

UNIVERSIDAD MAYOR DE SAN ANDRÉS
FACULTAD DE INGENIERÍA
CARRERA DE INGENIERÍA ELECTRÓNICA



PROYECTO DE GRADO

“SISTEMA DE CONTROL DE PRODUCCIÓN Y ADMINISTRATIVO DE CALZADO”

PARA OPTAR AL TÍTULO DE:
INGENIERO ELECTRÓNICO

POSTULANTE: UNIV. REYNALDO EDWIN CUNO LUQUE

TUTOR: ING. OSCAR MARCELO POMA CALLE

LA PAZ – BOLIVIA

2023



**UNIVERSIDAD MAYOR DE SAN ANDRÉS
FACULTAD DE INGENIERIA**



LA FACULTAD DE INGENIERIA DE LA UNIVERSIDAD MAYOR DE SAN ANDRÉS AUTORIZA EL USO DE LA INFORMACIÓN CONTENIDA EN ESTE DOCUMENTO SI LOS PROPÓSITOS SON ESTRICTAMENTE ACADÉMICOS.

LICENCIA DE USO

El usuario está autorizado a:

- a) Visualizar el documento mediante el uso de un ordenador o dispositivo móvil.
- b) Copiar, almacenar o imprimir si ha de ser de uso exclusivamente personal y privado.
- c) Copiar textualmente parte(s) de su contenido mencionando la fuente y/o haciendo la cita o referencia correspondiente en apego a las normas de redacción e investigación.

El usuario no puede publicar, distribuir o realizar emisión o exhibición alguna de este material, sin la autorización correspondiente.

TODOS LOS DERECHOS RESERVADOS. EL USO NO AUTORIZADO DE LOS CONTENIDOS PUBLICADOS EN ESTE SITIO DERIVARA EN EL INICIO DE ACCIONES LEGALES CONTEMPLADAS EN LA LEY DE DERECHOS DE AUTOR.

Dedicatoria

En el día de culminación de este proyecto de grado, quiero expresarles mis más profundos agradecimientos y dedicarles este logro con todo mi amor y gratitud a mis padres Andrés y Francisca, gracias por ser mi mayor fuente de apoyo y aliento incondicional a lo largo de toda mi vida.

Agradecimientos

Quiero agradecer en primer lugar a Dios por darme la fortaleza y perseverancia para culminar este proyecto de grado con éxito, por darme salud y por renovar mis fuerzas en momentos de flaqueza para poder llegar hasta donde ahora estoy.

Mi agradecimiento especial a mis padres y familiares por su amor incondicional, aliento y apoyo emocional durante mi trayectoria académica. Su confianza en mí fue un motor para superar los desafíos y alcanzar mis metas.

Agradezco sinceramente a mi docente Ing. Freddy Valle Velasquez, por su orientación, apoyo y dedicación a lo largo de todo el proceso de desarrollo del proyecto. Sus conocimientos y consejos han sido invaluable para el éxito de este proyecto.

Agradezco a mi tutor Ing. Oscar Marcelo Poma Calle, cuya colaboración, consejos, recomendaciones y disposición fueron fundamentales para el desarrollo de este proyecto.

No puedo dejar de agradecer a todos mis amigos y compañeros de clase por compartir conmigo momentos de estudio, risas y camaradería. Su compañía hizo más ameno este camino académico.

Quiero reconocer y agradecer a la Lic. Ana Isabel Rocha Romo, por abrirme las puertas de la Empresa de Calzado Omar Alex, por la colaboración y disposición de su tiempo para el desarrollo de este proyecto.

¡De todo corazón muchísima gracias!

RESUMEN

El presente proyecto de grado se enfoca en el diseño, desarrollo e implementación de un sistema para el control de producción y administración de calzado en una empresa del sector. El sistema tiene como objetivo mejorar la eficiencia y la gestión en todas las etapas del proceso de producción y comercialización de calzado.

En la fase inicial del proyecto, se realizó un análisis exhaustivo de las necesidades y requerimientos de la empresa. En la parte de control de producción, el sistema permitirá gestionar los pedidos de calzado, realizar un seguimiento en tiempo real del proceso de fabricación y productos terminados, obtener informes y estadísticas para una mejor toma de decisiones. Así mismo, la administración de calzado se verá optimizada mediante la gestión de catálogos de productos, control de precios, seguimiento de ventas, gestión de clientes y pedidos, entre otras funcionalidades que facilitarán el trabajo del equipo de venta.

