las ubicaciones del recorrido de cada proveedor que tiene a cargo ese producto y finalmente los puntos de distribución que tengan disponible el producto.		
Observación:	Ninguna.		

Fuente: Elaborado por el investigador

Tabla 3.22: Historia de usuario: Registro en el sistema

Historia de usuario			
Número:	H05	Usuario:	Administrador, proveedores, controladores
Nombre:	Registro en el sistema		
Prioridad:	Alta	Riesgo:	Alto
Puntos estimados:	2	Iteración asignada:	2
Programador responsable:	Sebastián Mayorga		
Descripción:	Los usuarios que deseen ingresar al sistema, deberán registrar su información personal.		
Observación:	Cada campo del registro será obligatorio y se necesitará la validación de un administrador para tener acceso al sistema.		

Fuente: Elaborado por el investigador

Tabla 3.23: Historia de usuario: Acceso al sistema

Historia de usuario			
Número:	H06	Usuario:	Desarrollador
Nombre:	Acceso al sistema		
Prioridad:	Alta	Riesgo:	Alto
Puntos estimados:	4	Iteración asignada:	2
Programador responsable:	Sebastián Mayorga		
Descripción:	Los usuarios registrados y con su cuenta habilitada podrán iniciar sesión utilizando su correo y contraseña que hayan registrado.		
Observación:	Ninguna		

Fuente: Elaborado por el investigador

Tabla 3.24: Historia de usuario: Administrar usuarios

Historia de usuario			
Número:	H07	Usuario:	Administrador
Nombre:	Administrar usuarios		
Prioridad:	Alta	Riesgo:	Alto
Puntos estimados:	7	Iteración asignada:	3
Programador responsable:	Sebastián Mayorga		
Descripción:	El usuario podrá asignar roles a los usuarios registrados en el sistema y actualizar su información en caso de ser necesario.		
Observación:	Ninguna		

Fuente: Elaborado por el investigador

Tabla 3.25: Historia de usuario: Gestión de puntos de distribución

Historia de usuario			
Número:	H08	Usuario:	Administrador
Nombre:	Gestión de puntos de distribución		
Prioridad:	Alta	Riesgo:	Alto
Puntos estimados:	5	Iteración asignada:	3
Programador responsable:	Sebastián Mayorga		
Descripción:	El usuario podrá crear, modificar, eliminar y visualizar los puntos de distribución establecidos en el sistema.		
Observación:	Ninguna		

Fuente: Elaborado por el investigador

Tabla 3.26: Historia de usuario: Gestión de personal por producto

Historia de usuario			
Número:	H09	Usuario:	Administrador
Nombre:	Gestión de personal por producto		
Prioridad:	Alta	Riesgo:	Alto
Puntos estimados:	4	Iteración asignada:	3
Programador responsable:	Sebastián Mayorga		
Descripción:	El usuario podrá establecer un personal encargado de un producto y visualizar cuales han sido asignados a dicho producto.		
Observación:	Ninguna		

Fuente: Elaborado por el investigador

Tabla 3.27: Historia de usuario: Asignar ubicaciones por producto

Historia de usuario			
Número:	H10	Usuario:	Proveedor
Nombre:	Asignar ubicaciones por producto		
Prioridad:	Alta	Riesgo:	Alto
Puntos estimados:	3	Iteración asignada:	4
Programador responsable:	Sebastián Mayorga		
Descripción:	El usuario podrá establecer, editar, eliminar y visualizar n ubicaciones de un producto que se le haya asignado.		
Observación:	Ninguna		

Fuente: Elaborado por el investigador

Tabla 3.28: Historia de usuario: Validar ubicaciones

Historia de usuario			
Número:	H11	Usuario:	Controlador
Nombre:	Validar ubicaciones		
Prioridad:	Alta	Riesgo:	Alto
Puntos estimados:	3	Iteración asignada:	4
Programador responsable:	Sebastián Mayorga		
Descripción:	El usuario podrá visualizar los productos que se le hayan asignado para validar las ubicaciones registradas por los proveedores.		
Observación:	Ninguna		

Fuente: Elaborado por el investigador

Tabla 3.29: Historia de usuario: Designar puntos de distribución

Historia de usuario			
Número:	H12	Usuario:	Controlador
Nombre:	Designar puntos de distribución		
Prioridad:	Alta	Riesgo:	Alto
Puntos estimados:	4	Iteración asignada:	4
Programador responsable:	Sebastián Mayorga		
Descripción:	El usuario podrá visualizar, agregar o remover los puntos de distribución que tiene un producto.		
Observación:	Ninguna		

Fuente: Elaborado por el investigador

3.2.1.4. Tareas

Son actividades que tienen a un responsable asignado para desarrollarlas, además tienen un inicio, un fin y un tiempo establecido.

Tabla 3.30: Plantilla de tarea de historia de usuario

Tarea			
Número de tarea:		Código de historia:	
Nombre de tarea:			
Tipo de tarea:		Puntos estimados:	
Fecha inicio:		Fecha fin:	
Responsable:			
Descripción:			

Fuente: Elaborado por el investigador

La plantilla de tareas esta conformada por los siguientes elementos:

- **Número de tarea:** identificador asignado a la tarea.
- **Código de historia:** identificador que hace referencia a una historia de usuario.
- **Nombre de tarea:** describe de manera general a una tarea.
- **Tipo de tarea:** tipo que pertenece una tarea.
- **Puntos estimados:** cantidad de días estimados para el desarrollo de una tarea.
- **Fecha inicio:** fecha inicial de la creación de la tarea.
- **Fecha fin:** fecha de conclusión de la tarea.
- **Responsable:** persona encargada de desarrollar la tarea.
- **Descripción:** Información adicional que detallada de mejor manera una tarea.

Teniendo en cuenta las historias de usuario definidas y sus iteraciones asignadas respectivamente, el siguiente paso fue la definición de las tareas, mismas que se muestran a continuación:

Tabla 3.31: Tarea: Análisis del sistema

Tarea			
Número de tarea:	T01	Código de historia:	H01
Nombre de la tarea:	Determinar la estructura del sistema		
Tipo de tarea:	Análisis	Puntos estimados:	3
Fecha inicio:	08/03/2022	Fecha fin:	10/03/2022
Responsable:	Sebastián Mayorga		
Descripción:	Se analizará la estructura del sistema y se determinarán las tecnologías a utilizarse.		

Fuente: Elaborado por el investigador

Tabla 3.32: Tarea: Creación del comercio electrónico

Tarea			
Número de tarea:	T02	Código de historia:	H02
Nombre de la tarea:	Creación del comercio electrónico		
Tipo de tarea:	Desarrollo	Puntos estimados:	5
Fecha inicio:	11/03/2022	Fecha fin:	15/03/2022
Responsable:	Sebastián Mayorga		
Descripción:	Se modelará una página web utilizando wordpress y woocommerce como plugin para crear una tienda virtual.		

Fuente: Elaborado por el investigador

Tabla 3.33: Tarea: Análisis de la base de datos

Tarea			
Número de tarea:	T03	Código de historia:	H03
Nombre de la tarea:	Análisis de la base de datos		
Tipo de tarea:	Análisis	Puntos estimados:	3
Fecha inicio:	16/03/2022	Fecha fin:	19/03/2022
Responsable:	Sebastián Mayorga		
Descripción:	Se realizará un análisis de entidades para crear de un diagrama de clases que servirá de apoyo para modelar la base de datos.		

Fuente: Elaborado por el investigador

Tabla 3.34: Tarea: Diseño de interfaz de inicio del sistema

Tarea			
Número de tarea:	T04	Código de historia:	H04
Nombre de la tarea:	Diseño de interfaz de inicio del sistema		
Tipo de tarea:	Desarrollo	Puntos estimados:	3
Fecha inicio:	20/03/2022	Fecha fin:	22/03/2022
Responsable:	Sebastián Mayorga		
Descripción:	Se desarrollará el diseño de la interfaz que todos los usuarios podrán visualizar al abrir el sistema en su navegador.		

Fuente: Elaborado por el investigador

Tabla 3.35: Tarea: Diseño de interfaz del detalle de un producto seleccionado

Tarea			
Número de tarea:	T05	Código de historia:	H04
Nombre de la tarea:	Diseño de interfaz del detalle de un producto seleccionado		
Tipo de tarea:	Desarrollo	Puntos estimados:	3
Fecha inicio:	20/03/2022	Fecha fin:	22/03/2022
Responsable:	Sebastián Mayorga		
Descripción:	Se realizará el diseño de la interfaz que un usuario visualizará al abrir la información de un producto disponible.		

Fuente: Elaborado por el investigador

Tabla 3.36: Tarea: Elaboración de métodos para obtener los productos disponibles

Tarea			
Número de tarea:	T06	Código de historia:	H04
Nombre de la tarea:	Elaboración de métodos para obtener los productos disponibles		
Tipo de tarea:	Desarrollo	Puntos estimados:	3
Fecha inicio:	23/03/2022	Fecha fin:	25/03/2022
Responsable:	Sebastián Mayorga		
Descripción:	Se desarrollarán los métodos para obtener todos los productos disponibles y su información detallada.		

Fuente: Elaborado por el investigador

Tabla 3.37: Tarea: Diseño de la interfaz de registro de usuario

Tarea			
Número de tarea:	T07	Código de historia:	H05
Nombre de la tarea:	Diseño de la interfaz de registro de usuario		
Tipo de tarea:	Desarrollo	Puntos estimados:	2
Fecha inicio:	26/03/2022	Fecha fin:	27/03/2022
Responsable:	Sebastián Mayorga		
Descripción:	Se realizará el diseño de la interfaz para el registro de un usuario en el sistema.		

Fuente: Elaborado por el investigador

Tabla 3.38: Tarea: Elaboración de métodos para registrar usuarios

Tarea			
Número de tarea:	T08	Código de historia:	H05
Nombre de la tarea:	Elaboración de métodos para registrar usuarios		
Tipo de tarea:	Desarrollo	Puntos estimados:	4
Fecha inicio:	28/03/2022	Fecha fin:	31/03/2022
Responsable:	Sebastián Mayorga		
Descripción:	Se desarrollarán los métodos para registrar un usuario en el sistema.		

Fuente: Elaborado por el investigador

Tabla 3.39: Tarea: Diseño de la interfaz de inicio de sesión

Tarea			
Número de tarea:	T09	Código de historia:	H06
Nombre de la tarea:	Diseño de la interfaz de inicio de sesión		
Tipo de tarea:	Desarrollo	Puntos estimados:	2
Fecha inicio:	01/04/2022	Fecha fin:	02/04/2022
Responsable:	Sebastián Mayorga		
Descripción:	Se realizará el diseño de la interfaz de inicio de sesión para los usuarios que se hayan registrado en el sistema.		

Fuente: Elaborado por el investigador

Tabla 3.40: Tarea: Elaboración de métodos para el inicio de sesión

Tarea			
Número de tarea:	T10	Código de historia:	H06
Nombre de la tarea:	Elaboración de métodos para el inicio de sesión		
Tipo de tarea:	Desarrollo	Puntos estimados:	6
Fecha inicio:	03/04/2022	Fecha fin:	08/04/2022
Responsable:	Sebastián Mayorga		
Descripción:	Se desarrollarán los métodos necesarios para validar el ingreso de sesión, utilizando tokens de inicio de sesión.		

Fuente: Elaborado por el investigador

Tabla 3.41: Tarea: Diseño de interfaz para la gestión de usuarios

Tarea			
Número de tarea:	T11	Código de historia:	H07
Nombre de la tarea:	Diseño de interfaz para la gestión de usuarios		
Tipo de tarea:	Desarrollo	Puntos estimados:	3
Fecha inicio:	09/04/2022	Fecha fin:	11/04/2022
Responsable:	Sebastián Mayorga		
Descripción:	Se realizará la interfaz para gestionar los usuarios registrados en el sistema.		

Fuente: Elaborado por el investigador

Tabla 3.42: Tarea: Elaboración de métodos para gestionar usuarios registrados en el sistema

Tarea			
Número de tarea:	T12	Código de historia:	H07
Nombre de la tarea:	Elaboración de métodos para gestionar usuarios registrados en el sistema		
Tipo de tarea:	Desarrollo	Puntos estimados:	5
Fecha inicio:	12/04/2022	Fecha fin:	16/04/2022
Responsable:	Sebastián Mayorga		
Descripción:	Se desarrollarán los métodos necesarios para gestionar la información, activación de cuenta y los roles que tiene un usuario registrado en el sistema.		

Fuente: Elaborado por el investigador

Tabla 3.43: Tarea: Diseño de interfaz para gestionar puntos de distribución

Tarea			
Número de tarea:	T13	Código de historia:	H08
Nombre de la tarea:	Diseño de interfaz para gestionar puntos de distribución		
Tipo de tarea:	Desarrollo	Puntos estimados:	3
Fecha inicio:	17/04/2022	Fecha fin:	19/04/2022
Responsable:	Sebastián Mayorga		
Descripción:	Se realizará el diseño de la interfaz para gestionar los puntos de distribución que estarán disponibles en el sistema.		

Fuente: Elaborado por el investigador

Tabla 3.44: Tarea: Elaboración de métodos para gestionar puntos de distribución

Tarea			
Número de tarea:	T14	Código de historia:	H08
Nombre de la tarea:	Elaboración de métodos para gestionar puntos de distribución		
Tipo de tarea:	Desarrollo	Puntos estimados:	4
Fecha inicio:	20/04/2022	Fecha fin:	23/04/2022
Responsable:	Sebastián Mayorga		
Descripción:	Se desarrollarán los métodos para gestionar los puntos de distribución disponibles en el sistema.		

Fuente: Elaborado por el investigador

Tabla 3.45: Tarea: Diseño de interfaz para gestionar personal por producto

Tarea			
Número de tarea:	T15	Código de historia:	H09
Nombre de la tarea:	Diseño de interfaz para gestionar personal por producto		
Tipo de tarea:	Desarrollo	Puntos estimados:	3
Fecha inicio:	24/04/2022	Fecha fin:	26/04/2022
Responsable:	Sebastián Mayorga		
Descripción:	Se realizará el diseño de la interfaz para gestionar el personal encargado de cada producto.		

Fuente: Elaborado por el investigador

Tabla 3.46: Tarea: Elaboración de métodos para gestionar personal por producto

Tarea			
Número de tarea:	T16	Código de historia:	H09
Nombre de la tarea:	Elaboración de métodos para gestionar personal por producto		
Tipo de tarea:	Desarrollo	Puntos estimados:	5
Fecha inicio:	27/04/2022	Fecha fin:	01/05/2022
Responsable:	Sebastián Mayorga		
Descripción:	Se desarrollará los métodos para gestionar el personal por cada producto que se encuentre registrado en el sistema.		

Fuente: Elaborado por el investigador

Tabla 3.47: Tarea: Diseño de interfaz para gestionar ubicaciones de un producto

Tarea			
Número de tarea:	T17	Código de historia:	H10
Nombre de la tarea:	Diseño de interfaz para gestionar ubicaciones de un producto		
Tipo de tarea:	Desarrollo	Puntos estimados:	3
Fecha inicio:	02/05/2022	Fecha fin:	04/05/2022
Responsable:	Sebastián Mayorga		
Descripción:	Se realizará el diseño de la interfaz para gestionar las ubicaciones de un producto.		

Fuente: Elaborado por el investigador

Tabla 3.48: Tarea: Elaboración de métodos para gestionar ubicaciones de un producto

Tarea			
Número de tarea:	T18	Código de historia:	H10
Nombre de la tarea:	Elaboración de métodos para gestionar ubicaciones de un producto		
Tipo de tarea:	Desarrollo	Puntos estimados:	6
Fecha inicio:	05/05/2022	Fecha fin:	10/05/2022
Responsable:	Sebastián Mayorga		
Descripción:	Se desarrollarán los métodos para gestionar las ubicaciones que tiene un producto.		

Fuente: Elaborado por el investigador

Tabla 3.49: Tarea: Diseño de interfaz para validar ubicaciones de productos

Tarea			
Número de tarea:	T19	Código de historia:	H11
Nombre de la tarea:	Diseño de interfaz para validar ubicaciones de productos		
Tipo de tarea:	Desarrollo	Puntos estimados:	3
Fecha inicio:	11/05/2022	Fecha fin:	13/05/2022
Responsable:	Sebastián Mayorga		
Descripción:	Se realizará el diseño de la interfaz que muestra las ubicaciones registradas por proveedores para su respectiva validación por parte del controlador.		

Fuente: Elaborado por el investigador

Tabla 3.50: Tarea: Elaboración de métodos para validar ubicaciones de productos

Tarea			
Número de tarea:	T20	Código de historia:	H11
Nombre de la tarea:	Elaboración de métodos para validar ubicaciones de productos		
Tipo de tarea:	Desarrollo	Puntos estimados:	4
Fecha inicio:	14/05/2022	Fecha fin:	17/05/2022
Responsable:	Sebastián Mayorga		
Descripción:	Se desarrollarán los métodos para validar las ubicaciones de un producto que un proveedor haya registrado en el sistema.		

Fuente: Elaborado por el investigador

Tabla 3.51: Tarea: Diseño de la interfaz para gestionar puntos de distribución de un producto

Tarea			
Número de tarea:	T21	Código de historia:	H12
Nombre de la tarea:	Diseño de la interfaz para designar puntos de distribución de un producto		
Tipo de tarea:	Desarrollo	Puntos estimados:	4
Fecha inicio:	18/05/2022	Fecha fin:	21/05/2022
Responsable:	Sebastián Mayorga		
Descripción:	Se desarrollarán los métodos para designar las ubicaciones de un producto que un proveedor haya registrado en el sistema.		

Fuente: Elaborado por el investigador

Tabla 3.52: Tarea: Elaboración de métodos para designar puntos de distribución de un producto

Tarea			
Número de tarea:	T22	Código de historia:	H12
Nombre de la tarea:	Elaboración de métodos para designar puntos de distribución de un producto		
Tipo de tarea:	Desarrollo	Puntos estimados:	5
Fecha inicio:	22/05/2022	Fecha fin:	26/05/2022
Responsable:	Sebastián Mayorga		
Descripción:	Se desarrollarán los métodos para designar los puntos de distribución que tiene un producto.		

Fuente: Elaborado por el investigador

3.2.1.5. Valoración de historias de usuario

En las historias de usuario se detallaron los requerimientos que tendrá el sistema, cada una de ellas se conformará por valoraciones definidas, que permitieron definir que se trabajará un total de 5 horas diarias.

3.2.1.6. Estimación de historias de usuario

Se realizó la estimación del esfuerzo que se va a realizar en las historias de usuario, que se representaran con un total de cuatro iteraciones según la funcionalidad, obteniendo de esta manera una aproximación de la duración del proyecto a través

del tiempo estimado, el cuál fue calculado en función de los días que en cada historia de trabajo se estableció y el total de horas que se definió en la valoración de las historias de usuario.

Tabla 3.53: Estimación de historias de usuario

Iteración	Número	Historia de usuario	Tiempo estimado	
			Días	Horas
1	H01	Estructura del sistema	4	20
1	H02	Comercio electrónico	8	40
1	H03	Diseño de la base de datos	3	15
2	H04	Productos e información georeferenciada	8	40
2	H05	Registro en el sistema	2	10
2	H06	Acceso al sistema	4	20
3	H07	Administrar usuarios	7	35
3	H08	Gestión de puntos de distribución	5	25
3	H09	Gestión de personal por producto	4	20
4	H10	Asignar ubicaciones por producto	3	15
4	H11	Validar ubicaciones	3	15
4	H12	Designar puntos de distribución	4	20
Tiempo total de estimación (días/horas)			55	275

Fuente: Elaborado por el investigador

Iteración 1

Tabla 3.54: Resumen iteración 1

Iteración	Número	Historia de usuario	Tiempo estimado	
			Días	Horas
1	H01	Estructura del sistema	4	20
1	H02	Comercio electrónico	8	40
1	H03	Diseño de la base de datos	3	15
Total			15	75

Fuente: Elaborado por el investigador

Iteración 2

Tabla 3.55: Resumen iteración 2

Iteración	Número	Historia de usuario	Tiempo estimado	
			Días	Horas
2	H04	Productos e información georeferenciada	8	40
2	H05	Registro en el sistema	2	10
2	H06	Acceso al sistema	4	20
Total			14	70

Fuente: Elaborado por el investigador

Iteración 3

Tabla 3.56: Resumen iteración 3

Iteración	Número	Historia de usuario	Tiempo estimado	
			Días	Horas
3	H07	Administrar usuarios	7	35
3	H08	Gestión de puntos de distribución	5	25
3	H09	Gestión de personal por producto	4	20
Total			16	80

Fuente: Elaborado por el investigador

Iteración 4

Tabla 3.57: Resumen iteración 1

Iteración	Número	Historia de usuario	Tiempo estimado	
			Días	Horas
4	H10	Asignar ubicaciones por producto	3	15
4	H11	Validar ubicaciones	3	15
4	H12	Designar puntos de distribución	4	20
Total			10	50

Fuente: Elaborado por el investigador

3.2.1.7. Plan de entrega

Tabla 3.58: Plan de entrega

N°	Historia de usuario	Tiempo estimado		Iteración asignada				Entrega asignada			
		Días	Horas	1	2	3	4	1	2	3	4
001	Estructura del sistema	4	20	X				X			
002	Comercio electrónico	8	40	X				X			
003	Diseño de la base de datos	3	15	X				X			
004	Productos e información georeferenciada	8	40		X				X		
005	Registro en el sistema	2	10		X				X		
006	Acceso al sistema	4	20		X				X		
007	Administrar usuarios	7	35			X				X	
008	Gestión de puntos de distribución	5	25			X				X	
009	Gestión de personal por producto	4	20			X				X	
010	Asignar ubicaciones por producto	3	15				X				X
011	Validar ubicaciones	3	15				X				X
012	Designar puntos de distribución	4	20				X				X

Fuente: Elaborado por el investigador

3.2.2. Fase 2: Diseño

3.2.2.1. Estructura del proyecto

La aplicación web desarrollada para aportar con la ejecución del proyecto de investigación SFFISEI02, proporcionará información georeferenciada de productos que estén disponibles en una tienda virtual.

Arquitectura del software

El sistema realizado tiene la siguiente arquitectura, en la **Figura 3.12** se puede observar que tanto la tienda en línea como el sistema woomap fueron alojados en hostinger y para el despliegue del backend se utilizó railway.

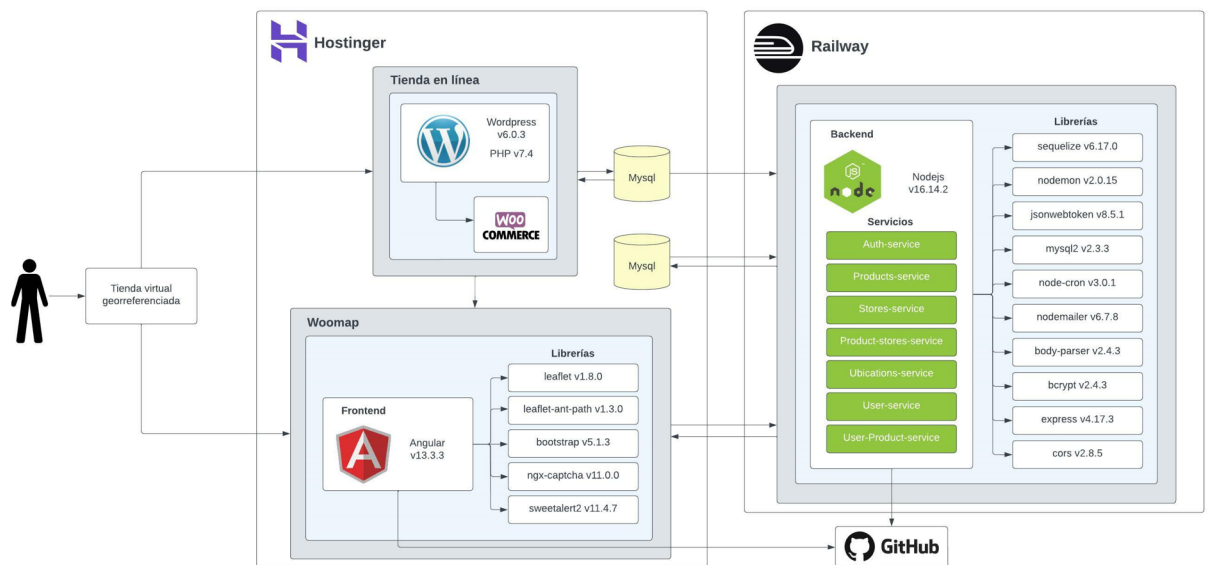


Figura 3.1: Arquitectura del proyecto woomap
Fuente: Elaborado por el investigador

3.2.2.2. Herramientas utilizadas

Hardware

- Laptop Msi GL 63 8RE, 16GB de RAM, intel core i7-8750H, SSD 1TB.
- Monitores Quasad, Samsung.

Software

- Package manager: npm versión 8.5.0.
- Editor de código fuente: visual studio code.
- Página web: la página de comercio electrónico fue implementada utilizando el plugin de woocommerce y utilizando wordpress 6.0.3 como CMS.
- Base de datos: para la persistencia de datos se utilizó MySQL y como administrador de base de datos se usó phpMyAdmin versión 5.1.1.

- Frontend: para el desarrollo de las interfaces y funcionalidades se utilizó angular versión 13.3.1, los estilos utilizados para las interfaces se implementaron con bootstrap , angular material y estilos propios.
- Backend: para el desarrollo de las APIs se utilizó node js versión 16.14.2 y express como framework, además para las consultas realizadas en cada servicio se utilizo un sequelize como ORM.

3.2.2.3. Esquema del proyecto

El presente proyecto al ser desarrollado utilizando angular se tendrá diferentes carpetas que almacenarán los archivos necesarios según su función, esta estructura permite manejar de una manera mas ordenada los archivos y de esta manera poder implementar con facilidad futuras funcionalidades en la aplicación desarrollada.

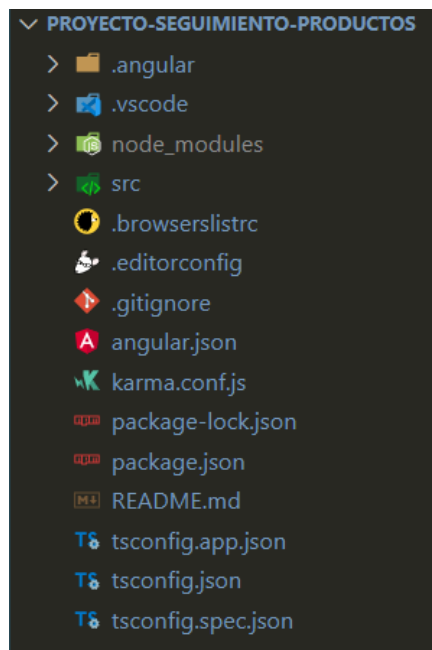


Figura 3.2: Estructura del frontend
Fuente: Elaborado por el investigador

- **node_modules:** incluye los módulos de angular, mismos que son necesarios para inicializar un proyecto.
- **src:** contiene los modulos creados para el sistema web.

Del lado del servidor se utilizó node js, el cual permite llevar una estructura MVC y a través de express se inició un servidor, el cual funcionó para realizar peticiones programadas (GET, POST, PUT y DELETE).

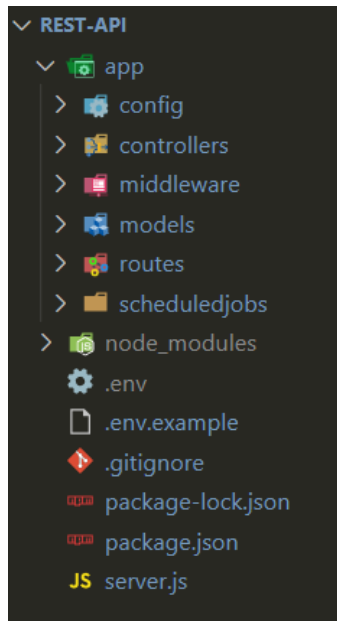


Figura 3.3: Estructura del backend
Fuente: Elaborado por el investigador

- **config:** contiene archivos de configuración, como credenciales de las bases de datos y llaves secretas para encriptar información.
- **controllers:** se encuentran todos los métodos para consultar, ingresar, eliminar o modificar información de la base de datos.
- **middleware:** contiene los métodos necesarios para realizar validaciones adicionales de seguridad al ejecutar un servicio.
- **models:** se encuentran definidos los modelos que existen en el sistema web, además contiene el archivo de configuración para modelar la base de datos a través de estos modelos definidos.
- **routes:** contiene las rutas de acceso para consumir un servicio.
- **scheduledjobs:** se encuentran los cronjobs que el sistema ejecutará.

3.2.2.4. Diseño de la base de datos

Al analizar la funcionalidad del sistema se realizó el correcto modelamiento de la base de datos, misma que tendrá un total de 8 tablas relacionadas, el motor de base de datos utilizado es MySQL.

- **Roles:** esta tabla almacena los roles de usuario definidos en el sistema que son administrador, proveedor y controlador.

- **Users:** en esta tabla se encuentran los usuarios que se hayan registrado en el sistema.
- **Users_roles:** es la tabla que contiene los roles referenciados con cada usuario.
- **Ubications:** esta tabla contiene las ubicaciones registradas de un producto por parte de un usuario.
- **Users_products:** en esta tabla esta los productos referenciados a un controlador o proveedor.
- **Products:** aqui se almacenan los productos que posee la tienda virtual.
- **Products_stores:** en esta tabla se encuentran las sucursales relacionadas con un producto o productos.
- **Stores:** posee las sucursales registradas en el sistema.

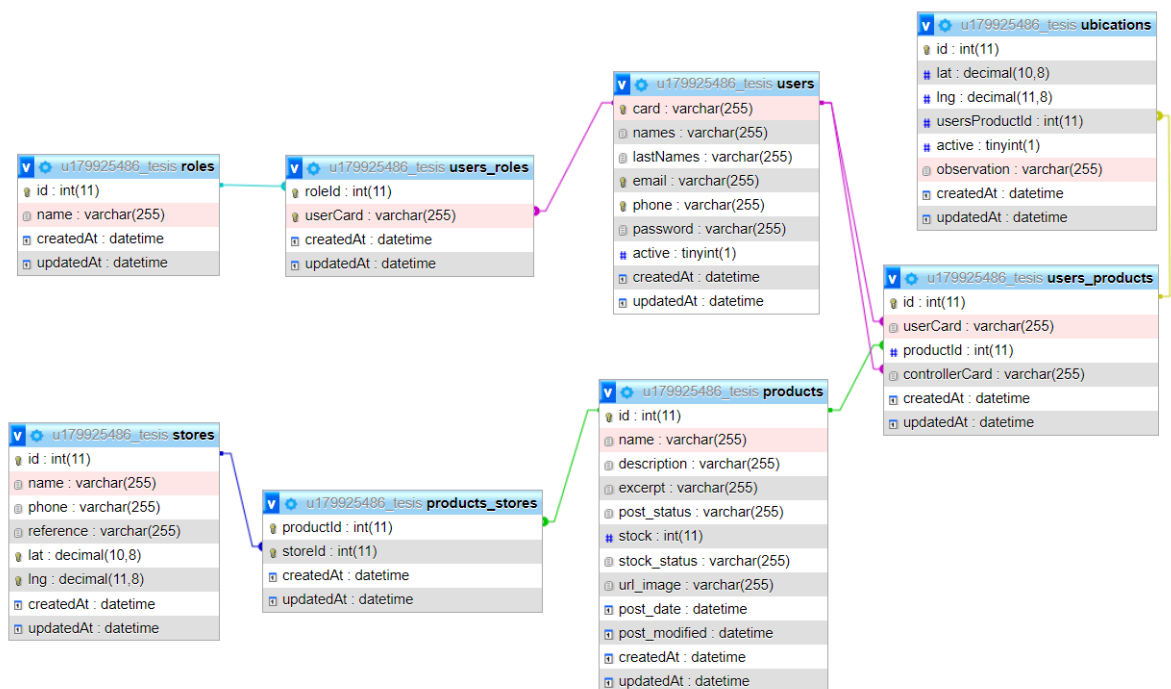


Figura 3.4: Esquema de base de datos del sistema web.

Fuente: Elaborado por el investigador

La base de datos que se utilizó para la página encargada del comercio electrónico de igual manera es MySQL, dicha base se modeló de manera automática al

implementar wordpress como CMS, la gestión de datos almacenados en esta base de datos se realiza a través de la página web creada como e-commerce.

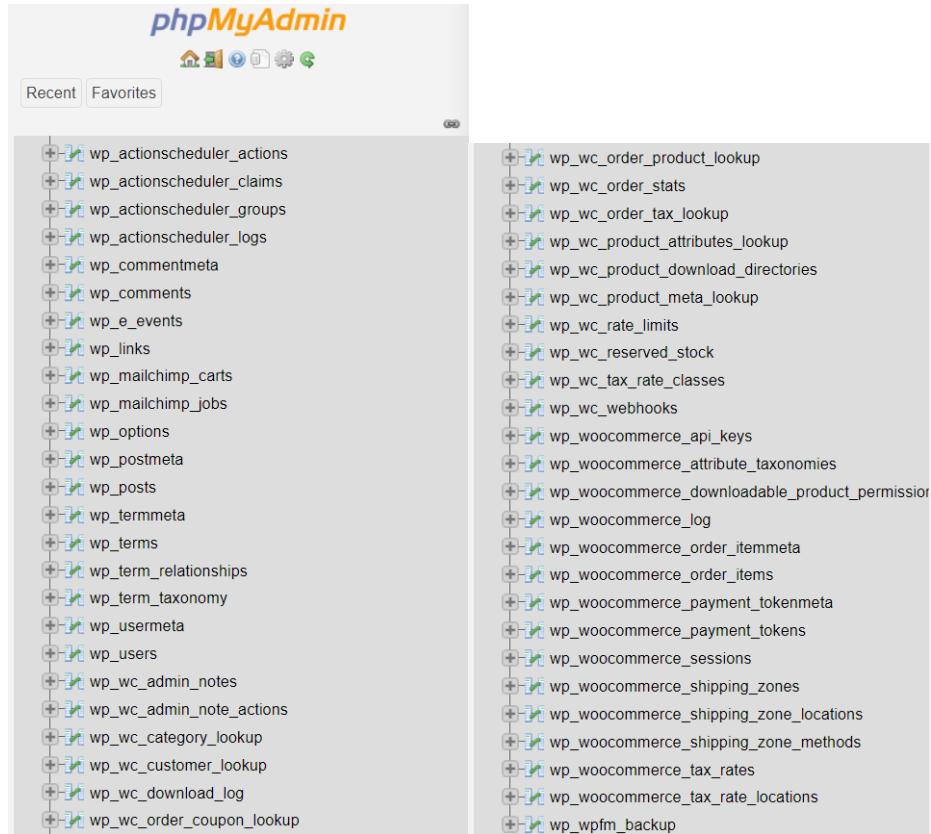


Figura 3.5: Tablas de base de datos de la página web.
Fuente: Elaborado por el investigador

3.2.2.5. Tarjetas CRC

Las tarjetas CRC (Clase-Responsabilidades-Colaboradores) permiten conocer cuales clases u objetos componen el sistema y cuales de ellas interactúan entre sí.

Tabla 3.59: Tarjeta CRC - Usuario

Clase: Usuario	
Responsabilidades	Colaboradores
Ingresar usuarios.	
Asignar roles de usuario.	
Asignar proveedores y controladores a los productos existentes.	Productos.

Fuente: Elaborado por el investigador

Tabla 3.60: Tarjeta CRC - Producto

Clase: Producto	
Responsabilidades	Colaboradores
Ingresar productos.	
Listar productos.	

Fuente: Elaborado por el investigador

Tabla 3.61: Tarjeta CRC - Sucursal

Clase: Sucursal	
Responsabilidades	Colaboradores
Ingresar sucursales.	
Listar sucursales.	
Gestionar sucursales de productos.	Productos.

Fuente: Elaborado por el investigador

Tabla 3.62: Tarjeta CRC - Ubicación

Clase: Ubicación	
Responsabilidades	Colaboradores
Ingresar ubicaciones.	
Listar ubicaciones.	
Gestionar ubicaciones de productos.	Productos

Fuente: Elaborado por el investigador

3.2.2.6. Prototipos de interfaces

Los prototipos de las interfaces son fundamentales para tener una idea de como va a ser el sistema una vez finalizado las iteraciones respectivas, para la creación de los prototipos se utilizó adobexd.

Prototipo del módulo de inicio de sesión

El prototipo de inicio de sesión se construyó en base a diseños ya establecidos por defecto, como son el campo de correo electrónico y la contraseña de la cuenta y un botón para iniciar sesión.

WooMap Inicio Registro Iniciar sesión

Iniciar sesión

Email:

Contraseña:

Iniciar sesión

Figura 3.6: Prototipo de interfaz de login
Fuente: Elaborado por el investigador

Prototipo del módulo de registro

Para el prototipo de registro se consideraron los campos establecidos en los requerimientos.

WooMap Inicio Registro Iniciar sesión

Registro de usuario

Nombre: Apellido:

Cédula: Teléfono:

Clave: Repetir Clave:

Email:

Registrarse

Figura 3.7: Prototipo de interfaz de registro de usuario
Fuente: Elaborado por el investigador

Prototipo del módulo de inicio

El prototipo para la interfaz de inicio del sistema web, fue desarrollado con la idea

de que el usuario al momento de ingresar al sistema se le muestre directamente las ubicaciones de las sucursales existentes.



Figura 3.8: Prototipo de interfaz de inicio
Fuente: Elaborado por el investigador

Existen tres tipos de usuarios y cada uno de ellos tendrá diferentes ventanas a mostrarse, por eso se ha desarrollado diferentes prototipos separados por el rol que cualquier usuario puede tener.

Los prototipos desarrollados para los tres tipos de usuarios de igual manera siguen un estándar de colores en sus interfaces.

Prototipos para el rol administrador

- Prototipo del módulo para gestionar el personal



Figura 3.9: Prototipo de interfaz de gestión de personal
Fuente: Elaborado por el investigador

- Prototipo del módulo para administrar puntos de distribución



Figura 3.10: Prototipo de interfaz para administrar puntos de distribución
Fuente: Elaborado por el investigador

- Prototipo del módulo de gestión de usuarios



Figura 3.11: Prototipo de interfaz de gestión de usuarios
Fuente: Elaborado por el investigador

Prototipos para el rol controlador

- Prototipo del módulo para administrar puntos de distribución por producto



Figura 3.12: Prototipo de interfaz para administrar puntos de distribución por producto
Fuente: Elaborado por el investigador

- Prototipo del módulo para validar ubicaciones



Figura 3.13: Prototipo de interfaz para validar ubicaciones
Fuente: Elaborado por el investigador

Prototipos para el rol proveedor

- Prototipo del módulo para asignar ubicaciones a un producto



Figura 3.14: Prototipo de interfaz para asignar ubicaciones a un producto
Fuente: Elaborado por el investigador

3.2.2.7. Diseño sistema web

Sistema web e-commerce

Pantalla de inicio

En la pantalla de inicio se muestra resumidamente la información de la tienda virtual.

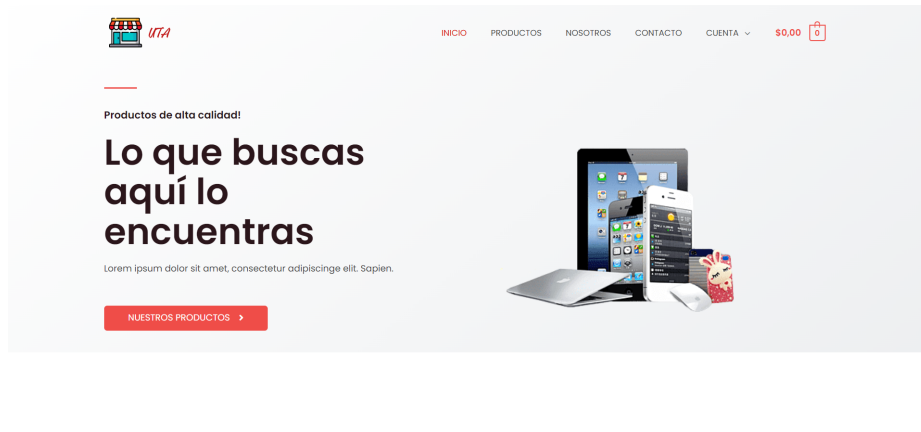


Figura 3.15: Pantalla de inicio de tienda virtual
Fuente: Elaborado por el investigador

Pantalla portafolio de productos

En esta pantalla se muestran los productos que la tienda virtual tiene disponibles, también se tiene diferentes filtros para encontrar el producto que se desee.

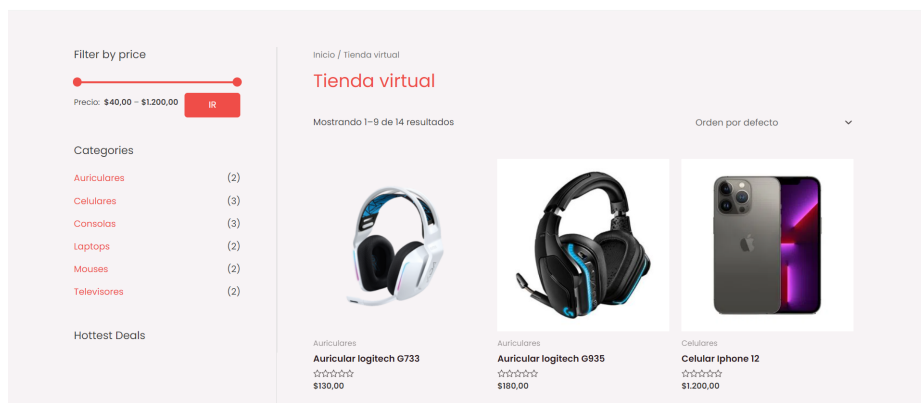


Figura 3.16: Pantalla portafolio de productos
Fuente: Elaborado por el investigador

Pantalla detalle de producto

En esta pantalla se muestra la información mas detallada de un producto seleccionado, tambien se tiene dos botones que permiten añadir al carrito de compras y visualizar la información georeferenciada del producto.

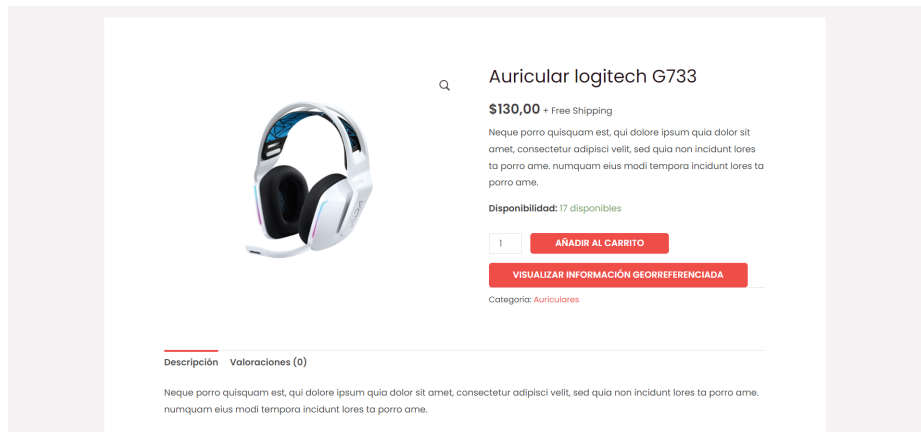


Figura 3.17: Pantalla detalle de producto
Fuente: Elaborado por el investigador

Pantalla de panel de administración de wordpress

En esta pantalla se tiene a disposición todos los complementos que permite utilizar wordpress para crear una página web, así como la edición de páginas, instalación de plugins, creación de entradas para un blog, entre otras.

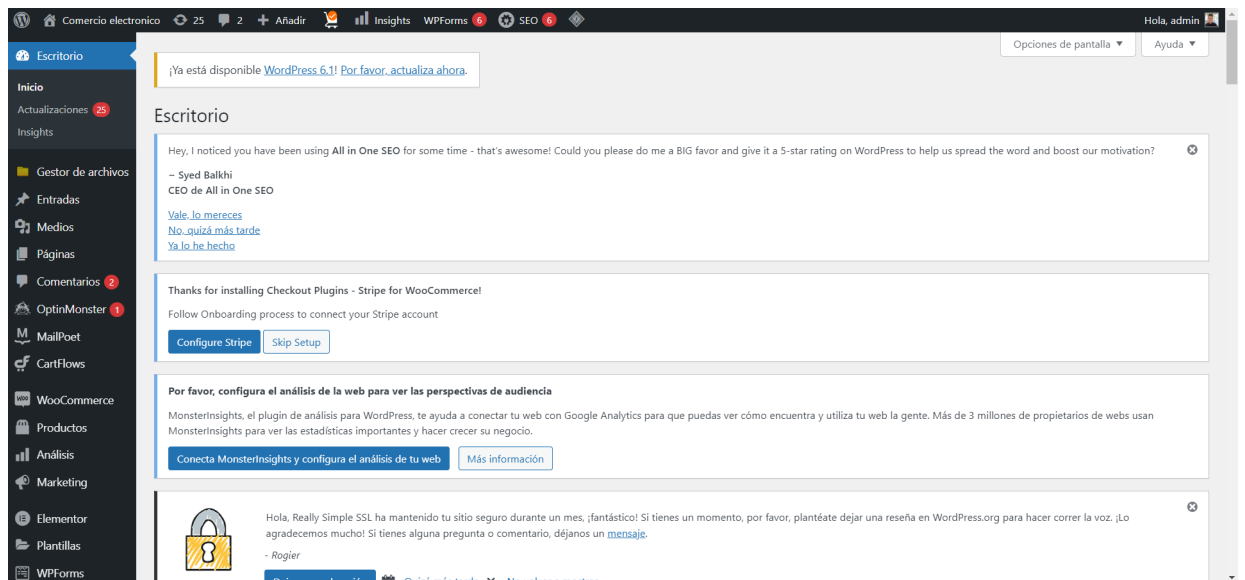


Figura 3.18: Pantalla de administración de wordpress
Fuente: Elaborado por el investigador

Pantalla de estadísticas para ventas realizadas

En esta pantalla se puede visualizar con detalle las ventas que haya tenido la página web, también tiene gráficos estadísticos para visualizar con mayor exactitud el incremento o disminución de ventas.