El sistema se desarrolló utilizando tecnologías como el lenguaje de programación PHP, HTML, CSS y base de datos MariaDB que permiten crear un software moderno y seguro garantizando un funcionamiento eficiente y una adecuada protección de los datos. En el desarrollo de software, se utilizó una metodología ágil, análisis y diseño orientados a objetos con UML (Lenguaje de Modelado Unificado) para visualizar y documentar los requisitos y diseños del sistema de manera clara y concisa.

Como resultado de esta combinación, se logró desarrollar un sistema con una interfaz amigable e intuitiva que permite a los usuarios acceder fácilmente a las diferentes funcionalidades logrando satisfacer las necesidades de la empresa de calzado.

Palabras clave: Sistema, control de producción, calzado, pedidos, PHP, MariaDB.

ABSTRACT

This degree project focuses on the design, development, and implementation of a system for production control and administration footwear in a company in the sector. The system aims to improve efficiency and management across all stages of footwear production and marketing process.

During the initial phase of the project, a exhaustive analysis of the company's needs and requirements was carried out. In the production control part, the system will enable footwear order management, real-time monitoring of the manufacturing process and finished products, and provide reports and statistics for better decision-making. Additionally, the administration of footwear will be optimized through the management of products catalogs, price control, sales monitoring, customer and order management, and other features that will facilitate the sales team's work.

The system was developed using technologies such as PHP, HTML, CSS, and MariaDB database, ensuring a modern and secure software, guaranteeing efficient operation and adequate data protection. Agile methodology was employed in software development, object-oriented analysis and design using UML (Unified Modeling Language) to provide clear and concise visualization and documentation of system requirements and designs.

As a result of this combination, it was possible to develop a system with user-friendly and intuitive interface, allowing users easy access to various functionalities, effectively meeting the footwear company's needs.

Keywords: System, production control, footwear, orders, PHP, MariaDB.

Índice de contenido

1	Introducción	1
1.1	Justificación.....	1
1.2	Planteamiento del problema	3
2	Marco referencial	4
2.1	Introducción	4
2.2	Ingeniería de software	4
2.3	Calidad del Software	6
2.4	Ingeniería de software basada en UML.....	8
2.4.1	Análisis de requisitos:	9
2.4.2	Diseño.....	9
2.4.3	Implementación.....	9
2.4.4	Pruebas y validación.....	9
2.4.5	Mantenimiento y evolución.....	10
2.5	Patrones de diseño	10
2.5.1	Ventajas de usar patrones de diseño.....	11
2.6	Patrones de arquitectura	12
2.7	MVC (Modelo-Vista-Controlador)	13
2.7.1	Elementos del patrón MVC.....	16
2.7.2	Modelo (Model)	16

2.7.3	Vista (View)	17
2.7.4	Controlador (Controller)	17
2.8	Bases de datos	17
2.8.2	Modelo Entidad-Relación (ER).....	20
2.9	Herramientas para la implementación.....	21
2.9.1	MariaDB.....	21
2.9.2	PHP.....	23
2.9.3	JavaScript	24
2.9.4	CSS.....	28
2.9.5	HTML.....	28
2.9.6	AdminLTE	30
3	Estado del arte.....	32
3.1	Contextualización.....	32
3.2	Clasificación.....	33
3.2.1	Diseño de un sistema de gestión de inventario de materia prima de la empresa fábrica de calzado ECCO de Nicaragua, S.A. (Hurtado, Molina, & Vega, 2018).....	33
3.2.2	Diseño de un sistema de ventas en Calzados Gorillaz (Guerra, 2022).	34
3.2.3	Desarrollo de un sistema de gestión de inventarios para el control de materias primas y productos terminados (Gayón & Ospina, 2019).	35