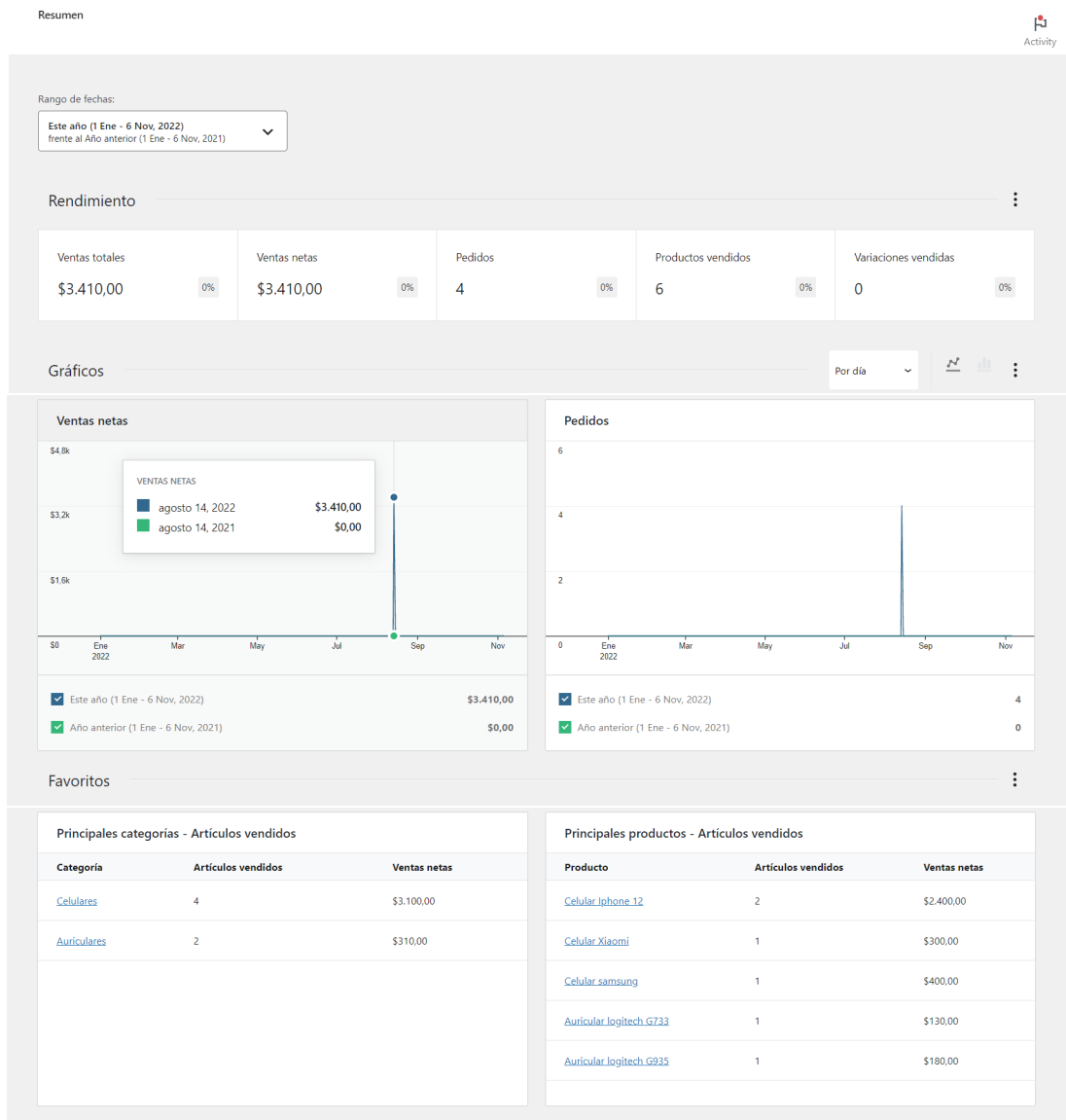


Figura 3.19: Pantalla de administración de wordpress
 Fuente: Elaborado por el investigador

Sistema web woomap

Pantalla de inicio del sistema web

En la pantalla de inicio se muestran con mayor detalle los puntos de distribución disponibles de la empresa.

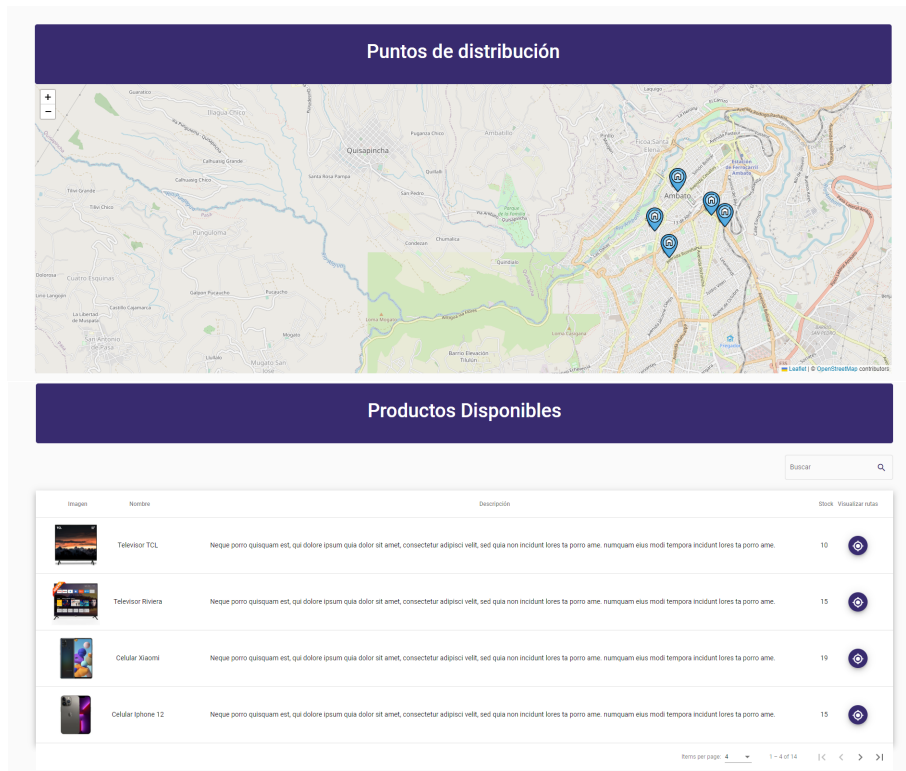


Figura 3.20: Pantalla de inicio del sistema web
Fuente: Elaborado por el investigador

Pantalla de registro de nuevo usuario

La pantalla de registro de usuario contendrá todos los datos necesarios para crear un nuevo usuario dentro del sistema.

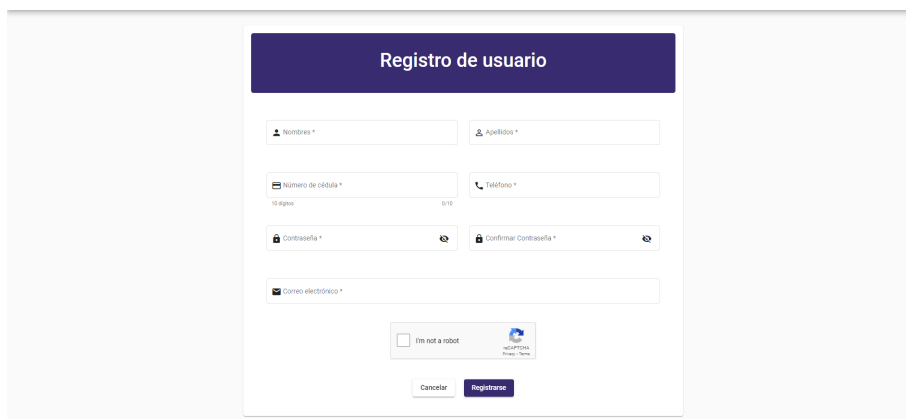


Figura 3.21: Pantalla de registro de nuevo usuario
Fuente: Elaborado por el investigador

Pantalla de ingreso al sistema

En la pantalla para ingresar al sistema se mostrará dos campos obligatorios para iniciar sesión que serán el correo electrónico y una contraseña.



Figura 3.22: Pantalla de ingreso al sistema
Fuente: Elaborado por el investigador

Pantallas de administrador

El usuario administrador tendrá dos pantallas disponibles, la primera pantalla será administrar usuarios y la segunda administrar puntos de distribución.

Pantalla para administrar usuarios

Esta pantalla muestra de una manera detallada la cantidad de usuarios que existen en el sistema, además se tiene un listado detallado con la información de cada usuario y la posibilidad de asignar o remover roles a un usuario específico, se puede activar o desactivar una cuenta de usuario y editar la información que posea dicho usuario.



The screenshot displays a dashboard for 'Usuarios Registrados'. At the top, there are four summary cards: 'Total de usuarios: 9' (purple), 'Total de administradores: 2' (blue), 'Total de proveedores: 7' (orange), and 'Total de controladores: 5' (grey). Below these is a table of registered users with columns for 'Cédula', 'Nombre', 'Apellido', 'Correo electrónico', 'Teléfono', 'Administrador', 'Proveedor', 'Contributor', 'Cuenta activada / desactivada', and 'Acciones'. The table contains four rows of user data.

Cédula	Nombre	Apellido	Correo electrónico	Teléfono	Administrador	Proveedor	Contributor	Cuenta activada / desactivada	Acciones
1726844689	Paulina	Torres	pg@gmail.com	0983276532	✗	✓	✓	✓	⚙️
1803925056	William	Almagro	w@gmail.com	0979127612	✗	✓	✗	✓	⚙️
1804629226	Sebastian	Mayorga	sebasmayorga@gmail.com	0962692697	✓	✗	✓	✓	⚙️
1804898755	Ernesto	Suarez	e@gmail.com	0872356327	✗	✗	✓	✓	⚙️

Figura 3.23: Pantalla para administrar usuarios
Fuente: Elaborado por el investigador

Pantalla para administrar los puntos de distribución

Esta pantalla permitirá al usuario crear, eliminar, modificar y visualizar los puntos de distribución creados en el sistema.

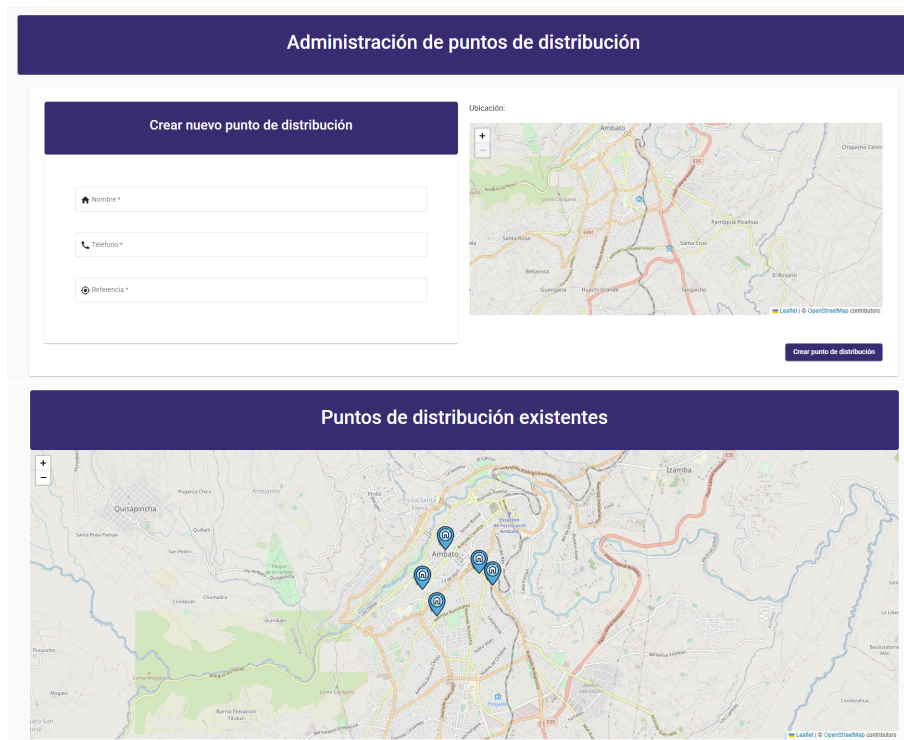


Figura 3.24: Pantalla para administrar los puntos de distribución
Fuente: Elaborado por el investigador

Pantalla para administrar el personal por producto

Esta pantalla es la encargada de asignar proveedores como controladores a un producto para su respectivo recorrido.

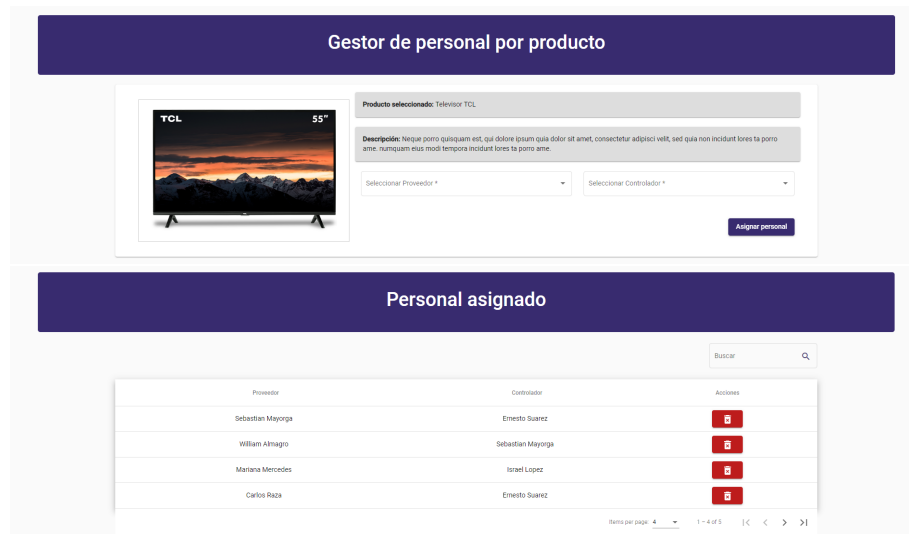


Figura 3.25: Pantalla para administrar el personal por producto
Fuente: Elaborado por el investigador

Pantalla de proveedor

Pantalla para asignar ubicaciones de un producto

Esta pantalla permite al usuario asignar el recorrido que haya realizado con un producto.

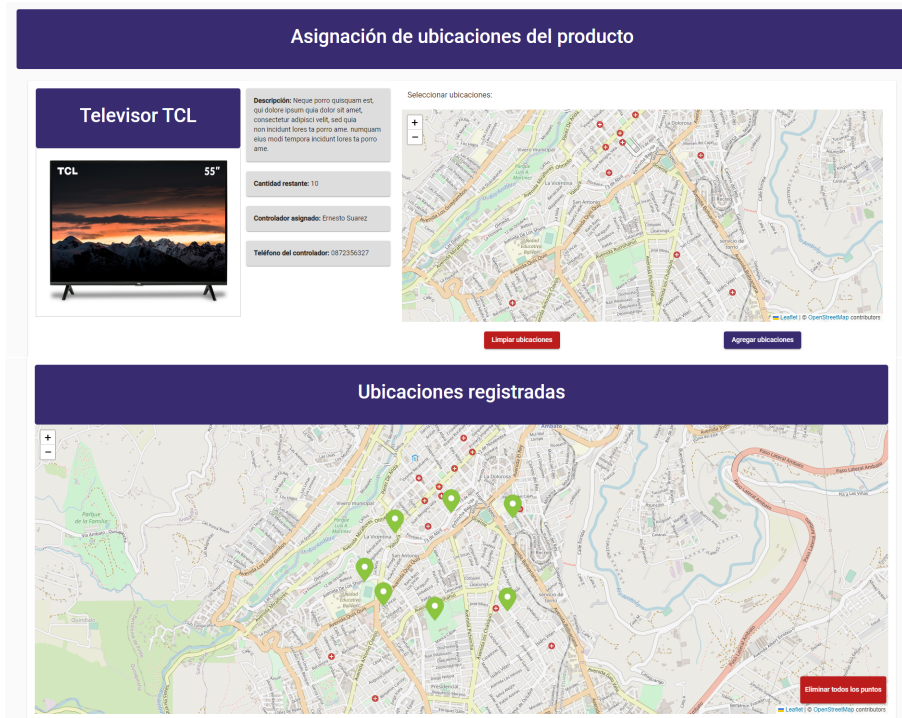


Figura 3.26: Pantalla para asignar ubicaciones de un producto
Fuente: Elaborado por el investigador

Pantallas de controlador

Pantalla para validar ubicaciones registradas por un proveedor

En esta pantalla se aprueban o no las ubicaciones que un proveedor haya registrado, se puede registrar un comentario en el caso de que sea necesario.

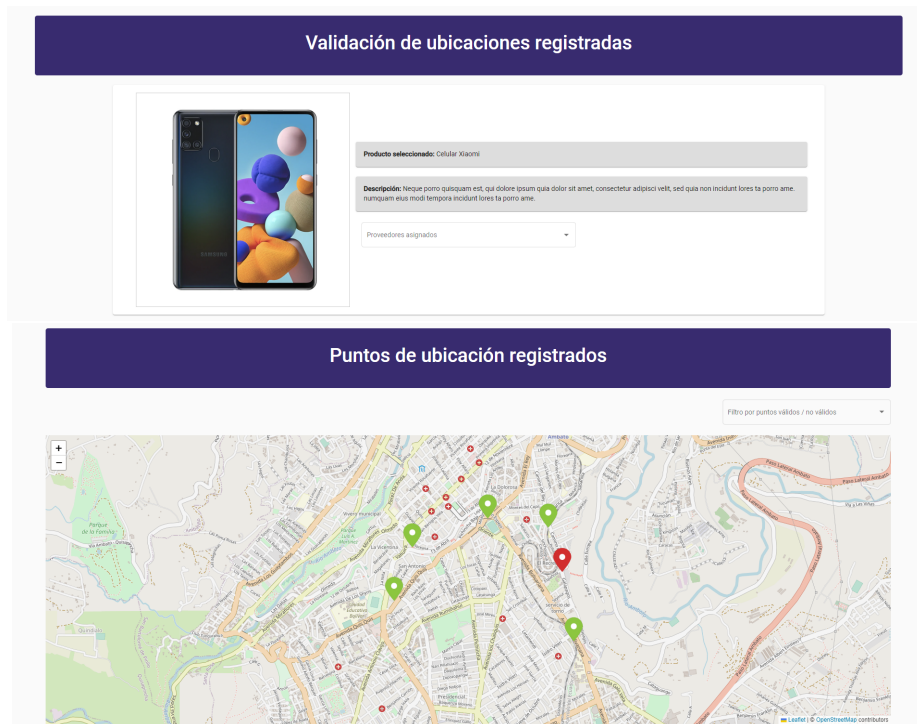


Figura 3.27: Pantalla para validar ubicaciones registradas por un proveedor
Fuente: Elaborado por el investigador

Pantalla para designar puntos de distribución a un producto

En esta pantalla se asignan puntos de distribución a un producto.

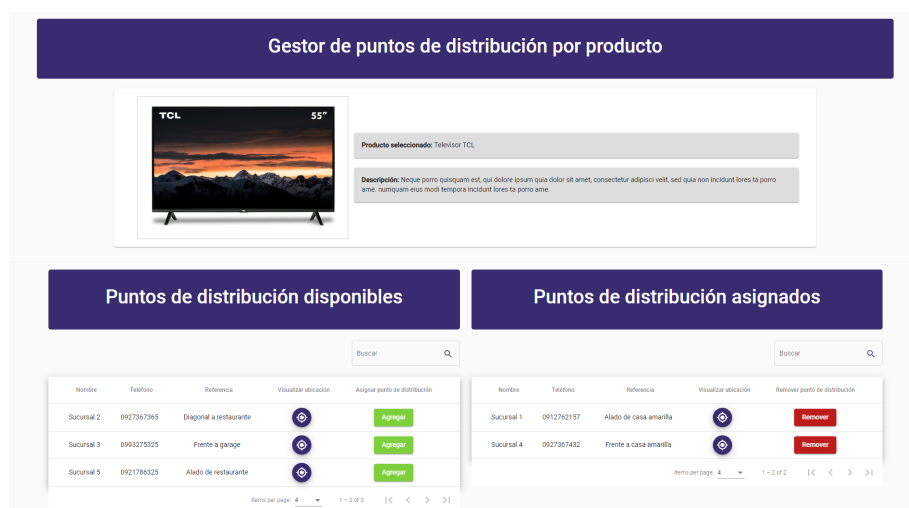


Figura 3.28: Pantalla para designar puntos de distribución a un producto
Fuente: Elaborado por el investigador

3.2.3. Fase 3: Codificación

3.2.3.1. Comandos para crear el proyecto

Para la creación del proyecto de angular que nos sirvió para el frontend se utilizó el siguiente comando “ng new nameProject”.

```
PS C:\Users\sebas\OneDrive\Escritorio\Tesis Sebas\Programacion> ng new proyecto-seguimiento-productos
```

Figura 3.29: Comando para crear el proyecto angular
Fuente: Elaborado por el investigador

Del lado del servidor se ejecutó el siguiente comando que nos permitirá inicializar una aplicación de node js.

```
PS C:\Users\sebas\OneDrive\Escritorio\Tesis Sebas\Programacion\rest-api> npm init
```

Figura 3.30: Comando para crear una aplicación de nodejs
Fuente: Elaborado por el investigador

Tambien se instalaron los módulos de sequelize, body-parser, mysql2 y express utilizando el comando “npm install express sequelize mysql2 body-parser cors --save”

```
PS C:\Users\sebas\OneDrive\Escritorio\Tesis Sebas\Programacion\proyecto-seguimiento-productos> npm install express sequelize mysql2 body-parser cors --save
```

Figura 3.31: Comando para instalar módulos en la aplicación de node js
Fuente: Elaborado por el investigador

3.2.3.2. Modificación de interfaz en wordpress

Para modificar una interfaz o un conjunto de interfaces en wordpress es necesario establecerlo en el archivo functions.php, cabe resaltar que al utilizar woocommerce como plugin para tener una tienda virtual se puede obtener la id del producto que se encuentre seleccionado utilizando la función `get_id()`.

```
13 add_action( 'woocommerce_single_product_summary', 'addExtraButtonInProductDetail', 30 );
14
15 function addExtraButtonInProductDetail() {
16     global $product;
17     $id = $product->get_id();
18     $URL = "http://localhost:4200/productview/".$id;
19     echo '<a href="'.$URL.'" class="button" target="_blank">Visualizar información georeferenciada</a>';
20 }
```

Figura 3.32: Código para agregar un botón a la interfaz de un tema de wordpress
Fuente: Elaborado por el investigador

3.2.3.3. Conexión a base de datos

El sistema web utiliza dos conexiones a diferentes base de datos, la primera conexión es la que utiliza wordpress para su sistema de comercio electrónico y la segunda conexión es la que utiliza el sistema de woomap para la persistencia de sus datos.

```
1 require("dotenv").config();
2 module.exports = {
3   HOST: process.env.HOST,
4   USER: process.env.USER,
5   PASSWORD: process.env.PASSWORD,
6   DB: process.env.DATABASE,
7   dialect: "mysql",
8   pool: {
9     max: 5,
10    min: 0,
11    acquire: 30000,
12    idle: 10000,
13  },
14 };
```

Figura 3.33: Conexión a base de datos de woomap

Fuente: Elaborado por el investigador

```
1 require("dotenv").config();
2 module.exports = {
3   HOST: process.env.HOSTSTORE,
4   USER: process.env.USERSTORE,
5   PASSWORD: process.env.PASSWORDSTORE,
6   DB: process.env.DATABASESTORE,
7   dialect: "mysql",
8   pool: {
9     max: 5,
10    min: 0,
11    acquire: 30000,
12    idle: 10000,
13  },
14 };
```

Figura 3.34: Conexión a base de datos de wordpress

Fuente: Elaborado por el investigador

Como se puede observar en la figura 3.33 y 3.34, en la conexión a las bases de datos se especificaron variables de entorno, que se instancian en un archivo llamado .env y se encuentra localizado en el proyecto, además se tiene configurado un pool el cual es un conjunto que tiene un mínimo y un máximo de conexiones manejadas por el servidor, de esta manera los usuarios pueden utilizarlas al realizar múltiples consultas o actualizaciones en la información almacenada.

Inserción de productos en base de datos de woomap

El siguiente método ejecuta una tarea programada cada minuto, comprobando la existencia de los productos de la base de datos de wordpress en la base de datos de woomap y de esta forma se crean, actualizan o se eliminan uno o varios productos simultáneamente, de manera que en el sistema web se reflejen dichos cambios.

```

56 exports.signin = (req, res) => {
57   User.findOne({
58     where: {
59       email: req.body.email,
60     },
61   })
62   .then((user) => {
63     if (!user) {
64       return res
65         .status(404)
66         .send({
67           message: "El email proporcionado no se encuentra registrado.",
68         });
69     }
70     if (user.active == 0) {
71       return res
72         .status(404)
73         .send({
74           message: "No puedes iniciar sesión, tu cuenta debe estar activada.",
75         });
76     }
77
78     var passwordIsValid = bcrypt.compareSync(
79       req.body.password,
80       user.password
81     );
82     if (!passwordIsValid) {
83       return res.status(401).send({
84         accessToken: null,
85         message: "La contraseña ingresada es incorrecta!",
86       });
87     }
88     var token = jwt.sign({ card: user.card }, config.secret, {
89       expiresIn: 86400, // 24 horas
90     });
91     var authorities = [];
92     user.getRoles().then((roles) => {
93       for (let i = 0; i < roles.length; i++) {
94         authorities.push(roles[i].name);
95       }
96       res.status(200).send({
97         card: user.card,
98         names: user.names,
99         lastNames: user.lastNames,
100        email: user.email,
101        phone: user.phone,
102        active: user.active,
103        roles: authorities,
104        accessToken: token,
105      });
106    });
107  })
108  .catch((err) => {
109    res.status(500).send({ message: err.message });
110  });
111 });

```

Figura 3.35: Método de inicio de sesión
Fuente: Elaborado por el investigador

Los principales métodos de frontend y backend se encuentran descritos en el **Anexo A**, además se tiene a disposición los repositorios que contienen el código fuente del sistema creado.

3.2.4. Fase 4: Pruebas

3.2.4.1. Pruebas de aceptación

Las pruebas de aceptación permiten validar la funcionalidad del sistema web, permitiendo al usuario final verificar el cumplimiento de los requerimientos establecidos y de esta manera se pueda realizar mejoras o correcciones.

Tabla 3.63: Prueba de aceptación 1

Prueba de aceptación	Número: 1
N° de historia de usuario: 002	
Nombre: Comercio electrónico	
Descripción: En esta página web se puede visualizar los productos que se tienen disponibles, su información correspondiente y también permite comprar uno o varios productos.	
Condiciones de ejecución: Al seleccionar un producto mostrará la información respectiva como su nombre, descripción y precio, además se tendrá dos botones uno para agregar al carrito de compras y otro para visualizar su información georeferenciada.	
Interfaz: La página web estará conformada por un menú de opciones, el cuerpo de la interfaz en donde se mostrará la información que se haya seleccionado y el footer que tendrá enlaces para el fácil acceso del usuario.	
Resultado esperado: La página web podrá mostrar todos los productos que se dispongan, se pueden realizar compras y a través del panel de wordpress se puede llevar un control detallado de las ventas realizadas.	
Resultado de la prueba: Prueba satisfactoria	

Fuente: Elaborado por el investigador

Tabla 3.64: Prueba de aceptación 2

Prueba de aceptación	Número: 2
N° de historia de usuario: 004	
Nombre: Productos e información georeferenciada	
Descripción: Módulo que muestra los productos que se tengan disponibles, su información y datos georeferenciados como las ubicaciones que un proveedor ha recorrido, los puntos de distribución que tengan disponibilidad del producto.	
Condiciones de ejecución: Al ingresar a la información de un producto, esta mostrará toda su información cargada, al seleccionar un proveedor toda la información georeferenciada cambiará por la información registrada por parte del proveedor seleccionado.	
Interfaz: La pantalla muestra una imagen del producto, su información, también contiene dos mapas en uno se encuentran ubicados los puntos de distribución que tienen disponibles el producto y en el otro mapa se encuentran las ubicaciones registradas por parte del proveedor seleccionado.	
Resultado esperado: Al seleccionar un producto toda su información debe cargarse inmediatamente, además si se selecciona un proveedor la información georeferenciada del producto debe cambiar.	
Resultado de la prueba: Prueba satisfactoria	

Fuente: Elaborado por el investigador

Tabla 3.65: Prueba de aceptación 3

Prueba de aceptación	Número: 3
N° de historia de usuario: 005	
Nombre: Registro en el sistema	
Descripción: Módulo que permite el registro de nuevos usuarios en el sistema, se registraran datos importantes como los nombres, apellidos, cédula, teléfono, contraseña y correo electrónico.	
Condiciones de ejecución: Todos los campos deben ser completados para proceder con el registro del usuario, el número de cédula y correo electrónico deben ser únicos y no tienen que estar registrados en el sistema, cabe resaltar que el número de cédula tiene que ser válido, además se validará el usuario a través de reCAPTCHA para comprobar que no sea un robot.	
Interfaz: La pantalla muestra un total de siete textbox que permite al usuario ingresar su información, también una checkbox que valida a un usuario como persona, también tiene dos botones uno para cancelar la acción del registro y otro para que el usuario pueda registrarse.	
Resultado esperado: Al ingresar toda la información del usuario y validar que no es un robot, el usuario obtendrá un mensaje de confirmación que su usuario fue creado.	
Resultado de la prueba: Prueba satisfactoria	

Fuente: Elaborado por el investigador

Tabla 3.66: Prueba de aceptación 4

Prueba de aceptación	Número: 4
N° de historia de usuario: 006	
Nombre: Acceso al sistema	
Descripción: Módulo que permite al usuario ingresar al sistema a través de sus credenciales.	
Condiciones de ejecución: El usuario debe ingresar el correo electrónico y la contraseña que haya ingresado en el registro.	
Interfaz: La pantalla muestra dos campos de texto uno para que se ingrese el correo electrónico y otro para ingresar una contraseña, también existe un botón que realiza la acción de iniciar sesión.	
Resultado esperado: El usuario ingresa sus credenciales de inicio de sesión, se comprueba su existencia y si su cuenta se encuentra activada podrá ingresar al sistema caso contrario le aparecerá un mensaje de error.	
Resultado de la prueba: Prueba satisfactoria	

Fuente: Elaborado por el investigador

Tabla 3.67: Prueba de aceptación 5

Prueba de aceptación	Número: 5
N° de historia de usuario: 007	
Nombre: Administrar usuarios	
Descripción: Módulo que permite al administrador visualizar la cantidad de usuarios y administrarlos.	
Condiciones de ejecución: El usuario administrador podrá visualizar el total de usuarios registrados en el sistema, los usuarios que son administradores, proveedores y controladores, además puede asignar roles a cualquier usuario, activar o desactivar cuentas de usuario y modificar información de cualquier usuario registrado.	
Interfaz: La pantalla cuenta con cuatro cajas de diferentes colores que muestran la cantidad de usuarios, el total de administradores, proveedores y controladores, también tiene una tabla que lista a todos los usuarios registrados en el sistema, en cada registro listado se tiene diferentes botones para asignar roles, activar o desactivar una cuenta y modificar la información de un usuario específico.	
Resultado esperado: El usuario administrador visualizará la información de la cantidad de cuentas del sistema, el total de usuarios pertenecientes a cada rol existente y puede modificar los roles, la activación o desactivación de cuentas y la información de cualquier usuario listado.	
Resultado de la prueba: Prueba satisfactoria	

Fuente: Elaborado por el investigador

Tabla 3.68: Prueba de aceptación 6

Prueba de aceptación	Número: 6
N° de historia de usuario: 008	
Nombre: Gestión de puntos de distribución	
Descripción: Módulo que permite administrar puntos de distribución en el sistema.	
Condiciones de ejecución: El usuario administrador podrá registrar, visualizar, modificar o eliminar puntos de distribución del sistema en caso de ser necesario.	
Interfaz: La pantalla muestra los campos necesarios para registrar un punto de distribución, también un mapa para establecer la ubicación y otro mapa para visualizar los puntos de distribución existentes.	
Resultado esperado: El usuario administrador al ingresar en el módulo puede registrar nuevos puntos de distribución, modificar los puntos existentes y eliminarlos.	
Resultado de la prueba: Prueba satisfactoria	

Fuente: Elaborado por el investigador

Tabla 3.69: Prueba de aceptación 7

Prueba de aceptación	Número: 7
N° de historia de usuario: 009	
Nombre: Gestión de personal por producto	
Descripción: Módulo que permite gestionar el personal encargado de un producto en específico.	
Condiciones de ejecución: El usuario administrador escogerá un proveedor y un controlador para asignarlos a un producto del sistema, además podrá eliminar las asignaciones realizadas.	
Interfaz: La pantalla tendrá dos combobox pertenecientes a los proveedores y controladores del sistema, además tiene una tabla que lista todas las asignaciones de personal realizadas al producto seleccionado.	
Resultado esperado: El usuario administrador puede asignar nuevo personal al producto, si un proveedor esta asignado ya no puede asignarse nuevamente, el listado del personal asignado deberá aparecer una vez cargada la pantalla, al eliminar un personal listado mostrará un mensaje de confirmación.	
Resultado de la prueba: Prueba satisfactoria	

Fuente: Elaborado por el investigador

Tabla 3.70: Prueba de aceptación 8

Prueba de aceptación	Número: 8
N° de historia de usuario: 010	
Nombre: Asignar ubicaciones por producto	
Descripción: Módulo que le permite a un proveedor asignar ubicaciones a un producto que se le haya asignado.	
Condiciones de ejecución: El usuario proveedor ingresará las ubicaciones por donde haya recorrido con el producto, posteriormente en el mapa de visualización aparecerán reflejadas estas ubicaciones con la diferencia que se tornarán de color verde si se encuentran aprobadas o color rojas si no están aprobadas por un usuario controlador.	
Interfaz: La pantalla muestra la información del producto seleccionado, un mapa para registrar las ubicaciones y un mapa para visualizar las ubicaciones registradas.	
Resultado esperado: Si el usuario proveedor ingresa a la interfaz puede registrar varios puntos de ubicaciones, cuando realice el registro aparecerán reflejadas en el mapa de visualización, puede eliminar uno o todos los puntos registrados.	
Resultado de la prueba: Prueba satisfactoria	

Fuente: Elaborado por el investigador

Tabla 3.71: Prueba de aceptación 9

Prueba de aceptación	Número: 9
N° de historia de usuario: 011	
Nombre: Validar ubicaciones	
Descripción: Módulo que le permite a un controlador validar ubicaciones registradas por un proveedor.	
Condiciones de ejecución: El usuario controlador debe seleccionar un producto, posteriormente debe escoger un proveedor y se le cargarán todas las ubicaciones registradas por parte del proveedor, al presionar sobre una ubicación podrá validar o no el punto registrado y dejar una observación de ser el caso necesario.	
Interfaz: La pantalla cuenta con una imagen del producto, su información un combobox que contiene los proveedores asignados al producto y un mapa para visualizar las ubicaciones registradas.	
Resultado esperado: El usuario controlador puede escoger cualquier proveedor y validar las ubicaciones registradas por el mismo, además puede ingresar una observación a través de un textarea.	
Resultado de la prueba: Prueba satisfactoria	

Fuente: Elaborado por el investigador

Tabla 3.72: Prueba de aceptación 10

Prueba de aceptación	Número: 10
N° de historia de usuario: 012	
Nombre: Designar puntos de distribución	
Descripción: Módulo que permite a un usuario controlador asignar puntos de distribución registrados en el sistema a un producto en específico.	
Condiciones de ejecución: El usuario controlador escogerá un producto, podrá asignar puntos de distribución o remover los asignados al producto, también visualizará la ubicación de cualquier punto de distribución.	
Interfaz: La pantalla cuenta con la imagen del producto, su información, dos tablas que mostrarán los puntos de distribución disponibles y los puntos asignados al producto seleccionado, cada registro listado tendrá dos botones uno para visualizar la ubicación de cualquier punto y otro para agregarlo o removerlo.	
Resultado esperado: El usuario controlador al seleccionar un producto se le cargará toda la información del producto, sus puntos de distribución asignados y los puntos de distribución disponibles, si presiona en visualizar ubicación se le desplegará la ubicación del punto seleccionado, además podrá asignar o remover puntos de distribución.	
Resultado de la prueba: Prueba satisfactoria	

Fuente: Elaborado por el investigador

3.2.4.2. Evaluación heurística con los principios de Jakob Nielsen

Para validar el mecanismo implementado se utilizó una evaluación de usabilidad, la cual se creó en base a los diez principios heurísticos de Nielsen. El cuestionario del **Anexo B** se aplicó a 3 expertos en el área y se obtuvieron los siguientes resultados:

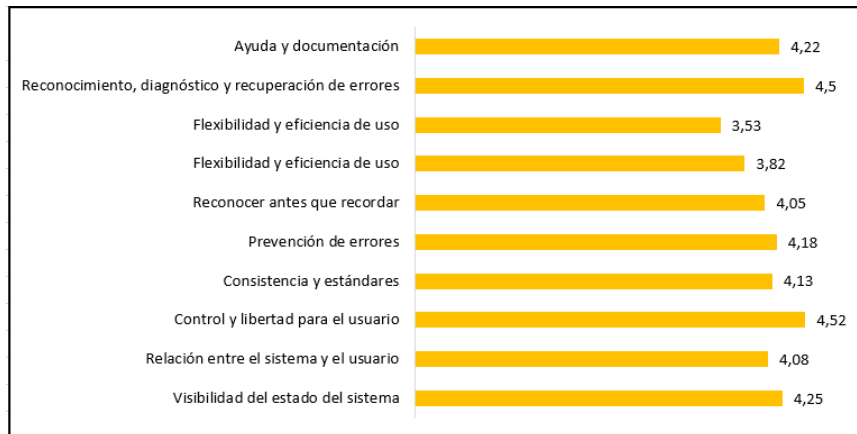


Figura 3.36: Resultados de la evaluación heurística

Fuente: Elaborado por el investigador

Los resultados que se muestran en la imagen 3.36 fueron obtenidos a través del promedio de la valoración de cada pregunta del cuestionario expuesto en el **Anexo B**, dicha valoración fue brindada en base a la validación del sistema realizado por cada experto. A través de estos resultados obtenidos se pudo concluir que el promedio para todas las preguntas de cada principio es de 4.13 sobre una calificación de 5, teniendo una usabilidad del 82.57%, demostrando de esta manera que el sistema cumple con la mayoría de los estándares de usabilidad establecidos por Nielsen.

CAPÍTULO IV

CONCLUSIONES Y RECOMENDACIONES

4.1. Conclusiones

El desarrollo del presente proyecto permitió definir un mecanismo de georeferenciación para la cadena de distribución de una iniciativa de comercio electrónico, teniendo como conclusión lo siguiente:

- Se definió leaflet como mecanismo de georeferenciación, mismo que fue implementado en el prototipo funcional del sistema woomap y probado por un grupo de 14 personas que realizan compras en línea; a través de esta prueba del sistema se obtuvo información mediante una encuesta, con la cual se determinó que el 93% de esta población están de acuerdo que implementar un mecanismo de georeferenciación en una plataforma de comercio electrónico permite incrementar las ventas en línea.
- Se investigaron los mecanismos de georeferenciación que se puede utilizar en un sistema de comercio electrónico basado en software libre de los cuales se seleccionó leaflet debido a que al realizar una comparación con las otras tecnologías este ofrece más características de customización para su implementación dentro de un sistema web.
- La implementación del sistema web como del prototipo funcional se realizó satisfactoriamente mediante su despliegue en línea, además como se pudo observar en la presente investigación para la creación de dicho prototipo se utilizaron tecnologías de software libre.
- El sistema web y el sistema de comercio electrónico fueron validados a través de pruebas de uso que fueron realizadas por usuarios expertos en el tema, siguiendo una evaluación heurística que utiliza los 10 principios de Jakob Nielsen para evaluar la usabilidad del sistema web.

4.2. Recomendaciones

- Se recomienda a la unidad de investigación leer el manual de usuario adjunto al presente proyecto para que de esta manera no exista confusiones al utilizar

el sistema.

- Se recomienda tener precaución en el manejo de las conexiones a las bases de datos que utiliza el sistema, debido a que son dos conexiones distintas, una para obtener los productos del sistema de comercio electrónico y otra para la persistencia de datos del sistema desarrollado.
- Se recomienda seguir utilizando una herramienta de control de versiones como es el caso de github, con la finalidad de tener el código siempre respaldado y que de esta manera no se pueda perder o inclusive si llega a existir un error en el sistema web poder restablecerlo a una versión anterior.

Bibliografía

- [1] S. Mdz, “Conceptos generales de la arquitectura de aplicaciones web,” 2021.
- [2] M. soluciones de negocio, “Metodologías de desarrollo de software,” 2020.
- [3] T. soluciones, “10 razones para usar la metodología kanban en tu organización,” 2021.
- [4] M. B. Oriols and J. A. G. Gutiérrez, “El gran libro de angular,” 2020.
- [5] S. P. Estévez, N. R. Lamela, and C. A. Morales, “Api de servicios web orientados a accesibilidad,” 2016.
- [6] B. S. T. Barrionuevo, “Aplicación móvil multiplataforma para la gestión de información georreferenciada de sitios turísticos de interés social y cultural en la ciudad de ambato,” 2020.
- [7] J. P. T. Bastidas, “Aplicación móvil multiplataforma para la gestión de información georreferencial y servicio técnico comunitario de plomería, aplicando geolocalización offline, en la junta administradora de agua potable de los barrios occidentales de aloasí,” 2021.
- [8] W. P. C. Chirán, “Aplicación móvil con geolocalización para la gestión y monitorización de los comerciantes catastrados y autónomos regularizados del gobierno autónomo descentralizado municipal de ambato (gadma),” 2021.
- [9] D. X. C. Calvopiña, “Desarrollo de un aplicativo móvil que integre capacidades de visualización geográfica y georreferenciación para mejorar la perspectiva de los atractivos turísticos de la ciudad latacunga,” 2018.
- [10] H. Selman, *Marketing Digital*. No. 1944278931, Ibukku, 2017.
- [11] J. Ramos, *Marketing de contenidos*. No. 15190505421, Independently published, 2016.
- [12] S. Dakduk, R. Dicarlo, L. Ottati, and A. Portalanza, “Situación de las empresas durante el covid-19 ecuador,” 2020.

- [13] V. de comercio exterior, “Covid-19 en el ecuador impacto económico y perspectivas,” 2020.
- [14] J. R. Fernández and J. N. Noles, “Diseño y desarrollo de interfaz gráfica (front end): formularios, controladores y lógica de negocio para el sistema ”gearh grupo elastix administración de recursos humanos.^{en} palosanto solutions,” 2017.
- [15] E. Vivanco, “Estrategia digital: qué es y cómo se diseña,” 2017.
- [16] K. Rodríguez, O. Ortiz, A. Quiroz, and M. Parrales, “El e-commerce y las mipymes en tiempos de covid-19 e-commerce and msms in times of covid-19,” 2020.
- [17] K. Reinoso and J. Jordan, “Marketing digital como una ventaja competitiva para las pymes en pandemia tungurahua - ecuador,” 2021.
- [18] M. G. V. Toapanta, “E-commerce y su impacto en el desarrollo económico de las pequeñas y medianas empresas de la provincia de tungurahua,” 2016.
- [19] F. G. P. Montesdeoca, “Página web y su incidencia en el volumen de ventas del almacén de electrodomésticos credicomercio cevallos, en el cantón cevallos de la provincia de tungurahua,” 2015.
- [20] I. Coral and C. Iza, “Análisis, diseño y construcción de un sistema e-commerce para web y dispositivos android,” 2018.
- [21] W. D. L. C. García, “Estrategias de publicidad para incrementar las ventas de calzado buterri en la zona centro del país,” 2015.
- [22] R. Bojorque, “Sistemas gestores de contenido (cms),” 2008.
- [23] G. H. Heber, D. D. Rogelio, P. C. Juan, S. R. M. Alicia, and L. N. Richard, “Sistema informático web de georreferenciación del mapa de delito de la ciudad de huacho,” 2017.
- [24] M. R. V. Pardo, J. A. H. Tapia, A. S. G. Moreno, and L. F. V. Sánchez, “Comparación de tendencias tecnológicas en aplicaciones web,” 2018.
- [25] J. Cervantes Ojeda and M. d. C. Gómez Fuentes, “Taxonomía de los modelos y metodologías de desarrollo de software más utilizadas,” 2020.
- [26] M. H. Bejarno, “Ciclo de vida de desarrollo ágil de software seguro,” 2020.

- [27] A. M. de la Torre Monzón, “Introducción a nodejs a través de koans,” 2013.
- [28] U. Panamericana, “Restful api,” 2020.
- [29] C. Technologies, “Una guía para el diseño de api y rest,” 2015.
- [30] B. Jallow and S. Kashef, “Object relational mapping and entity framework,” 2018.
- [31] S. Fulakeza, “Node js sequelize,” 2016.
- [32] J. G. Checa, “Herramienta para la evaluación y certificación de procesos software,” 2015.
- [33] M. F. L. Cisternas, “Métodos de evaluación de usabilidad para aplicaciones web transaccionales,” 2012.
- [34] C. F. Gallego, “Cálculo del tamaño de la muestra,” 2012.
- [35] H. C. Oviedo and A. Campo-Arias, “Aproximación al uso del coeficiente alfa de cronbach,” 2005.

Anexos

Anexo A

Código de programación

1.1. Métodos del backend

Antes de revisar los principales métodos utilizados en la programación del backend, hay que recordar que se utilizó sequelize como ORM, de esta manera se pudo optimizar tiempos de consulta ya que posee métodos que nos permiten de una manera más fácil obtener, actualizar, crear o eliminar datos.

Los métodos principales que se utilizaron en la programación del backend fueron:

- findAll(): busca múltiples tuplas en el modelo que se necesite.
- findOne(): busca una tupla específica utilizando un parámetro where en donde se instancia una condición con la cual realiza la búsqueda.
- create(): crea una tupla con los datos necesarios de acuerdo al modelo que se necesite.
- bulkCreate(): crea n tuplas enviando un array de datos que tengan la misma estructura que el modelo instanciado.
- update(): actualiza los datos de una tupla enviándole un parámetro where que indica la condición para realizar la modificación del dato o datos.
- destroy(): elimina una tupla de un modelo existente.