3.2.4	Gestión, control de materia prima, venta de productos e inventarios usando Supply Chain Management (Vasquez, 2020).	36
3.2.5	Sistema web de administración y control del almacenamiento de artículos en el almacén central y en subalmacenes (Espejo, 2020).	36
3.3	Categorización	37
4	Objetivos y metodología	39
4.1	Objetivos	39
4.1.1	Objetivo general	39
4.1.2	Objetivos específicos	39
4.2	Metodología	39
4.2.1	Descripción gráfica de los sistemas	41
4.2.2	Modelo de casos de uso	42
5	Marco aplicativo	50
5.1	Planificación del Proyecto	50
5.1.1	Ejecución de Entrevistas	50
5.1.2	Especificación de roles y tareas	52
5.1.3	Diagrama de casos de uso	54
5.1.4	Escenario de casos de uso	55
5.2	Diagrama de flujo de datos a nivel de contexto	70
5.3	Diseño navegacional	71

5.3.1	Esquema de clases navegacionales no abstractas.....	71
5.3.2	Esquema de contexto navegacional mejorado	75
5.4	Diseño de interfaz no abstracta	77
5.5	Implementación.....	78
5.5.1	Modelo de base de datos	79
5.5.2	Interfaz de usuario.....	80
6	Conclusión y Recomendaciones	86
6.1	Conclusiones	86
6.2	Recomendaciones.....	87
7	Referencias bibliográficas.....	89
	ANEXOS	91

Índice de figuras

Figura 1 <i>Interrelación entre los elementos del patrón MVC.</i>	16
Figura 2 <i>Los símbolos básicos de un diagrama de flujo de datos.</i>	42
Figura 3 <i>Símbolos de los casos de uso.</i>	44
Figura 4 <i>Escenario de caso de uso.</i>	47
Figura 5 <i>Identificación de roles.</i>	52
Figura 6 <i>Diagrama de caso de uso de negocio.</i>	55
Figura 7 <i>Diagrama de flujo de datos a nivel de contexto del sistema.</i>	70
Figura 8 <i>Esquema de clase navegacional no abstracta para el administrador.</i>	72
Figura 9 <i>Esquema de clase navegacional no abstracta para el director general.</i>	73
Figura 10 <i>Esquema de clase navegacional no abstracta para el diseñador.</i>	74
Figura 11 <i>Esquema de clase navegacional no abstracta para el jefe de producción.</i>	74
Figura 12 <i>Esquema de contexto navegacional mejorado para el administrador.</i>	75
Figura 13 <i>Esquema de contexto navegacional mejorado para el director general.</i>	76
Figura 14 <i>Esquema de contexto navegacional mejorado para el diseñador.</i>	76
Figura 15 <i>Esquema de contexto navegacional mejorado para el jefe de producción.</i>	77
Figura 16 <i>Diseño de interfaz no abstracta para pedidos de calzado.</i>	78
Figura 17 <i>Modelo de base de datos del Sistema de Control de Producción y Administración de Calzado.</i>	79
Figura 18 <i>Interfaz gráfica del administrador.</i>	80
Figura 19 <i>Interfaz gráfica del director general.</i>	81
Figura 20 <i>Interfaz gráfica del consolidado de forro.</i>	81
Figura 21 <i>Reporte de consolidado de forro.</i>	82
Figura 22 <i>Reporte de una hoja de pedido.</i>	82

Figura 23 <i>Interfaz gráfica de administración de zapatillas.</i>	83
Figura 24 <i>Interfaz gráfica para agregar una zapatilla.</i>	83
Figura 25 <i>Interfaz gráfica para administrar materiales y costos.</i>	84
Figura 26 <i>Interfaz gráfica para asignar módulos a un perfil.</i>	84
Figura 27 <i>Interfaz gráfica para administrar módulos.</i>	85