Autenticación

Este método verifica que las credenciales que el usuario ingreso, si están correctas genera un token de inicio de sesión, y obtiene los roles que el usuario tenga, caso contrario envía un mensaje de error.

```

56 exports.signin = (req, res) => {
57   User.findOne({
58     where: {
59       email: req.body.email,
60     },
61   })
62   .then((user) => {
63     if (!user) {
64       return res
65         .status(404)
66         .send({
67           message: "El email proporcionado no se encuentra registrado.",
68         });
69     }
70     if (user.active == 0) {
71       return res
72         .status(404)
73         .send({
74           message: "No puedes iniciar sesión, tu cuenta debe estar activada.",
75         });
76     }
77
78     var passwordIsValid = bcrypt.compareSync(
79       req.body.password,
80       user.password
81     );
82     if (!passwordIsValid) {
83       return res.status(401).send({
84         accessToken: null,
85         message: "La contraseña ingresada es incorrecta!",
86       });
87     }
88
89     var token = jwt.sign({ card: user.card }, config.secret, {
90       expiresIn: 86400, // 24 horas
91     });
92     var authorities = [];
93     user.getRoles().then((roles) => {
94       for (let i = 0; i < roles.length; i++) {
95         authorities.push(roles[i].name);
96       }
97       res.status(200).send({
98         card: user.card,
99         names: user.names,
100        lastNames: user.lastNames,
101        email: user.email,
102        phone: user.phone,
103        active: user.active,
104        roles: authorities,
105        accessToken: token,
106      });
107    });
108  })
109  .catch((err) => {
110    res.status(500).send({ message: err.message });
111  });
};

```

Figura 1.1: Método para autenticar un usuario
Fuente: Elaborado por el investigador

Registrar usuario

Encriptar datos sensibles es fundamental en un sistema web, tal es el caso de las contraseñas que los usuarios ingresan al registrarse, la librería que nos ayuda a realizar este proceso es bcryptjs que es una función de hashing basado en el cifrado de blowfish.

```

8 exports.signup = (req, res) => {
9   // Save User to Database
10  const cardUser = req.body.card;
11  User.create({
12    card: cardUser,
13    names: req.body.names,
14    lastNames: req.body.lastNames,
15    email: req.body.email,
16    phone: req.body.phone,
17    password: bcrypt.hashSync(req.body.password, 8),
18    active: 0, // Si el usuario esta activado o no por defecto estara desactivado
19  })
20  .then((user) => {
21    if (req.body.roles) {
22      Role.findAll({
23        where: {
24          name: {
25            [Op.or]: req.body.roles,
26          },
27        },
28      }).then((roles) => {
29        user.setRoles(roles).then(() => {
30          if (roles[0].name == "admin") {
31            const activeUser = {
32              active: 1,
33            };
34            User.update(activeUser, {
35              where: { card: cardUser },
36            });
37          }
38          res.send({
39            message: "El usuario fue registrado satisfactoriamente!",
40          });
41        });
42      });
43    } else {
44      // user role = 2 (provider)
45      user.setRoles([2]).then(() => {
46        res.send({
47          message: "El usuario fue registrado satisfactoriamente!",
48        });
49      });
50    }
51  })
52  .catch((err) => {
53    res.status(500).send({ message: err.message });
54  });
55 });

```

Figura 1.2: Método para registrar un nuevo usuario
Fuente: Elaborado por el investigador

Verificación de token válido

Para el consumo de los servicios expuestos en el backend es necesario verificar el token de inicio de sesión, como se puede observar en el código fuente se utiliza instancia una variable llamada token, donde se le asigna el valor enviado como header y que tiene el nombre de x-access-token, y a través de la librería de jwt con su método verify se valida dicho token y se envía un mensaje de error de ser el caso.

```

5  verifyToken = (req, res, next) => {
6    let token = req.headers["x-access-token"];
7    if (!token) {
8      return res.status(403).send({
9        message: "No ingreso un token válido!"
10     });
11   }
12   jwt.verify(token, config.secret, (err, decoded) => {
13     if (err) {
14       return res.status(401).send({
15         message: "No autorizado!"
16       });
17     }
18     console.log(decoded)
19     req.userCard = decoded.card;
20     next();
21   });
22 };

```

Figura 1.3: Método para verificar token de inicio de sesión
Fuente: Elaborado por el investigador

Listar usuarios

Método para obtener todos los usuarios registrados en el sistema.

```

71  exports.findAll = (req, res) => {
72    User.findAll()
73      .then(data => {
74        res.send(data);
75      })
76      .catch(err => {
77        res.status(500).send({
78          message:
79            err.message ||
80            "Algún error ocurrió mientras se estaba consultando los usuarios.",
81        });
82      });
83 };

```

Figura 1.4: Método para obtener los usuarios del sistema
Fuente: Elaborado por el investigador

Actualizar usuario

Este método actualiza un usuario validando que tanto el número de cédula como el correo electrónico no se encuentren registrados.

```

114 exports.update = (req, res) => {
115   // Validar petición
116   if (
117     !req.body.card ||
118     !req.body.names ||
119     !req.body.lastNames ||
120     !req.body.email ||
121     !req.body.password
122   ) {
123     res.status(400).send({
124       message: "No puede enviar parámetros vacíos!",
125     });
126     return;
127   }
128
129   const id = req.params.id;
130   const user = {
131     card: req.body.card,
132     names: req.body.names,
133     lastNames: req.body.lastNames,
134     email: req.body.email,
135     password: bcrypt.hashSync(req.body.password, 8),
136     active: 0,
137   };
138   User.update(user, {
139     where: { id: id },
140   })
141     .then((num) => {
142       if (num == 1) {
143         res.send({
144           message: "El usuario fue actualizado correctamente.",
145         });
146       } else {
147         res.send({
148           message: `No se puede actualizar el usuario con la id=${id}.
149             Pueda que el usuario no funcione o el req.body este vacio!`,
150         });
151       }
152     })
153     .catch((err) => {
154       res.status(500).send({
155         message: "Error al actualizar el usuario con la id=" + id,
156       });
157     });
158   });

```

Figura 1.5: Método para actualizar un usuario del sistema
Fuente: Elaborado por el investigador

Eliminar usuario

Este método elimina a un usuario de la base de datos.

```

165 exports.delete = (req, res) => {
166   const id = req.params.id;
167   User.destroy({
168     where: { id: id },
169   })
170   .then((num) => {
171     if (num == 1) {
172       res.send({
173         message: "El usuario fue eliminado correctamente!",
174       });
175     } else {
176       res.send({
177         message: `No se puede eliminar el usuario con la id=${id}.
178         Puede que el usuario no funcione!`,
179       });
180     }
181   })
182   .catch((err) => {
183     res.status(500).send({
184       message: "No se puede eliminar el usuario con la id=" + id,
185     });
186   });
187 });

```

Figura 1.6: Método para eliminar un usuario del sistema
Fuente: Elaborado por el investigador

Listar productos

Método para listar los productos que contenga la base de datos.

```

10 exports.findAllProducts = (req, res) => {
11   Product.findAll()
12   .then((data) => {
13     res.send(data);
14   })
15   .catch((err) => {
16     res.status(500).send({
17       message:
18         err.message ||
19         "Algún error ocurrió mientras se estaba consultando los productos.",
20     });
21   });
22 });

```

Figura 1.7: Método para obtener los usuarios del sistema
Fuente: Elaborado por el investigador

Buscar producto

Método para buscar un producto a través de su id, nombre, descripción y stock.

```

29 exports.searchProductsInformation = (req, res) => {
30   const productInfo = req.params.productInfo;
31   Product.findAll({
32     where: {
33       [Op.or]: [
34         {
35           id: {
36             [Op.like]: "%" + productInfo + "%",
37           },
38         },
39         {
40           name: {
41             [Op.like]: "%" + productInfo + "%",
42           },
43         },
44         {
45           description: {
46             [Op.like]: "%" + productInfo + "%",
47           },
48         },
49         {
50           stock: {
51             [Op.like]: "%" + productInfo + "%",
52           },
53         },
54       ],
55     },
56   })
57   .then((data) => {
58     res.send(data);
59   })
60   .catch((err) => {
61     res.status(500).send({
62       message:
63         err.message ||
64         "Algún error ocurrió mientras se estaba consultando los productos.",
65     });
66   });
67 };

```

Figura 1.8: Método para buscar un producto
Fuente: Elaborado por el investigador

Crear sucursal

Método que crea una sucursal.

```

16 exports.createStore = (req, res) => {
17   if (
18     !req.body.name ||
19     !req.body.phone ||
20     !req.body.reference ||
21     !req.body.lat ||
22     !req.body.lng
23   ) {
24     res.status(400).send({
25       message: "No puede enviar parámetros vacíos!",
26     });
27     return;
28   }
29   const store = {
30     name: req.body.name,
31     phone: req.body.phone,
32     reference: req.body.reference,
33     lat: req.body.lat,
34     lng: req.body.lng,
35   };
36   Store.create(store)
37     .then((data) => {
38       res.send(data);
39     })
40     .catch((err) => {
41       res.status(500).send({
42         message:
43           err.message || "Algún error ocurrió mientras se creaba la sucursal.",
44       });
45     });
46 };

```

Figura 1.9: Método para crear una sucursal

Fuente: Elaborado por el investigador

Asignar sucursal a producto

Método para asignar una sucursal a un determinado producto.


```

66 exports.createProductStore = (req, res) => {
67
68   const id = req.body.id;
69
70   Product.findById(id)
71     .then((product) => {
72       if (product) {
73         if (req.body.stores) {
74           Store.findAll({
75             where: {
76               id: {
77                 [Op.or]: req.body.stores,
78               },
79             },
80           }).then((stores) => {
81
82             product.addStores(stores).then(() => {
83               res.send({
84                 message:
85                   "El producto y sus sucursales fueron registradas satisfactoriamente!",
86               });
87             });
88           });
89         } else {
90           res.status(404).send({
91             message:
92               "No puedes enviar un producto sin el punto de distribución que posea stock de dicho producto",
93           });
94         }
95       } else {
96         res.status(404).send({
97           message: `No se puede encontrar el producto con id=${id}.`,
98         });
99       }
100     })
101     .catch((err) => {
102       res.status(500).send({
103         message: "Error al recuperar el producto con la id=" + id,
104       });
105     });
106   });

```

Figura 1.10: Método para asignar una sucursal a un producto
Fuente: Elaborado por el investigador

Registrar ubicaciones del recorrido de un proveedor

Método que crea un arreglo el cual contiene las ubicaciones que un proveedor ha ingresado y las inserta en la base de datos utilizando la función bulkCreate propia de sequelize.

```

232 exports.insertProductUbications = (req, res) => {
233   // Validar petición
234   if (!req.body.ubications || !req.body.userProductId) {
235     res.status(400).send({
236       message: "No puede enviar parámetros vacíos!",
237     });
238     return;
239   }
240   let ubications = [];
241   for (let i = 0; i < req.body.ubications.length; i++) {
242     ubications.push({
243       lat: req.body.ubications[i].lat,
244       lng: req.body.ubications[i].lng,
245       usersProductId: req.body.userProductId,
246       active: 0,
247       observation: "Ninguna",
248     });
249   }
250   Ubication.bulkCreate(ubications)
251     .then(() => {
252       res.send({
253         message: "Las ubicaciones se insertaron correctamente",
254       });
255     })
256     .catch((err) => {
257       res.status(500).send({ message: err.message });
258     });
259 };

```

Figura 1.11: Método para insertar ubicaciones del recorrido de un proveedor
Fuente: Elaborado por el investigador

Mostrar ubicaciones del recorrido de un proveedor

Método que obtiene todas las ubicaciones validadas de un producto por su proveedor y controlador, estas ubicaciones se obtienen utilizando el método `findAll` de `sequelize` y validando que las mismas se encuentren activas.

```

88 exports.getAllValidateUbicationsByUserCardProductId = (req, res) => {
89   const productId = req.params.productId;
90   const userCard = req.params.userCard;
91
92   User_Product.findOne({
93     where: {
94       userCard: userCard,
95       productId: productId,
96     },
97   })
98   .then((data) => {
99     if (data) {
100       Ubication.findAll({
101         where: { usersProductId: data.id, active: 1 },
102       })
103       .then((data) => {
104         res.send(data);
105       })
106       .catch((err) => {
107         res.status(500).send({
108           message:
109             err.message ||
110             "Algún error ocurrió mientras se estaba consultando las ubicaciones.",
111         });
112       });
113     } else {
114       res.status(500).send({
115         message: "No se encontro ningún dato registrado",
116       });
117     }
118   })
119   .catch((err) => {
120     res.status(500).send({
121       message:
122         err.message ||
123         "Algún error ocurrió mientras se estaba consultando los controladores.",
124     });
125   });
126 });

```

Figura 1.12: Método para obtener las ubicaciones validadas que ha recorrido un proveedor

Fuente: Elaborado por el investigador

Enviar correo electrónico

Para enviar correos electrónicos se utilizó la librería de nodemailer, a través de sus configuraciones se estableció el servicio de gmail para el envío del correo.

```

12 let transporter = nodemailer.createTransport({
13   service: "Gmail",
14   auth: {
15     user: process.env.MAILTOSENDMESSAGE,
16     pass: process.env.SITEPASSWORD,
17   },
18 });

```

Figura 1.13: Instanciación de nodemailer

Fuente: Elaborado por el investigador

```

257 function sendMailMessage(acDeAccount, userEmail, userNames) {
258   let url = process.env.URLSITE;
259   if (acDeAccount) {
260     let activatedAccountMessage =
261       "Estimado " + userNames + " tu cuenta ha sido activada, por favor vuelve a iniciar sesión en Woocommerce!";
262     let mailData = {
263       from: process.env.MAILTOSENDMESSAGE, // sender address
264       to: userEmail, // list of receivers
265       subject: "Tu cuenta en Woomap ha sido activada",
266       text: activatedAccountMessage,
267       html: `<b>Estimado ` + userNames + ` . </b> <br><br><br> Tu cuenta ha sido activada, por favor vuelve a iniciar sesión en <a href="" + url + "">Woomap</a>.`;
268     };
269     transporter.sendMail(mailData, function (err, info) {
270       if (err) {
271         console.log(err);
272       } else {
273         console.log(info);
274       }
275     });
276   } else {
277     let deactivatedAccountMessage = "Estimado " + userNames + " tu cuenta ha sido desactivada, ya no podrás iniciar sesión en Woomap!";
278     let mailData = {
279       from: process.env.MAILTOSENDMESSAGE, // sender address
280       to: userEmail, // list of receivers
281       subject: "Tu cuenta en Woomap ha sido desactivada",
282       text: deactivatedAccountMessage,
283       html: `<b>Estimado ` + userNames + ` . </b> <br><br><br> Tu cuenta ha sido desactivada, ya no podrás iniciar sesión en <a href="" + url + "">Woomap</a>.`;
284     };
285     transporter.sendMail(mailData, function (err, info) {
286       if (err) {
287         console.log(err);
288       } else {
289         console.log(info);
290       }
291     });
292   }
293 }

```

Figura 1.14: Método para enviar correo de gmail
Fuente: Elaborado por el investigador

1.2. Métodos del frontend

Token de inicio de sesión

Métodos para guardar el token de inicio de sesión que es enviado desde el backend y método para obtener dicho token.

```

public saveToken(token: string): void {
  window.sessionStorage.removeItem(TOKEN_KEY);
  window.sessionStorage.setItem(TOKEN_KEY, token);
}
public getToken(): string | null {
  return window.sessionStorage.getItem(TOKEN_KEY);
}

```

Figura 1.15: Métodos para guardar y obtener el token de inicio de sesión
Fuente: Elaborado por el investigador

Información del usuario

Métodos para guardar el usuario que inicia sesión y método que obtiene el usuario guardado en el session storage.

```

public saveUser(user: any): void {
  window.sessionStorage.removeItem(USER_KEY);
  window.sessionStorage.setItem(USER_KEY, JSON.stringify(user));
}
public getUser(): any {
  const user = window.sessionStorage.getItem(USER_KEY);
  if (user) {
    return JSON.parse(user);
  }
  return {};
}

```

Figura 1.16: Métodos para guardar y obtener el usuario logueado

Fuente: Elaborado por el investigador

Guards

Son interfaces que protegen el acceso a rutas, permitiendo o revocando el acceso al usuario, el siguiente método sirve para proteger el acceso a las rutas que necesitan que el usuario inicie sesión.

```

canActivate(
  route: ActivatedRouteSnapshot,
  state: RouterStateSnapshot): Observable<boolean | UrlTree> | Promise<boolean | UrlTree> | boolean | UrlTree {
  if (!this.isLoggedIn) {
    return this.router.navigate(['/login']).then(() => false);
  }
  return true;
}

```

Figura 1.17: Guard que redirecciona a la ventana de login

Fuente: Elaborado por el investigador

El siguiente método comprueba que el usuario este logueado y pueda solo acceder a recursos que le permita utilizar su rol.

```

canActivate(
  route: ActivatedRouteSnapshot,
  state: RouterStateSnapshot): Observable<boolean | UrlTree> | Promise<boolean | UrlTree> | boolean | UrlTree {
  if (this.isLoggedIn) {
    return this.router.navigate(['/home']).then(() => false);
  }
  return true;
}

```

Figura 1.18: Guard que redirecciona a la ventana de home

Fuente: Elaborado por el investigador

Métodos para iniciar sesión y registrarse

Los siguientes métodos inician sesión y registran un nuevo usuario, a través del consumo de una api externa.

```

17 login(email: string, password: string): Observable<any> {
18     return this.http.post(AUTH_API + 'signin', {
19         email,
20         password
21     }, httpOptions);
22 }
23
24 register(card: string, names:string, lastNames:string, email:string, phone:string, password: string):
25 Observable<any> {
26     return this.http.post(AUTH_API + 'signup', {
27         card,
28         names,
29         lastNames,
30         email,
31         phone,
32         password
33     }, httpOptions);
34 }
35 }

```

Figura 1.19: Métodos de inicio de sesión y registro de un usuario nuevo
Fuente: Elaborado por el investigador

Mensajes emergentes

Para mostrar mensajes emergentes al usuario se utilizó sweetalert que es una librería que nos permite customizar mensajes de error, aprobación, advertencia, indicación, entre otros.

```

116 openSuccessUpdate(): void {
117     this.dialogRef.close();
118     this.loadingState = false;
119     Swal.fire({
120         icon: 'success',
121         title: 'Ubicación actualizada correctamente',
122         showConfirmButton: false,
123         timer: 2000
124     });
125 }
126 }
127
128 openUpdateError(mensaje: string) : void {
129     Swal.fire({
130         icon: 'error',
131         title: 'No se pudo actualizar la ubicación',
132         text: mensaje,
133         confirmButtonColor: "#652D09",
134         confirmButtonText: "Aceptar",
135     });
136 }

```

Figura 1.20: Métodos para abrir un mensaje emergente de aprobación y error
Fuente: Elaborado por el investigador

Mapa leaflet

El uso de mapas opensource en el presente sistema fue fundamental es por eso que se utilizó leaflet, que es una librería que nos ayuda con la implementación de mapas totalmente customizables, además existen librerías externas como es el caso de antpath que nos permite dibujar un camino a través de una serie de puntos de locación.

```

76     <div id="storesStockMap"></div>
77     <div *ngIf="havePoints">
78         <div class="row justify-content-center pt-5" id="errorMessage">
79             <div class="col-sm-8">
80                 <mat-card fxFlexFill style="background-color: #FFC3B9;padding:5px; color: #000000;">
81                     <div class="align-items-center justify-content-center text-center pt-2">
82                         <p>No hay puntos de distribución registrados en el sistema</p>
83                     </div>
84                 </mat-card>
85             </div>
86         </div>
87     </div>
88 </div>

```

Figura 1.21: Declaración de contenedor para insertar mapa
Fuente: Elaborado por el investigador

```

55 initMap(lat: string, lng: string, active: boolean): void {
56
57     this.mapApproveUbications = L.map("mapApproveUbications").setView(this.initialPos, 15);
58
59     const tiles = L.tileLayer(
60         'https://{s}.tile.openstreetmap.org/{z}/{x}/{y}.png',
61         {
62             opacity: 0.7,
63             maxZoom: 20,
64             attribution:
65                 '&copy; <a href="https://www.openstreetmap.org/copyright">OpenStreetMap</a> contributors',
66         }
67     );
68
69     tiles.addTo(this.mapApproveUbications);
70     let marker = {};
71
72     // Establecemos el zoom minimo y maximo de nuestro mapa
73     this.mapApproveUbications.options.minZoom = 13;
74     this.mapApproveUbications.options.maxZoom = 16.5;
75
76     //Agregar marcador
77     if(active){
78         marker = L.marker([lat, lng], {
79             icon: this.passedIcon,
80         }).addTo(this.mapApproveUbications);
81     }else{
82         marker = L.marker([lat, lng], {
83             icon: this.storeIcon,
84         }).addTo(this.mapApproveUbications);
85     }
86
87     let zoomlvl = 16.5;
88     this.mapApproveUbications.flyTo([lat, lng], zoomlvl);
89
90     this.mapApproveUbications.on('click', (e) => {
91         this.mapApproveUbications.flyTo([lat, lng], zoomlvl);
92     });
93 }

```

Figura 1.22: Métodos para instanciar un mapa de leaflet
Fuente: Elaborado por el investigador

```

213     let path = antPath(latlngs, {
214         delay: 600,
215         dashArray: [10, 20],
216         weight: 5,
217         color: '#FF0000',
218         pulseColor: '#D50000',
219         paused: false,
220         reverse: false,
221         hardwareAccelerated: true,
222     });
223
224     this.ubicationsMap.addLayer(path);
225     this.ubicationsMap.fitBounds(path.getBounds());

```

Figura 1.23: Instanciación de librería antpath y uso en un mapa leaflet instanciado

Fuente: Elaborado por el investigador

Popup customizado en leaflet

Al momento de presionar sobre una ubicación establecida en un mapa de leaflet es posible mostrar información importante de dicho punto, para esto se utiliza el método propio de leaflet llamado bindPopup.

```

(data) => {
  let allStores = data;
  let sMap = this.storesStockMap;
  for (let i = 0; i < allStores.length; i++) {
    let marker = L.marker([allStores[i].lat, allStores[i].lng], {
      icon: this.storeIcon,
    })
    .addTo(this.storesStockMap)
    .on('click', function (e) {
      let zoomlvl = 16.5;
      sMap.flyTo(e.latlng, zoomlvl);
    });
    let messagePopUp =
      '<center><h3><b>' +
      allStores[i].name +
      '</b></h3></center><p><b>Teléfono: </b>' +
      allStores[i].phone +
      '</p><p><b>Referencia: </b>' +
      allStores[i].reference;
    marker.bindPopup(messagePopUp, {
      closeButton: false,
    });
  }
  if(allStores.length > 0){
    this.havePoints = false;
  }else{
    this.havePoints = true;
  }
},
(err) => {
  console.log(JSON.parse(err.error).message);
}

```

Figura 1.24: Método bindPopup

Fuente: Elaborado por el investigador

Ingresar recorrido de un proveedor

Para insertar el recorrido de un proveedor se consume el método creado en el backend, el cual solicita como parámetros un array de ubicaciones y la id del proveedor que se encuentre logueado.


```

400 insertUbications(): void {
401   if (this.selectedMarker) {
402     this.userProductsService
403       .getUsersProductByUserCardProductId(this.id, this.user.card)
404       .subscribe(
405         (data) => {
406           this.ubicacionService
407             .createProductUbications(this.markPositions, data.id)
408             .subscribe(
409               (data) => {
410                 this.openSuccessCreate();
411               },
412               (err) => {
413                 console.log(err.error.message);
414                 this.openErrorMessage(
415                   'No se pudieron ingresar las ubicaciones, por favor prueba ingresándolas nuevamente'
416                 );
417               }
418             );
419         },
420         (err) => {
421           console.log(err.error.message);
422           this.openErrorMessage(
423             'No se pudo obtener los datos del registro actual'
424           );
425         }
426       );
427   } else {
428     this.openErrorMessage('Debes seleccionar mínimo un punto de ubicación!');
429   }
430 }

```

Figura 1.25: Métodos para insertar una o varias ubicaciones
Fuente: Elaborado por el investigador