Índice de tablas

Tabla 1 <i>Patrones de diseño del Modelo-Vista-Controlador (MVC).</i>	15
Tabla 2 <i>Modelo final de cuestionario y entrevista.</i>	51
Tabla 3 <i>Identificación de tareas.</i>	53
Tabla 4 <i>Escenario de caso de uso, requerimiento de información.</i>	56
Tabla 5 <i>Escenario de caso de uso, registro de usuarios.</i>	57
Tabla 6 <i>Escenario de caso de uso, asignar roles y perfiles.</i>	58
Tabla 7 <i>Escenario de caso de uso, registro de pedidos.</i>	59
Tabla 8 <i>Escenario de caso de uso, consolidación de pedidos.</i>	60
Tabla 9 <i>Escenario de caso de uso, pedido de materia prima.</i>	61
Tabla 10 <i>Escenario de caso de uso, notas de venta.</i>	62
Tabla 11 <i>Escenario de caso de uso, registro de clientes.</i>	64
Tabla 12 <i>Escenario de caso de uso, generar cotizaciones.</i>	65
Tabla 13 <i>Escenario de caso de uso, registrar calzado.</i>	66
Tabla 14 <i>Escenario de caso de uso, administrar líneas y estilos de calzado.</i>	67
Tabla 15 <i>Escenario de caso de uso, catálogo de calzado.</i>	68

1 Introducción

En la actualidad, existe una creciente necesidad tecnológica de comunicación e información ágil y eficaz, lo que ha impulsado a todas las instituciones y organizaciones a tener una fuerte presencia en el ámbito digital y en la web.

La tecnología permite al campo empresarial, eficiencia en los flujos de información y mejores canales de comunicación, logrando un eficiente seguimiento y control de las actividades, procesos y seguimiento de una información oportuna, además una comunicación fluida entre los actores. Es considerable que toda empresa brinde la posibilidad de obtener información de todas las actividades de manera fácil, veraz para una acción oportuna.

La red de internet, es un canal de comunicación que permite a las empresas realizar acciones de manera remota o difundir la información de manera flexible y automática sin limitaciones de tiempo y espacio; brindando transparencia en los procesos de negocio, aspecto que acredita la calidad de la misma ante las entidades que rigen al ámbito empresarial, que tomando sus propios parámetros y evalúan de acuerdo a la información que durante la gestión se ha brindado.

Por lo tanto, es importante para la calidad de las empresas, tomar como estrategia el uso de herramientas de software que permiten el control y seguimiento de los procesos administrativos y las actividades que se desarrollan, reproduciendo de esta manera las buenas prácticas empresariales.

1.1 Justificación

La fábrica de calzado Omar Alex que se especializa en Zapatillas se encuentra ubicada en la Calle San Eduardo, N.º 403, Colonia San Manuel C.P. 37259 de la ciudad de León, estado de

Guanajuato – México, fue constituida a mediados de 1986 por los empresarios Francisco Rocha y Verónica Romo. Durante la pandemia del 2020 (COVID-19) se suspendieron labores obligando a cerrar la fábrica de calzado obteniendo perdidas considerables y la necesidad de despedir varios empleados. Se reanuda labores a partir de noviembre de 2021 teniendo una nueva administración a cargo de los hijos de los empresarios.

Durante muchos años los propietarios tenían la necesidad de contar con un sistema que facilite el control de producción y también la administración de la fábrica, viéndose perjudicada al no contar con información precisa y a tiempo de todo el movimiento de materia prima y la fabricación de calzado, si bien existen sistemas o software para la producción de calzado estos no cubrían todas las necesidades ya que estos fueron desarrollados de forma general o en algunos casos eran antiguos y obsoletos.

La fábrica de calzado Omar Alex tuvo que darse formas de registrar los pedidos que realizaban los clientes, pedidos de materia prima que se realiza a los proveedores, pagar salarios a los empleados, todo esto usando documentos físicos los cuales con el tiempo se perdían o era difícil de dar seguimiento a cada una de las actividades que se realizaba.

A medida que fueron pasando los años se empezó a utilizar Microsoft Excel para mejorar el control y la administración de la fábrica, de cierta forma mejoró pero en el tiempo se dieron cuenta que se tenía muchos documentos Excel y no era posible relacionar la información entre estos documentos y otra vez surgía un nuevo problema, así que vieron la necesidad de contratar personal que pueda organizar estos documentos utilizando macros en Excel, sin embargo, el resultado de tener tanta información en estos documentos hizo que se volviera lento y frustrante al abrir cada archivo Excel; es por lo que al día de hoy la fábrica de calzado Omar Alex requiere un software que resuelva este problema que vienen arrastrando por muchos años.

1.2 Planteamiento del problema

Debido a que la fábrica de calzado no cuenta con un control minucioso del movimiento de materia prima existen sobrantes o faltantes ocasionando pérdidas económicas considerables; por tal razón es importante que todos los actores del proceso de registro de productos informen de las actividades realizadas para tener un seguimiento y control de producción.

Por otra parte, no se cuenta con un registro adecuado de los pedidos de productos de materia prima, dichos productos son proporcionados por distintos proveedores en grandes cantidades lo cual dificulta tener control de los mismos, al mismo tiempo no se cuenta con registros adecuados de pedidos que los clientes realizan a la fábrica de calzado, estos pedidos son de distintas líneas de calzado y de distintos estilos, además que cada línea tiene diferente tipo de numeración, todo esto y más hacen que sea dificultoso o imposible tener total control de producción de calzado y nuevamente ocasionando pérdidas económicas.

El problema central de no contar con la información necesaria de todo el movimiento de los productos o que no logran ser del todo eficientes ocasionando desconocimiento y contradicciones entre los actores debido a que no se cuenta con una base de datos centralizada dificultando el análisis y toma de decisiones oportuna en la empresa.

De lo descrito anteriormente se puede formular la siguiente pregunta:

¿De qué manera se puede implementar un sistema de control de producción y administrativo de calzado coadyuvará en brindar una mejor calidad de información y comunicación, para facilitar la administración de la fábrica de calzado Omar Alex?

2 Marco referencial

2.1 Introducción

En el marco referencial de este proyecto, se llevará a cabo una revisión exhaustiva de los conceptos, teorías y tecnologías relevantes para la gestión de la producción y administración en la industria del calzado. Se analizarán casos de éxito y experiencias previas de implementación de sistemas similares en empresas del sector, con el fin de identificar buenas prácticas y lecciones aprendidas que puedan ser aplicables a la realidad del proyecto.

Además, se examinará la literatura académica y científica relacionada con la gestión de la producción, los sistemas de información, así como las ventas específicas para la industria del calzado. Esta revisión teórica permitirá sentar las bases sólidas para el desarrollo del sistema, alineándolo con las mejores prácticas y las tendencias actuales en el ámbito de estudio.

El marco referencial servirá como punto de partida para la implementación de un sistema de control de producción y administración de calzado que se adapte a las necesidades y características de la empresa Zapatillas Omar Alex. Se buscará que el sistema propuesto sea una herramienta efectiva y eficiente para la toma de decisiones informadas, la optimización de recursos y la mejora de la experiencia del usuario, contribuyendo así al desarrollo sostenible de la empresa en el competitivo mercado del calzado.

2.2 Ingeniería de software

La ingeniería de software es una disciplina que se enfoca en aplicar principios, métodos y herramientas de ingeniería al desarrollo, diseño, implementación y mantenimiento de sistemas de software. Su objetivo principal es crear soluciones tecnológicas efectivas y confiables que satisfagan las necesidades de los usuarios y las demandas del entorno en el que operan.

Esta disciplina se basa en un enfoque sistemático y estructurado para gestionar proyectos de desarrollo de software, abarcando desde la definición de requerimientos hasta la entrega del producto final. Los ingenieros de software se encargan de analizar las necesidades de los clientes, diseñar soluciones técnicas, codificar el software, realizar pruebas para asegurar su calidad y gestionar su implementación y mantenimiento.

La ingeniería de software considera aspectos fundamentales como la eficiencia, escalabilidad, mantenibilidad, seguridad y usabilidad del software desarrollado. Para ello, se utilizan metodologías de desarrollo, modelos de diseño, estándares de codificación y prácticas de aseguramiento de calidad que permiten producir software confiable y de alto rendimiento.

El proceso de ingeniería de software involucra la colaboración entre profesionales de distintas áreas, como analistas, diseñadores, programadores, probadores y administradores de proyectos. La comunicación efectiva y la gestión adecuada de recursos son esenciales para llevar a cabo proyectos exitosos en tiempo y forma.

A continuación, citamos algunas de las definiciones de autores reconocidos en el campo de la Ingeniería de Software y son utilizadas por académicos y profesionales para describir la disciplina y su enfoque en el desarrollo de software:

"La Ingeniería de Software es la aplicación de principios y prácticas de ingeniería para el diseño, desarrollo, implementación y mantenimiento de software de calidad" (Pressman, 2014).