Validar puntos del recorrido de un proveedor

Para que un controlador pueda validar los puntos que haya registrado un proveedor es necesario que las ubicaciones se encuentren representadas dentro de un mapa leaflet, para esto el siguiente método a través del consumo de una operación del servicio de ubicaciones del backend obtiene todos los puntos que se hayan registrado de un producto en específico y de esta manera recorriendo todas las ubicaciones con un ciclo for se insertaron dentro del mapa de la interfaz.

```

117 getUbications(
118   productId: string,
119   providerCard: string,
120   controllerCard: string
121 ): void {
122   // Obtenemos todas las ubicaciones existentes
123   this.ubicacionService
124     .getAllUbicationsByUserCardProductIdControllerCard(
125     productId,
126     providerCard,
127     controllerCard
128   )
129   .subscribe(
130     (data) => {
131     let allUbications = data;
132     let sMap = this.ubicationsProductMap;
133     for (let i = 0; i < allUbications.length; i++) {
134     if (allUbications[i].active) {
135     let marker = L.marker(
136     [allUbications[i].lat, allUbications[i].lng],
137     {
138     icon: this.passedIcon,
139     }
140     )
141     .addTo(this.ubicationsProductMap)
142     .on('click', function (e) {
143     let zoomlvl = 16.5;
144     sMap.flyTo(e.latlng, zoomlvl);
145     });
146     let messagePopUp =
147     '<p><b>Observación: </b> ' +
148     allUbications[i].observation +
149     '</p><center><button type="button" class="manageButton btn btn-primary btn-sm" '+
150     'style="background-color: #382B6F;">Administrar ubicación</button></center>';
151     marker
152     .bindPopup(messagePopUp, {
153     closeButton: false,
154     })
155     .on('popupopen', (a) => {
156     let popUp = a.target.getPopup();
157     // Boton editar
158     popUp
159     .getElement()
160     .querySelector('.manageButton')
161     .addEventListener('click', (e) => {
162     this.manageUbication(allUbications[i].id);
163     });
164     });
165     } else {
166     let marker = L.marker(
167     [allUbications[i].lat, allUbications[i].lng],
168     {
169     icon: this.storeIcon,
170     }
171     )
172     .addTo(this.ubicationsProductMap)
173     .on('click', function (e) {
174     let zoomlvl = 16.5;
175     sMap.flyTo(e.latlng, zoomlvl);
176     });
177     let messagePopUp =
178     '<p><b>Observación: </b> ' +
179     allUbications[i].observation +
180     '</p><center><button type="button" class="manageButton btn btn-primary btn-sm" '+
181     'style="background-color: #382B6F;">Administrar ubicación</button></center>';
182     marker
183     .bindPopup(messagePopUp, {
184     closeButton: false,
185     })
186     .on('popupopen', (a) => {
187     let popUp = a.target.getPopup();
188     // Boton editar
189     popUp
190     .getElement()
191     .querySelector('.manageButton')
192     .addEventListener('click', (e) => {
193     this.manageUbication(allUbications[i].id);
194     });
195     });
196     }
197   }
198   if(allUbications.length > 0){
199     this.havePoints = false;
200   }else{
201     this.havePoints = true;
202   }
203 },
204 (err) => {
205   console.log(JSON.parse(err.error).message);
206 }
207 );
208 }

```

Figura 1.26: Método que obtiene ubicaciones registradas
Fuente: Elaborado por el investigador

Para administrar los puntos registrados se utiliza el siguiente método que a través de un dialogRef abre una ventana flotante, en donde se puede aprobar los puntos registrados de parte del controlador.

```
208 manageUbication(ubicacionId: string): void {
209   this.ubicacionService.getUbicationById(ubicacionId).subscribe(
210     (data) => {
211       const dialogRef = this.dialog.open(ApproveUbicationComponent, {
212         width: '1000px',
213         data: {
214           id: ubicacionId,
215           lat: data.lat,
216           lng: data.lng,
217           active: data.active,
218           observation: data.observation,
219         },
220       });
221
222       dialogRef.afterClosed().subscribe((result) => {
223         let selectedProvider = this.providersData.find(
224           (x) => x.card == this.value
225         );
226         this.ubicacionsProductMap.remove();
227         this.initUbicationsMap();
228         this.getUbications(this.id, selectedProvider!.card, this.user.card);
229         this.selectedItem = undefined;
230       });
231     },
232     (err) => {
233       console.log(JSON.parse(err.error).message);
234       this.openErrorMessage(
235         'No se pudo consultar la información de la ubicación'
236       );
237     }
238   );
239 }
```

Figura 1.27: Método invocado para abrir ventana flotante de edición
Fuente: Elaborado por el investigador

```
100 updateUbicationStatus(): void {
101   let observation = this.manageUbicationForm.controls['commentController'].value;
102   this.loadingState = true;
103   //Actualizamos la información del usuario
104   this.ubicacionService.updateAUbicationById(this.data.id, this.activeDesactivePoint, observation).subscribe(
105     (data) => {
106       this.openSuccessUpdate();
107       this.loadingState = false;
108     },
109     (err) => {
110       this.openUpdateError(err.error.message);
111       this.loadingState = false;
112     }
113   );
114 }
```

Figura 1.28: Método que actualiza el estado de una ubicación registrada
Fuente: Elaborado por el investigador

Componentes de angular material

Para el diseño de la interfaz se utilizó angular material como proveedor de componentes.

```

24  const MaterialComponents = [
25    MatButtonModule,
26    MatFormFieldModule,
27    MatCardModule,
28    MatIconModule,
29    ReactiveFormsModule,
30    MatInputModule,
31    MatSnackBarModule,
32    MatSidenavModule,
33    MatToolbarModule,
34    MatListModule,
35    MatPaginatorModule,
36    MatTableModule,
37    MatProgressSpinnerModule,
38    MatDialogModule,
39    MatMenuModule,
40    MatSelectModule
41  ];
42
43  @NgModule({
44    imports: [MaterialComponents],
45    exports: [MaterialComponents],
46  })
47  export class MaterialModule {}

```

Figura 1.29: Declaración componentes angular material
Fuente: Elaborado por el investigador

Comprobación de sesión del usuario

Para consumir los servicios en el frontend es necesario comprobar que se encuentre una sesión activa, para esto se llama al servicio de token storage, el cual contiene el método para obtener el token generado al iniciar la sesión de un usuario

```

64  ngOnInit(): void {
65    this.loadingState = true;
66
67    // Obtenemos el usuario que inicio sesion
68    this.isLoggedIn = !!this.tokenStorageService.getToken();
69    if (this.isLoggedIn) {
70      const user = this.tokenStorageService.getUser();
71      this.currentUser = this.tokenStorageService.getUser();
72      this.loadDataSource();
73      this.validateUserData(user.card);
74
75      // Obtenemos la cantidad de usuarios, administradores y proveedores
76
77      this.getCountAllUsers();
78      this.getCountAllAdmins();
79      this.getCountAllDeliveryMans();
80      this.getCountAllControllers();
81    }
82  }

```

Figura 1.30: Comprobación de inicio de sesión de un usuario
Fuente: Elaborado por el investigador

1.3. Repositorios

En los siguientes repositorios de github se tiene el código completo de la aplicación realizada en caso de que se desee obtener una parte del código y de esta manera replicarlo.

Repositorio github del frontend: <https://github.com/Sebasmn/proyecto-seguimiento-productos.git>

Repositorio github del backend: <https://github.com/Sebasmn/rest-api.git>.

Anexo B

Cuestionario de la evaluación heurística utilizando los 10 principios de Jakob Nielsen para evaluar la usabilidad de la plataforma web

Cuestionario	Puntuación	Recomendaciones
1. Visibilidad del estado del sistema		
¿El usuario recibe la acción adecuada al interactuar con formularios, botones o enlaces?		
¿La interfaz de usuario y los iconos son coherentes en del sistema?		
2. Relación entre el sistema y el usuario		
¿Es fácil distinguir los elementos de la interfaz y las instrucciones de interacción?		
¿El contenido está redactado en un lenguaje comprensible para el usuario?		
3. Control y libertad para el usuario		
¿Es posible salir de todos los modos de la interfaz, como ventanas emergentes?		
¿El usuario puede interactuar con las páginas correspondientes a su rol?		
4. Consistencia y estándares		
¿Hay un diseño coherente para todos los elementos de la interfaz en la página?		
¿Los elementos de la interfaz cumplen con los estándares y usos esperados?		
5. Prevención de errores		
¿Se han establecido limitaciones visuales y útiles para evitar errores de los usuarios?		
¿Los mensajes de error están redactados con lenguaje gramaticalmente correcto para el usuario?		

6. Reconocer antes que recordar		
¿Existe una forma para que el usuario restablezca una nueva contraseña cuando sea necesario?		
¿Se usa el tamaño de letra, realce de fuente, subrayado, color, sombreado o tipografía para mostrar la relevancia de los ítems en la pantalla?		
7. Flexibilidad y eficiencia de uso		
¿Se le presenta al usuario información ambiental para acciones rápidas?		
¿Es fácil para los usuarios nuevos acostumbrarse al uso del sistema?		
8. Diálogos estéticos y diseño minimalista		
¿El diseño de la interfaz de usuario es simple y fácil de entender?		
¿Es evidente el significado de los íconos para el usuario y cuál es su función?		
9. Reconocimiento, diagnóstico y recuperación de errores		
¿Los usuarios reciben un mensaje de error descriptivo cuando proporcionan datos incorrectos?		
¿Los mensajes de error son amables y no culpan al usuario por los errores?		
10. Ayuda y documentación		
¿Los usuarios tienen acceso a información importante que necesiten para utilizar el sitio web?		
¿Existen canales de comunicación para que el usuario haga preguntas o resuelva dudas?		

Respuestas de ejecución del cuestionario a usuarios expertos:

Encuestado 1: Ing. Jefferzon Zambrano

Cuestionario	Puntuación	Recomendaciones
1. Visibilidad del estado del sistema		
¿El usuario recibe la acción adecuada al interactuar con formularios, botones o enlaces?	5	
¿La interfaz de usuario y los iconos son coherentes en del sistema?	4	
2. Relación entre el sistema y el usuario		
¿Es fácil distinguir los elementos de la interfaz y las instrucciones de interacción?	4	
¿El contenido está redactado en un lenguaje comprensible para el usuario?	4	
3. Control y libertad para el usuario		
¿Es posible salir de todos los modos de la interfaz, como ventanas emergentes?	4	
¿El usuario puede interactuar con las páginas correspondientes a su rol?	5	
4. Consistencia y estándares		
¿Hay un diseño coherente para todos los elementos de la interfaz en la página?	5	
¿Los elementos de la interfaz cumplen con los estándares y usos esperados?	3	
5. Prevención de errores		
¿Se han establecido limitaciones visuales y útiles para evitar errores de los usuarios?	4	
¿Los mensajes de error están redactados con lenguaje gramaticalmente correcto para el usuario?	4	

6. Reconocer antes que recordar		
¿Existe una forma para que el usuario restablezca una nueva contraseña cuando sea necesario?	4	
¿Se usa el tamaño de letra, realce de fuente, subrayado, color, sombreado o tipografía para mostrar la relevancia de los ítems en la pantalla?	3	
7. Flexibilidad y eficiencia de uso		
¿Se le presenta al usuario información ambiental para acciones rápidas?	2	
¿Es fácil para los usuarios nuevos acostumbrarse al uso del sistema?	4	
8. Diálogos estéticos y diseño minimalista		
¿El diseño de la interfaz de usuario es simple y fácil de entender?	3	
¿Es evidente el significado de los íconos para el usuario y cuál es su función?	3	
9. Reconocimiento, diagnóstico y recuperación de errores		
¿Los usuarios reciben un mensaje de error descriptivo cuando proporcionan datos incorrectos?	4	
¿Los mensajes de error son amables y no culpan al usuario por los errores?	5	
10. Ayuda y documentación		
¿Los usuarios tienen acceso a información importante que necesiten para utilizar el sitio web?	4	
¿Existen canales de comunicación para que el usuario haga preguntas o resuelva dudas?	5	

Encuestado 2: Ing, Alejandro Morales

Cuestionario	Puntuación	Recomendaciones
1. Visibilidad del estado del sistema		
¿El usuario recibe la acción adecuada al interactuar con formularios, botones o enlaces?	4.5	
¿La interfaz de usuario y los iconos son coherentes en del sistema?	4	
2. Relación entre el sistema y el usuario		
¿Es fácil distinguir los elementos de la interfaz y las instrucciones de interacción?	3.8	
¿El contenido está redactado en un lenguaje comprensible para el usuario?	4.7	
3. Control y libertad para el usuario		
¿Es posible salir de todos los modos de la interfaz, como ventanas emergentes?	4.2	
¿El usuario puede interactuar con las páginas correspondientes a su rol?	4.9	
4. Consistencia y estándares		
¿Hay un diseño coherente para todos los elementos de la interfaz en la página?	4.3	
¿Los elementos de la interfaz cumplen con los estándares y usos esperados?	4.5	
5. Prevención de errores		
¿Se han establecido limitaciones visuales y útiles para evitar errores de los usuarios?	4.7	
¿Los mensajes de error están redactados con lenguaje gramaticalmente correcto para el usuario?	4.4	

6. Reconocer antes que recordar		
¿Existe una forma para que el usuario restablezca una nueva contraseña cuando sea necesario?	4.3	Debería existir otra forma de recuperación de contraseña, puede ser a través del correo electrónico con un refresh token.
¿Se usa el tamaño de letra, realce de fuente, subrayado, color, sombreado o tipografía para mostrar la relevancia de los ítems en la pantalla?	4	
7. Flexibilidad y eficiencia de uso		
¿Se le presenta al usuario información ambiental para acciones rápidas?	4.3	
¿Es fácil para los usuarios nuevos acostumbrarse al uso del sistema?	4.6	
8. Diálogos estéticos y diseño minimalista		
¿El diseño de la interfaz de usuario es simple y fácil de entender?	4.5	
¿Es evidente el significado de los íconos para el usuario y cuál es su función?	3.7	
9. Reconocimiento, diagnóstico y recuperación de errores		
¿Los usuarios reciben un mensaje de error descriptivo cuando proporcionan datos incorrectos?	4.4	
¿Los mensajes de error son amables y no culpan al usuario por los errores?	4.6	
10. Ayuda y documentación		
¿Los usuarios tienen acceso a información importante que necesiten para utilizar el sitio web?	4.5	
¿Existen canales de comunicación para que el usuario haga preguntas o resuelva dudas?	3.8	

Cuestionario	Puntuación	Recomendaciones
1. Visibilidad del estado del sistema		
¿El usuario recibe la acción adecuada al interactuar con formularios, botones o enlaces?	4	
¿La interfaz de usuario y los iconos son coherentes en del sistema?	4	
2. Relación entre el sistema y el usuario		
¿Es fácil distinguir los elementos de la interfaz y las instrucciones de interacción?	3	
¿El contenido está redactado en un lenguaje comprensible para el usuario?	5	
3. Control y libertad para el usuario		
¿Es posible salir de todos los modos de la interfaz, como ventanas emergentes?	5	
¿El usuario puede interactuar con las páginas correspondientes a su rol?	4	
4. Consistencia y estándares		
¿Hay un diseño coherente para todos los elementos de la interfaz en la página?	4	
¿Los elementos de la interfaz cumplen con los estándares y usos esperados?	4	
5. Prevención de errores		
¿Se han establecido limitaciones visuales y útiles para evitar errores de los usuarios?	5	
¿Los mensajes de error están redactados con lenguaje gramaticalmente correcto para el usuario?	3	

6. Reconocer antes que recordar		
¿Existe una forma para que el usuario restablezca una nueva contraseña cuando sea necesario?	4	
¿Se usa el tamaño de letra, realce de fuente, subrayado, color, sombreado o tipografía para mostrar la relevancia de los ítems en la pantalla?	5	
7. Flexibilidad y eficiencia de uso		
¿Se le presenta al usuario información ambiental para acciones rápidas?	4	La información puede presentarse en el aplicativo web no solo en el e-commerce.
¿Es fácil para los usuarios nuevos acostumbrarse al uso del sistema?	4	
8. Diálogos estéticos y diseño minimalista		
¿El diseño de la interfaz de usuario es simple y fácil de entender?	3	
¿Es evidente el significado de los íconos para el usuario y cuál es su función?	4	
9. Reconocimiento, diagnóstico y recuperación de errores		
¿Los usuarios reciben un mensaje de error descriptivo cuando proporcionan datos incorrectos?	5	
¿Los mensajes de error son amables y no culpan al usuario por los errores?	4	
10. Ayuda y documentación		
¿Los usuarios tienen acceso a información importante que necesiten para utilizar el sitio web?	3	Subir documentación a la nube.
¿Existen canales de comunicación para que el usuario haga preguntas o resuelva dudas?	5	

Anexo C

Manual de usuario WooMap

El presente manual de usuario tiene como finalidad demostrar el uso del aplicativo desarrollado, de manera que el usuario que lo vaya a utilizar se familiarice rápidamente con el uso y funcionalidades que ofrece.

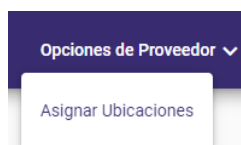
WooMap trabaja con tres roles:

- 1) Administrador
- 2) Proveedor
- 3) Controlador

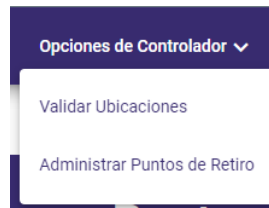
Administrador: Encargado de llevar la gestión de usuarios, administrar los puntos de distribución que la empresa tiene y asignar el personal que se encargará de llevar la información georeferenciada de cada producto.



Proveedor: Encargado de asignar ubicaciones de los productos que le haya asignado un administrador del sistema.

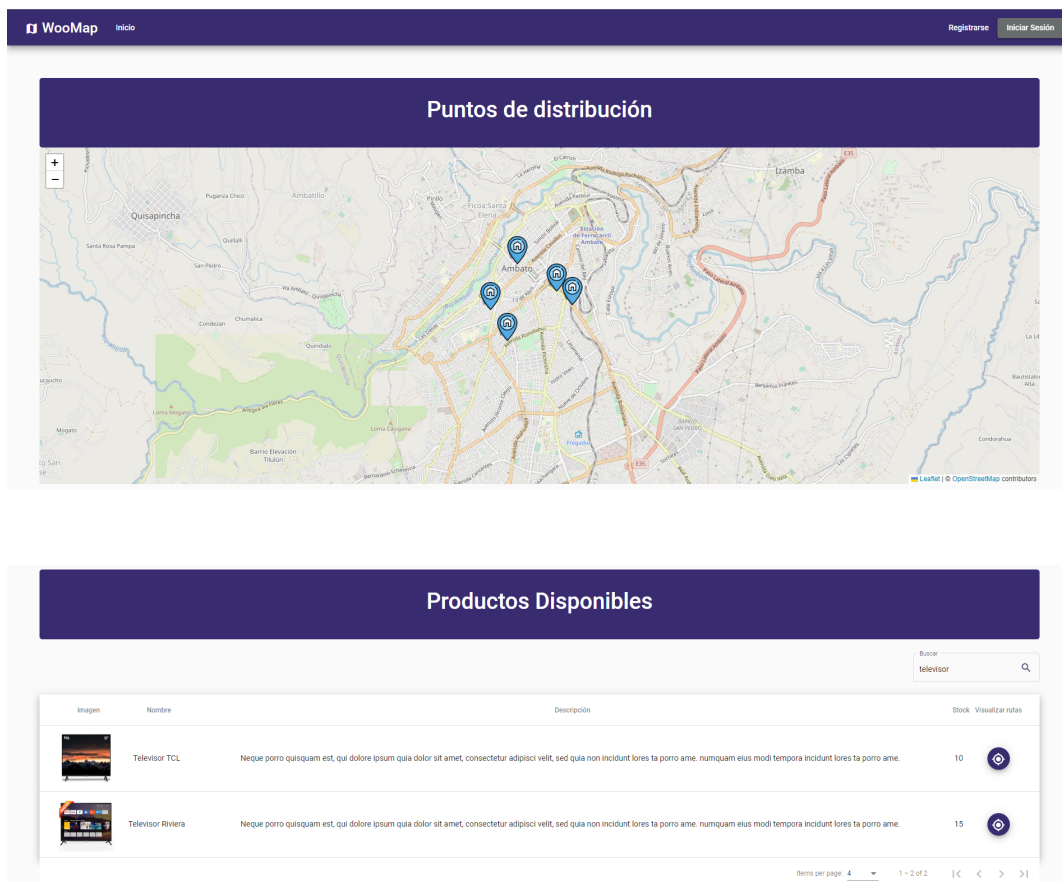


Controlador: Encargado de validar las ubicaciones de los productos asignadas por parte de los proveedores y administrar los puntos de retiro de cada producto (Lugares en donde tienen stock disponible del producto).



3.1. Navegación de usuario común

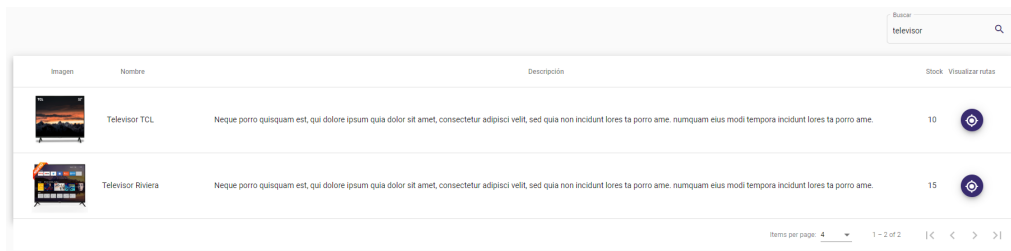
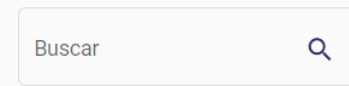
En caso de que un usuario no inicie sesión o se registre el sistema muestra por defecto la ventana de Inicio, que está conformada por dos secciones, los puntos de distribución y los productos disponibles del sistema.



En los puntos de distribución se tiene la potestad de dar clic sobre cualquiera de ellos para observar la información del nombre de la sucursal, el teléfono y alguna referencia de ubicación para llegar a ese destino.



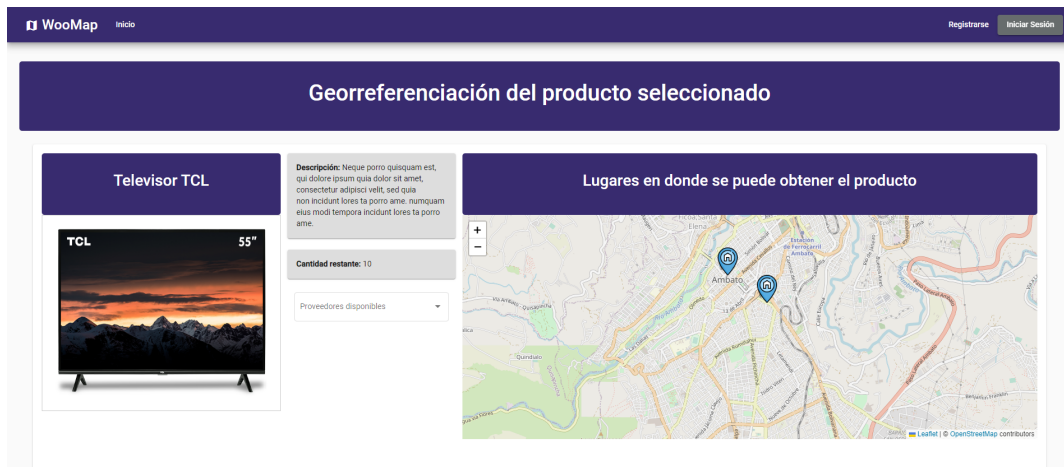
En los productos disponibles se tiene un buscador que permite encontrar el o los productos de una manera mas óptima y rápida, puede ser utilizado para buscar mediante el nombre, la descripción o el stock del producto.



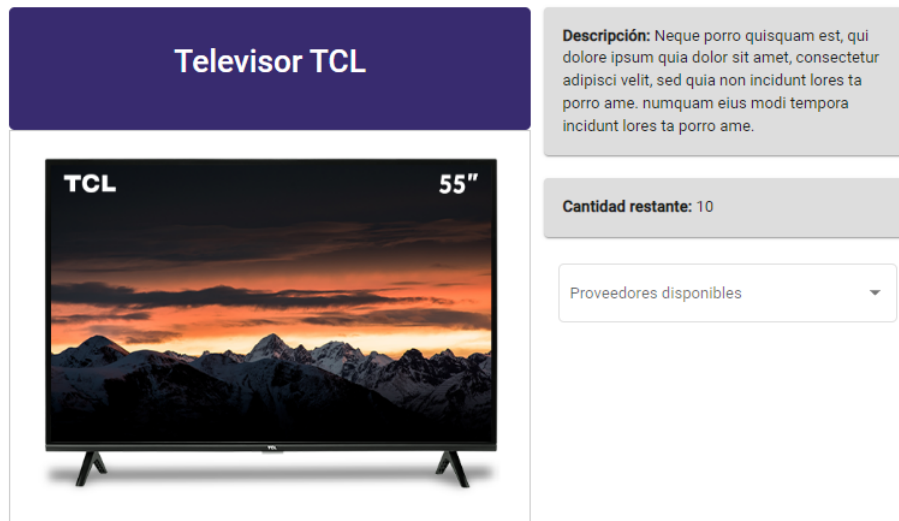
Cada producto tendrá un botón para visualizar la información georeferenciada del mismo.



Al presionar sobre el botón se abrirá la siguiente ventana.



En la ventana se muestra información del producto como su nombre, descripción y stock.



También se puede observar un mapa con los lugares que tienen disponibilidad de dicho producto.



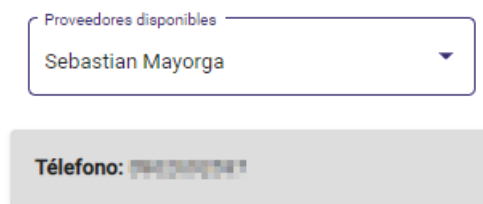
Si presiona en cualquiera de los puntos asignados en el mapa se despliega la información sobre la sucursal que tiene a disposición dicho producto.



Para observar la ruta realizada por un proveedor referente al producto seleccionado es necesario seleccionarlo del combobox.



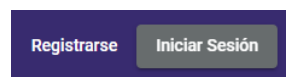
Ya seleccionado el proveedor que se desee conocer la información georeferenciada por donde haya pasado se agregara automáticamente a la interfaz el teléfono del proveedor.



También aparece un nuevo mapa, en el cual se visualiza la ruta que haya hecho el proveedor llevando el producto seleccionado, además se trazará el polígono de acuerdo a las ubicaciones que registró y que fueron validadas por un controlador.



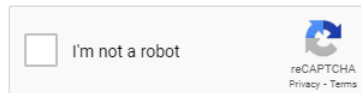
En el menú de la aplicación existen dos opciones extra, la de registro e inicio de sesión.



Si se da clic sobre Registrarse aparece la siguiente ventana, donde se debe ingresar los datos personales para la creación de la cuenta, además se validará que es una persona real resolviendo el algoritmo de reCAPTCHA.

Registro de usuario

<input type="text" value="Nombres *"/>	<input type="text" value="Apellidos *"/>
<input type="text" value="Número de cédula *"/> <small>10 dígitos 0/10</small>	<input type="text" value="Teléfono *"/>
<input type="password" value="Contraseña *"/>	<input type="password" value="Confirmar Contraseña *"/>
<input type="text" value="Correo electrónico *"/>	

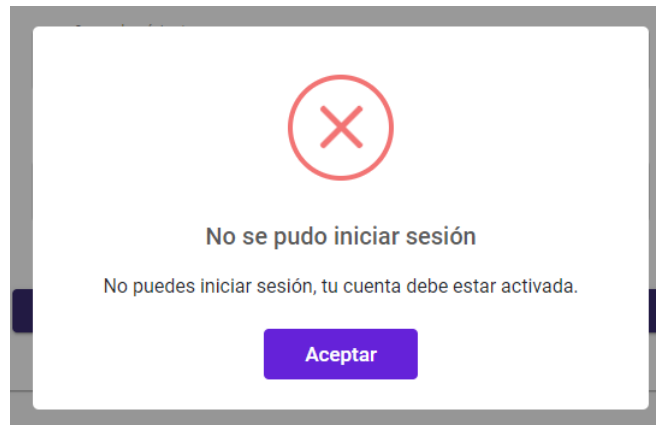


Si se da clic sobre Iniciar Sesión aparecerá la siguiente ventana de login.

Inicia sesión en tu cuenta

<input type="text" value="Correo electrónico *"/>
<input type="password" value="Contraseña *"/>
<input type="button" value="Iniciar sesión"/>

Para iniciar sesión en el sistema es importante saber que la cuenta debe estar activada caso contrario se le mostrará el siguiente mensaje de error.



3.2. Navegación de usuario logueado

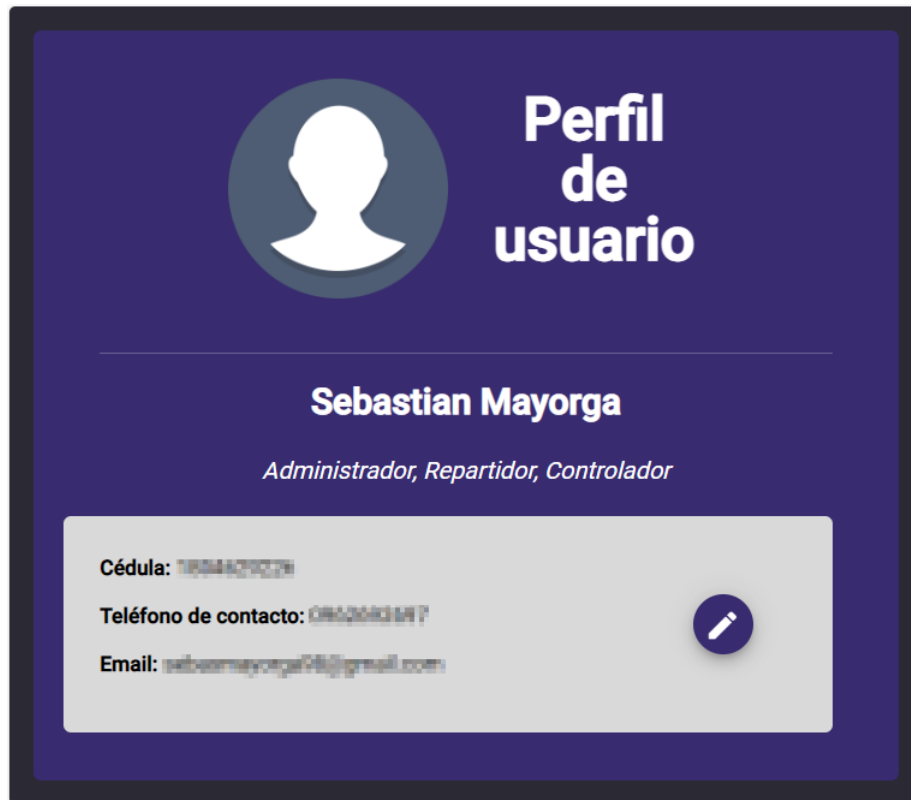
3.2.1. Aspectos generales

3.2.1.1. Perfil de usuario

Al iniciar sesión el usuario verá una bienvenida con su nombre completo a un costado del botón de cerrar sesión, mismo que está posicionado en la esquina superior derecha.

Bienvenido/a, Sebastian Mayorga

Al presionar sobre su nombre se despliega la siguiente ventana, donde se detallarán mas datos del usuario.



Para modificar la información del usuario que este con la sesión activa se debe presionar sobre el siguiente botón.



Posteriormente se abrirá la siguiente ventana emergente con todos sus datos cargados y con disponibilidad de edición a excepción de su cédula que es un dato que no podrá modificar.

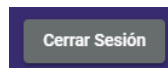


Editar usuario

<input type="text" value="Nombres *"/> <input type="text" value="Sebastian"/>	<input type="text" value="Apellidos *"/> <input type="text" value="Mayorga"/>
<input type="text" value="Número de cédula"/> <input type="text" value="XXXXXXXXXX"/> <small>10 dígitos</small>	<input type="text" value="Teléfono *"/> <input type="text" value="XXXXXXXXXX"/> <small>10/10</small>
<input type="password" value="Nueva contraseña"/> <small>Si deseas conservar la misma contraseña deja estos campos vacíos</small>	<input type="password" value="Confirmar Contraseña"/>
<input type="text" value="Correo electrónico *"/> <input type="text" value="sebastianmayorga@gmail.com"/>	

3.2.1.2. Cerrar sesión

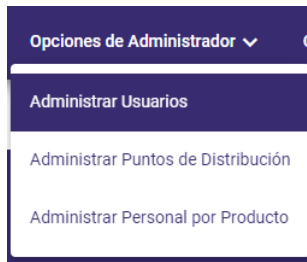
Para cerrar sesión en el sistema se tendrá un botón específico ubicado en la esquina superior derecha, solo se lo presiona y de este modo se cierra la sesión actual y automáticamente se redirige a la ventana de inicio.



3.2.2. Rol de administrador

3.2.2.1. Administrar usuarios

En las opciones de menú que tendrá el administrador seleccionaremos la primera que es Administrar usuarios.

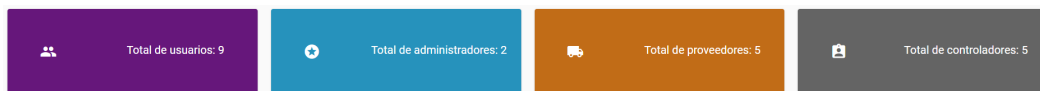


Al presionar sobre Administrar usuarios se nos desplegará la siguiente ventana.



Usuarios Registrados									
Cédula	Nombres	Apellidos	Correo electrónico	Teléfono	Administrador	Proveedor	Controlador	Cuenta activada / desactivada	Acciones
1736448888	Paulina	Torres	p@igmail.com	0993274533	✗	✗	✓	✗	
1803625056	William	Almagro	w@gmail.com	0978123612	✗	✓	✗	✓	
1804626236	Sebastian	Mayorga	sebsamayorga9@gmail.com	0962630817	✓	✗	✗	✓	
1884898155	Ernesto	Suarez	e@gmail.com	0817235637	✗	✗	✓	✓	

En paneles de diferentes colores se muestra el total de usuarios, administradores, proveedores y controladores registrados en el sistema.



Los usuarios que se vayan registrando en el sistema se muestran en la tabla de usuarios registrados.

Cédula	Nombres	Apellidos	Correo electrónico	Teléfono	Administrador	Proveedor	Controlador	Cuenta activada / desactivada	Acciones
1736448888	Paulina	Torres	p@gmail.com	0993274533	✗	✗	✓	✗	
1803625056	William	Almagro	w@gmail.com	0978123612	✗	✓	✗	✓	
1804626236	Sebastian	Mayorga	sebsamayorga9@gmail.com	0962630817	✓	✗	✗	✓	
1884898155	Ernesto	Suarez	e@gmail.com	0817235637	✗	✗	✓	✓	

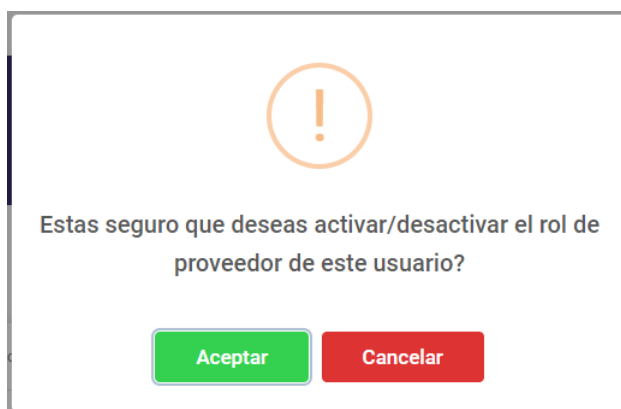
La tabla contiene información de cada usuario como la cédula, nombres, apellidos, correo electrónico y el número de teléfono.

Cédula	Nombres	Apellidos	Correo electrónico	Teléfono
1723644041	Paulina	Torres	p@p@gmail.com	0993317600
1800025004	William	Almagro	w@gmail.com	0978117411
1804027023	Sebastian	Mayorga	s@sebastianmayorga@gmail.com	0042002091
1804000101	Ernesto	Suarez	e@gmail.com	0977306011

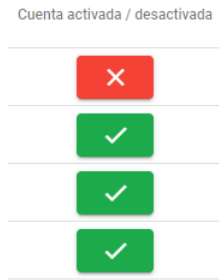
Los roles que tiene un usuario se representan con un botón verde y el botón rojo representa aquellos roles que no posee.

Administrador	Proveedor	Controlador
<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>

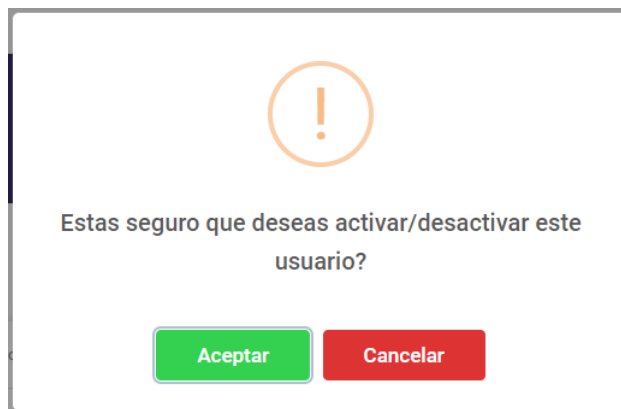
Para gestionar los roles de un usuario se da clic en el que se desee modificar, de esta manera aparecerá un mensaje de confirmación dependiendo del rol que se desee otorgar o revocar.



Se puede activar o desactivar la cuenta de cualquier usuario presionando el siguiente botón.




Aparecerá un mensaje de confirmación al activar o desactivar una cuenta.



Se tiene el botón de editar usuario en el caso de que sea necesario.



Al presionar sobre el botón de edición se abre la siguiente ventana emergente con los datos del usuario seleccionado para editar su información.



Editar usuario

Nombres* Paulina

Apellidos* Torres

Número de cédula
11723456789
10 dígitos 10/10

Teléfono*
0999999999

Nueva contraseña

Confirmar Contraseña

Si deseas conservar la misma contraseña deja estos campos vacíos

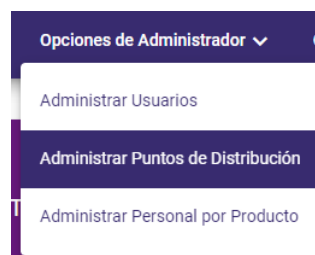
Correo electrónico*
p@pmail.com

Cancelar

Actualizar usuario

3.2.2.2. Administrar puntos de distribución

La segunda opción que puede utilizar el administrador es la de administrar puntos de distribución.



Presionando sobre dicha opción aparece la siguiente ventana.

WooMap Inicio Opciones de Administrador Bienvenido/a, Sebastian Mayorga Cerrar Sesión

Administración de puntos de distribución

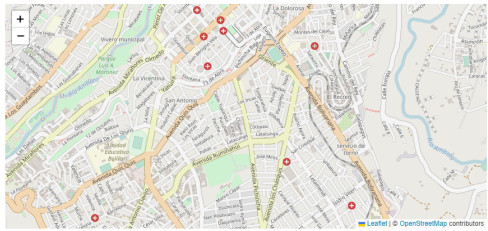
Crear nuevo punto de distribución

Nombre *

Teléfono *

Referencia *

Ubicación:



Crear punto de distribución

Puntos de distribución existentes



En esta ventana se puede administrar los puntos de distribución que se tengan registrados y también es posible registrar nuevos puntos si es necesario. Para crear un nuevo punto de distribución es necesario llenar la información del formulario el cual contiene el nombre, teléfono y una referencia a la ubicación, también se debe seleccionar la ubicación en el mapa.

Crear nuevo punto de distribución

Nombre *

🏠 Sucursal 6

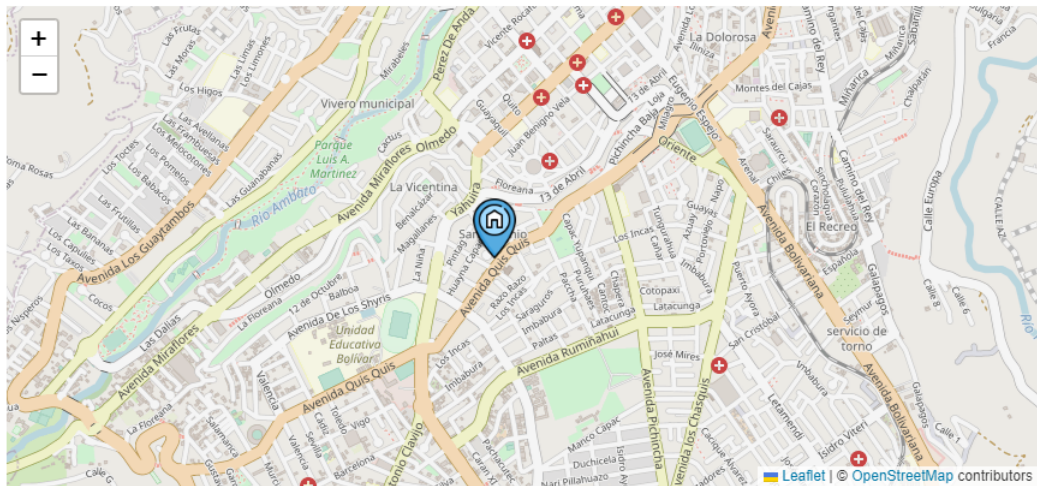
Teléfono *

☎️ 098172081507

Referencia *

📍 Alado de una casa verde

Ubicación:



➔ Crear punto de distribución

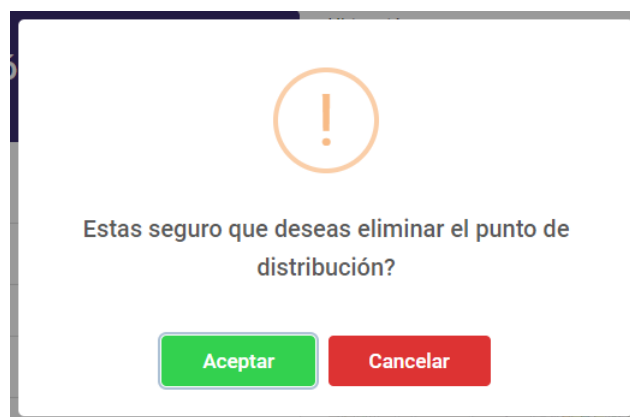
En el mapa de puntos de distribución existentes al presionar sobre una ubicación, se puede observar la información que tiene, además tiene las opciones de editarlo o eliminarlo.



Al editar un punto de distribución se abre una ventana emergente con sus datos y la ubicación que le hayan establecido, aquí se puede actualizar cualquier atributo.

A screenshot of a web form titled 'Editar punto de distribución'. The form has a dark blue header with the title in white. Below the header are three input fields: 'Nombre' with a house icon and the value 'Sucursal 4', 'Teléfono' with a phone icon and a redacted number, and 'Referencia' with a location pin icon and the value 'Frente a casa amarilla'. To the right of the form is a map showing the location of the distribution point. At the bottom of the form are two buttons: 'Cancelar' (white) and 'Actualizar' (dark blue). A red close button is visible in the top right corner of the form's container.

En el caso de que el administrador desee eliminar un punto de distribución aparece el siguiente mensaje de confirmación.



3.2.2.3. Administrar personal por producto

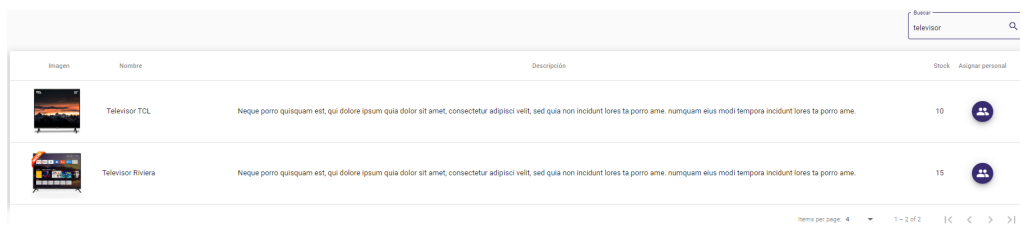
La última opción del menú que el administrador tiene es la de administrar personal por producto.



Al presionar sobre dicha opción se despliega la siguiente ventana, que contiene todos los productos disponibles.



En la interfaz existe un buscador que filtra la información de acuerdo al campo buscado, puede ser uno o varios productos que aparezcan cuando se realice la búsqueda.



Existe un botón específico por cada producto para asignar el personal.

Asignar personal



Al presionar el botón de un producto en específico se abre la siguiente ventana.

WooMap Inicio Opciones de Administrador Bienvenido/a, Sebastian Mayorga Cerrar Sesión

Gestor de personal por producto

Producto seleccionado: Celular Xiaomi

Descripción: Neque porro quisquam est, qui dolore ipsum quia dolor sit amet, consectetur adipisci velit, sed quia non incidunt lores ta porro ame. numquam eius modi tempora incidunt lores ta porro ame.

Seleccionar Proveedor * Seleccionar Controlador *

Asignar personal

Personal asignado

Buscar

Proveedor	Controlador	Acciones
Carlos Raza	Sebastian Mayorga	
Kevin Narango	Ernesto Suarez	

Items per page: 4 1 - 2 of 2

En la sección de gestor de personal por producto se tiene la información del producto seleccionado, también hay dos combobox para escoger a un proveedor y un controlador, cabe resaltar que se puede asignar varios proveedores y varios controladores al mismo producto.

Producto seleccionado: Celular Xiaomi

Descripción: Neque porro quisquam est, qui dolore ipsum quia dolor sit amet, consectetur adipisci velit, sed quia non incidunt lores ta porro ame. numquam eius modi tempora incidunt lores ta porro ame.

Seleccionar Proveedor * Seleccionar Controlador *

William Almagro Israel Lopez

Asignar personal

En la sección de personal asignado se tiene un buscador para encontrar de una manera mas rápida cualquier proveedor o controlador asignado.

Buscar

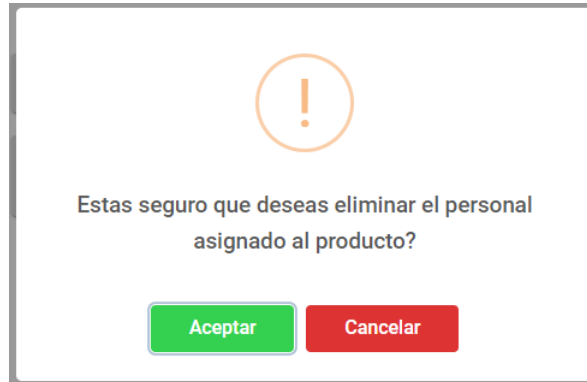
Sebastian

Para eliminar cualquier personal asignado al producto se lo realiza presionando el siguiente botón.

Acciones



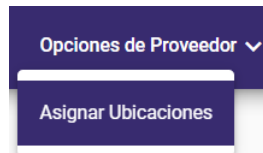
Para eliminar un proveedor y controlador asignado aparece el siguiente mensaje de confirmación.



3.2.3. Rol de proveedor

3.2.3.1. Asignar ubicaciones

Las opción que el menú de proveedor tiene es asignar ubicaciones.



Al presionar sobre dicha opción se despliega la siguiente ventana, donde estarán todos los productos asignados para el registro de ubicaciones.

WooMap Inicio Opciones de Proveedor Bienvenido a Sebastian Mayorga Centrar Sesión


Productos asignados

Buscar

Imagen	Nombre	Descripción	Stock	Asignar ubicaciones
	Televisor TCL	Neque porro quisquam est, qui dolore ipsum quia dolor sit amet, consectetur adipiscing elit, sed qui non incididunt lores ta porro ame. numquam eius modi tempora incididunt lores ta porro ame.	10	
	Celular Iphone 12	Neque porro quisquam est, qui dolore ipsum quia dolor sit amet, consectetur adipiscing elit, sed qui non incididunt lores ta porro ame. numquam eius modi tempora incididunt lores ta porro ame.	15	
	Auricular logtech 9935	Neque porro quisquam est, qui dolore ipsum quia dolor sit amet, consectetur adipiscing elit, sed qui non incididunt lores ta porro ame. numquam eius modi tempora incididunt lores ta porro ame.	14	
	Televisor Riviera	Neque porro quisquam est, qui dolore ipsum quia dolor sit amet, consectetur adipiscing elit, sed qui non incididunt lores ta porro ame. numquam eius modi tempora incididunt lores ta porro ame.	15	

Items per page: 4 1 - 4 of 4

Se puede realizar búsquedas de uno o varios productos a través del buscador que se encuentra encima de la tabla de los productos asignados.

Para asignar las ubicaciones por donde se haya pasado con un producto se debe presionar el siguiente botón.

Asignar
ubicaciones



Al presionar el botón se abrirá la siguiente ventana.

WooMap Inicio Opciones de Proveedor Bienvenido/a, Sebastian Mayorga Cerrar Sesión

Asignación de ubicaciones del producto

Auricular logitech G935



Descripción: Neque porro quisquam est, qui dolere ipsum quia dolor sit amet, consectetur adipiscing velit, sed qui non incidunt foret fa porro ante. Namquam eius modi tempora incidunt foret fa porro ante.

Cantidad restante: 14

Controlador asignado: Israel Lopez

Teléfono del controlador: 0971512127

Seleccionar ubicaciones:



Limpiar ubicaciones **Agregar ubicaciones**

Ubicaciones registradas



Eliminar todos los puntos

La ventana esta conformada por dos secciones, en la primera se despliega la información del producto seleccionado como su nombre, descripción, stock y también el controlador encargado de validar cada uno de los puntos que se registre en el producto.

Auricular logitech G935



Descripción: Neque porro quisquam est, qui dolore ipsum quia dolor sit amet, consectetur adipisci velit, sed quia non incidunt lores ta porro ame. numquam eius modi tempora incidunt lores ta porro ame.

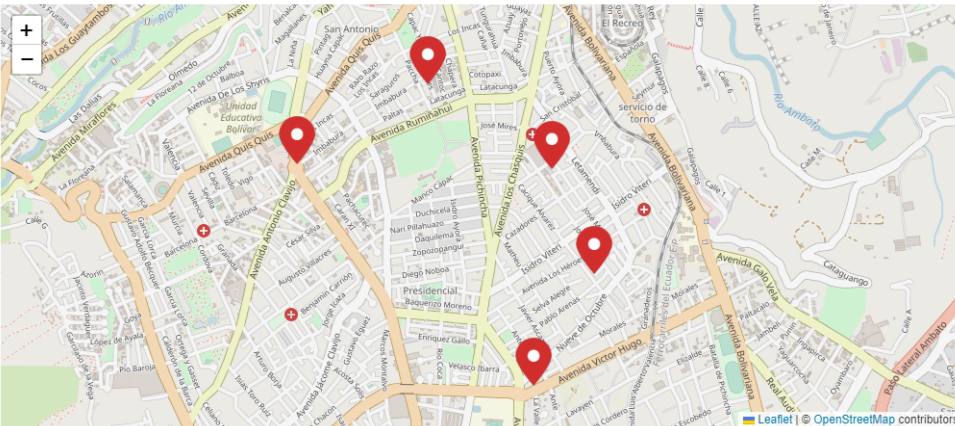
Cantidad restante: 14



Controlador asignado: Israel Lopez

Teléfono del controlador:    

También en la primera sección existe un mapa para agregar las ubicaciones por donde se haya pasado con del producto.

Seleccionar ubicaciones:



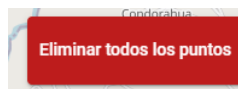
Si se requiere remover las ubicaciones seleccionadas tiene el botón de limpiar ubicaciones.



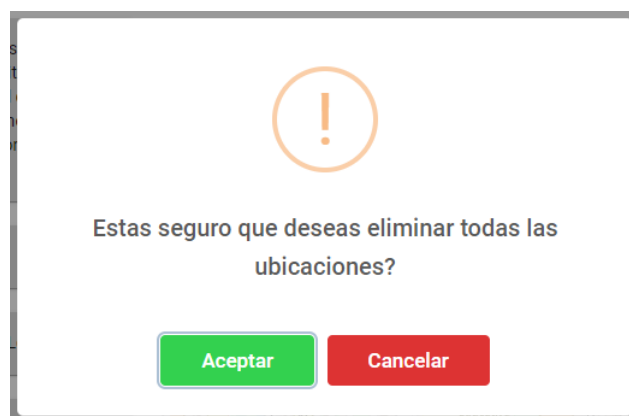
Cuando ya se agreguen las ubicaciones necesarias al producto estas se reflejaran en la sección de ubicaciones registradas, en este mapa el ícono de cada ubicación será rojo por defecto debido a que el controlador asignado todavía no las valida, caso contrario si alguna ubicación se encuentra validada esta se tornará de color verde.



Se puede eliminar todos los puntos registrados presionando en el botón Eliminar todos los puntos.



Al dar clic en dicho botón aparece el siguiente mensaje de confirmación.



Al dar clic en una de las ubicaciones agregadas, aparece un cuadro de texto conformado por una observación y dos botones que nos permiten editar o eliminar la ubicación seleccionada.



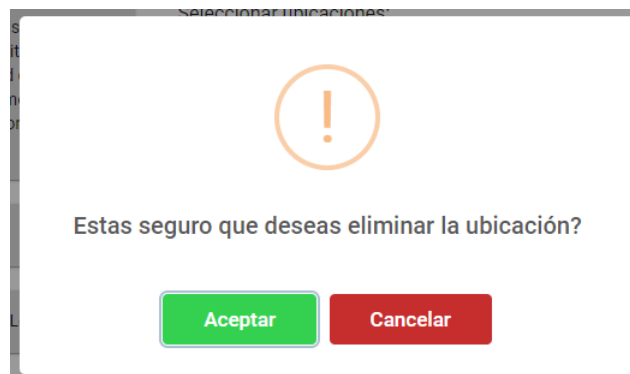
Al presionar en editar se abre la siguiente ventana emergente, donde se puede modificar la ubicación seleccionada.



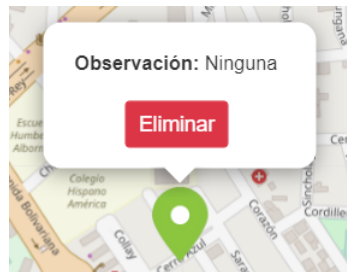
Si quiere eliminar el punto registrado se debe presionar el botón eliminar.



Una vez presionado dicho botón se despliega el siguiente mensaje de confirmación.



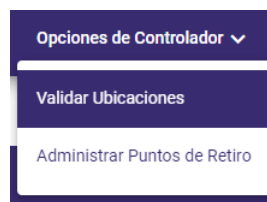
Para puntos validados no aparece la opción de editar, solo se muestra la observación que haya registrado el controlador y el botón de eliminar.



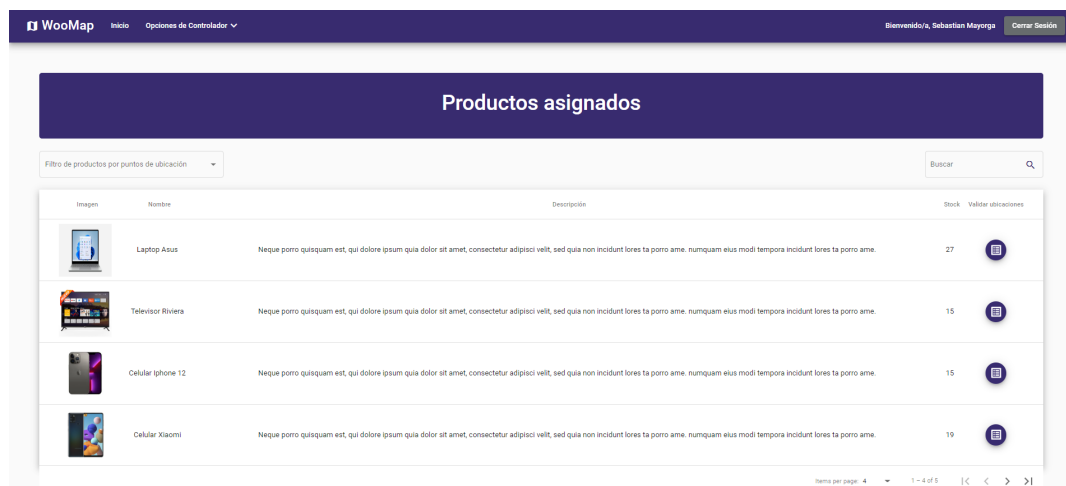
3.2.4. Rol de controlador

3.2.4.1. Validar ubicaciones

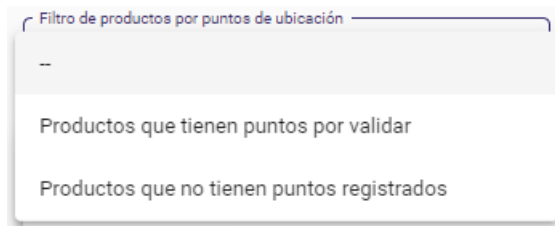
De las opciones del menú que tiene a disponibilidad el controlador la primera es validar ubicaciones.



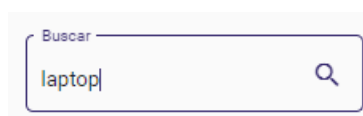
Al ingresar en esta opción se abre la siguiente ventana.



Se tiene un filtro de productos con las opciones de productos que tienen puntos de ubicación por validar y los productos que no tienen puntos registrados.



Se puede realizar búsquedas de uno o varios productos a través del nombre, descripción o el stock.



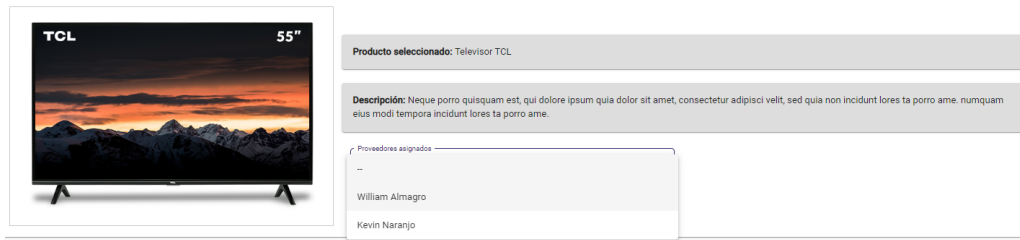
Para validar las ubicaciones que hayan registrado los proveedores de un producto se debe dar clic en el botón de validar ubicaciones.



Al presionar dicho botón se muestra la siguiente ventana.



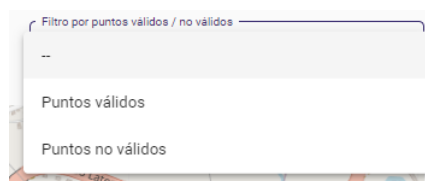
Como se puede observar se despliega la información del producto seleccionado y un combobox en el que se escoge el proveedor asignado al producto para la respectiva validación de sus puntos registrados (pueden ser varios los proveedores que estén asignados al producto).



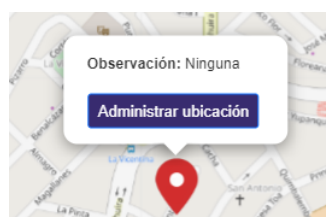
Al seleccionar un proveedor las ubicaciones que haya registrado se cargan en un mapa que aparece en la misma ventana.



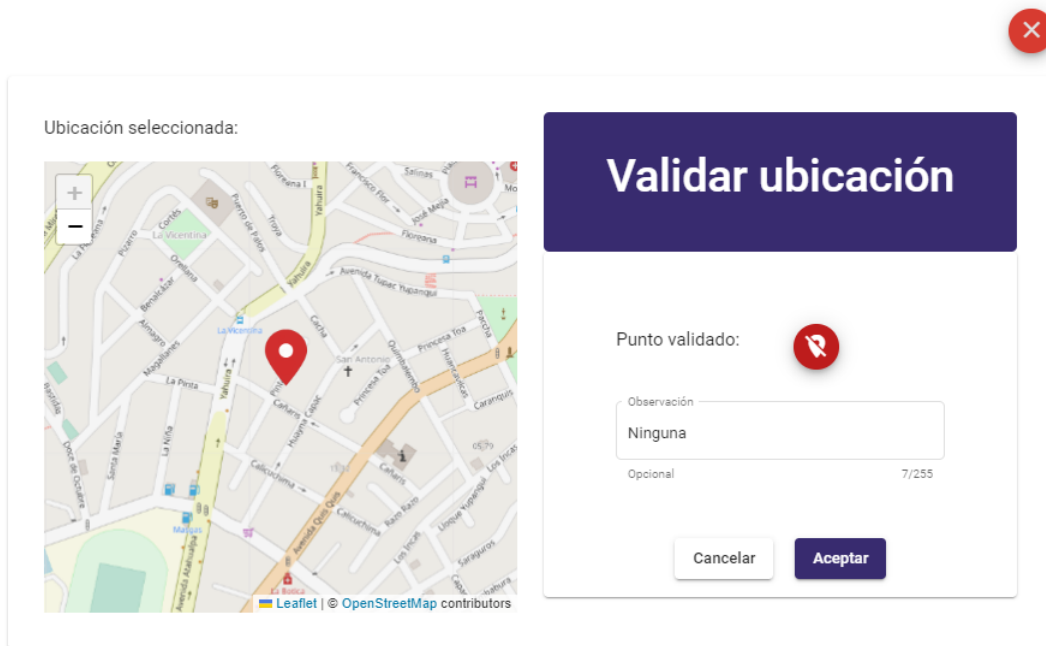
Se tiene a disposición un filtro, el cual tiene las opciones de puntos validados y aquellos que aún no se encuentran validados.



Para validar cualquier punto se debe dar clic encima del mismo, de esta manera se abre una caja de texto con la observación de dicho punto y un botón para administrar la ubicación.

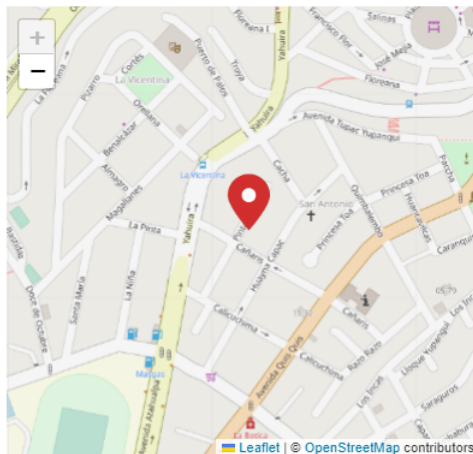


Al dar clic en el botón de administrar ubicación se abre la siguiente ventana emergente.



Dicha ventana muestra la ubicación del punto seleccionado.

Ubicación seleccionada:



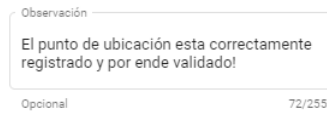
Para para cambiar el estado de validación se debe presionar el siguiente botón.



Dicho botón al ser presionado su color se tornará verde indicando que la ubicación será validada.

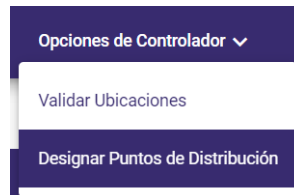


La ventana emergente también tiene un cuadro de texto para poner una observación en el caso de que sea necesario.

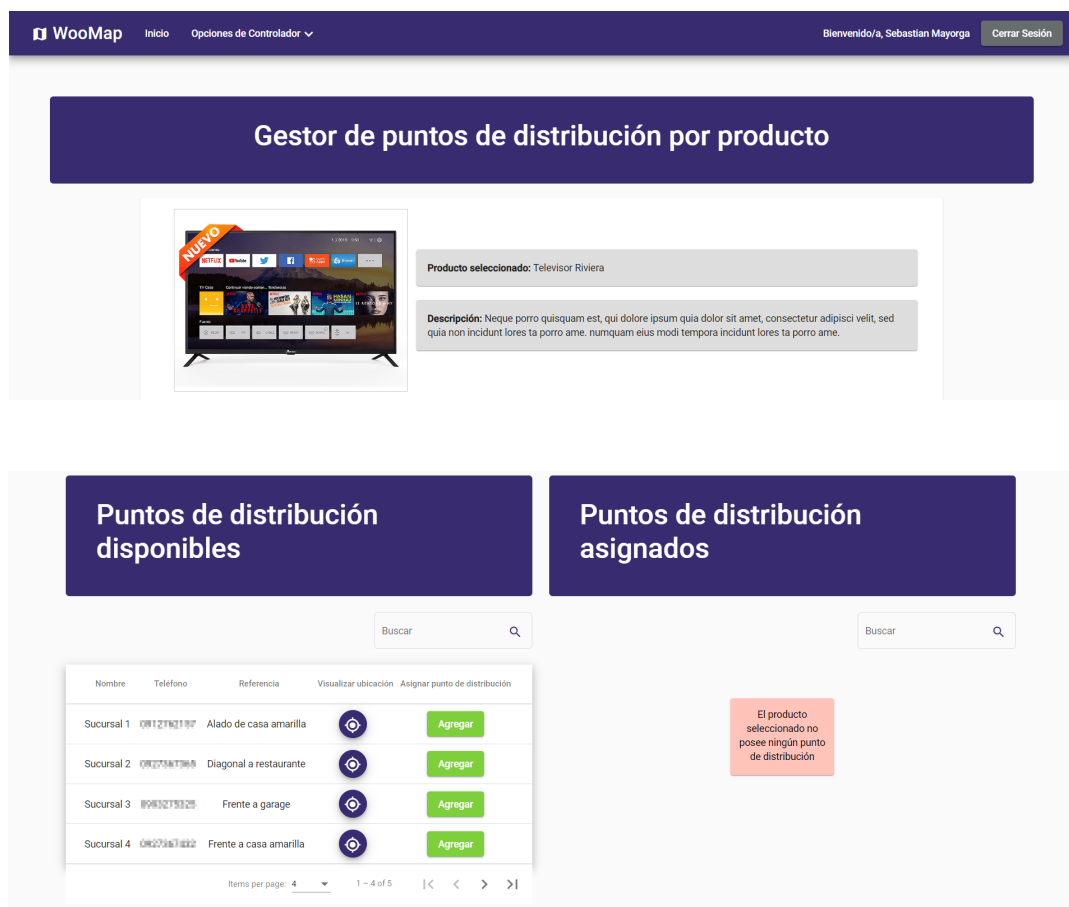


3.2.4.2. Designar puntos de distribución

La segunda opción que tiene disponible el controlador es la de designar puntos de distribución.



Cuando se ingresa a dicha opción se muestra la siguiente ventana.



En la primera sección se muestra la información del producto seleccionado.



Producto seleccionado: Televisor Riviera

Descripción: Neque porro quisquam est, qui dolore ipsum quia dolor sit amet, consectetur adipiscing velit, sed quia non incidunt lores ta porro ame. numquam eius modi tempora incidunt lores ta porro ame.

La segunda sección está dividida por los puntos de distribución que el sistema tenga registrado y los puntos de distribución que se han asignado al producto seleccionado, cada uno tendrá su buscador respectivo.

Puntos de distribución disponibles

Buscar

Nombre	Teléfono	Referencia	Visualizar ubicación	Asignar punto de distribución
Sucursal 1	0012345678	Alado de casa amarilla		<input type="button" value="Agregar"/>
Sucursal 2	0012345678	Diagonal a restaurante		<input type="button" value="Agregar"/>
Sucursal 3	0012345678	Frente a garage		<input type="button" value="Agregar"/>
Sucursal 4	0012345678	Frente a casa amarilla		<input type="button" value="Agregar"/>

Items per page: 4 | 1 - 4 of 5 | < > >>


Puntos de distribución asignados

Buscar

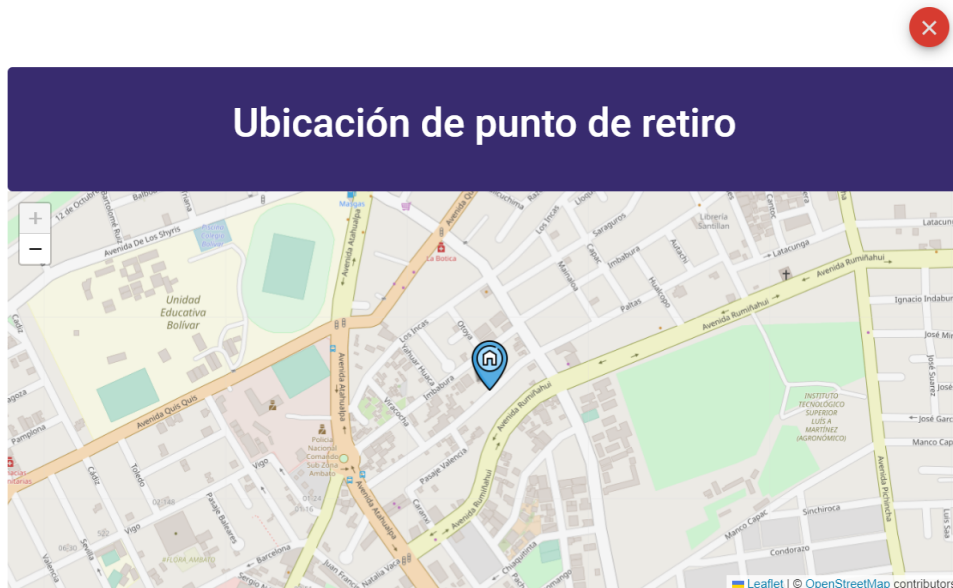
El producto seleccionado no posee ningún punto de distribución

En puntos de distribución disponibles se tiene dos botones por punto listado.

Visualizar ubicación Asignar punto de distribución



El primer botón que solo tiene un icono sirve para visualizar la ubicación del punto de distribución disponible, a través de una ventana emergente.



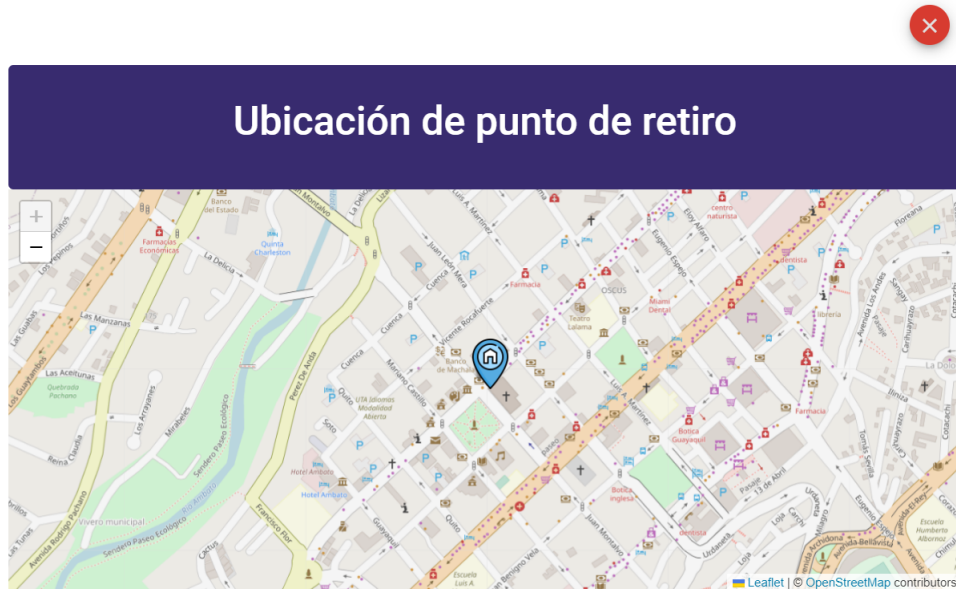
Respecto al segundo botón sirve para asignar el punto de distribución al producto seleccionado.



De igual manera los puntos de distribución asignados al producto tiene dos botones.



El primer botón visualiza la ubicación del punto de distribución asignado, a través de una ventana emergente.



El segundo botón remueve la asignación del punto de distribución del producto en el que se encuentra, para esto nos aparece el siguiente mensaje de confirmación.