"La Ingeniería de Software es el estudio y la aplicación de enfoques sistemáticos, disciplinados y cuantificables para el desarrollo, operación y mantenimiento de software, que incluye la recolección sistemática de requerimientos, el diseño cuidadoso, la

verificación rigurosa, y la prueba exhaustiva del software, tanto antes como después del despliegue" (IEEE Computer Society, 2014).

"La Ingeniería de Software es el uso de teorías, métodos y herramientas de la ingeniería para el desarrollo y mantenimiento de sistemas de software que son confiables, eficientes y de alta calidad" (Sommerville, 2011).

2.3 Calidad del Software

La calidad de software es un objetivo importante en la ingeniería de software, ya que un software de alta calidad garantiza la satisfacción del usuario, minimiza errores y fallos, y aumenta la productividad y eficiencia en su uso. Para evaluar y asegurar la calidad del software, se utilizan diferentes metodologías, estándares y herramientas de aseguramiento de calidad y pruebas de software.

La Norma ISO 9000:2015 define Calidad de manera general como “Grado en el que un conjunto de características inherentes cumple con los requisitos”¹. Según la definición, La calidad de un producto, se refiere al grado en que el producto cumple con los requisitos establecidos por el cliente y otras partes interesadas. Esto significa que un producto de calidad satisface las necesidades y expectativas del cliente, y es coherente con las especificaciones, estándares y características definidas para ese producto.

La IEEE (Institute of Electrical and Electronics Engineers) tiene un estándar específico que aborda la calidad del software, conocido como IEEE 730-2014 "Estándar para el Plan para el

¹ International Organization for Standardization.

Proceso de Desarrollo de Software". Aunque este estándar se centra principalmente en el plan de desarrollo de software, también proporciona una definición de calidad de software.

Según la IEEE 730-2014, la calidad de software se define de la siguiente manera: “La calidad del software es la concordancia con los requisitos funcionales y de rendimiento explícitamente establecidos, con los estándares de desarrollo explícitamente documentados y con las características implícitas que se esperan de todo software desarrollado profesionalmente”². Esta definición resalta la importancia de cumplir con los requisitos específicos, tanto los explícitos (que están claramente definidos) como los implícitos (que se esperan de cualquier software profesional). Además, se enfoca en la importancia de seguir los estándares de desarrollo y las mejores prácticas documentadas para asegurar la calidad del software. También es importante mencionar que esta definición se encuentra en el contexto del estándar IEEE 730-2014, que se enfoca en el plan de desarrollo de software y la gestión de calidad en el proceso de desarrollo, por lo que la IEEE puede tener otras definiciones relacionadas con la calidad del software en otros estándares y documentos.

Se establece una relación para la calidad entre satisfacción de un usuario que es igual al producto satisfactorio realizado dentro del presupuesto y en los tiempos establecidos, afirmación que implica a la gestión de la calidad en el proceso de desarrollo del producto.

Pressman, por otro lado, define la calidad como: “La calidad de software es la concordancia con los requisitos funcionales y de rendimiento explícitamente establecidos, con los estándares de desarrollo explícitamente documentados y con las características implícitas que se espera de todo software desarrollado profesionalmente” (Pressman, 2014).

Algunos aspectos clave que se consideran para evaluar la calidad de software son:

² The Institute of Electrical and Electronics Engineers.

- **Funcionalidad:** La capacidad del software para proporcionar las funciones y características requeridas y esperadas por los usuarios.
- **Fiabilidad:** La capacidad del software para realizar sus funciones de manera precisa y consistente, evitando errores y fallos.
- **Usabilidad:** La facilidad con la que los usuarios pueden interactuar y utilizar el software de manera efectiva y eficiente.
- **Eficiencia:** La capacidad del software para lograr sus objetivos con un uso óptimo de recursos, como memoria, procesamiento y tiempo de respuesta.
- **Mantenibilidad:** La facilidad con la que el software puede ser modificado, corregido o mejorado con el tiempo, sin afectar su funcionamiento.
- **Portabilidad:** La capacidad del software para ser adaptado y ejecutado en diferentes plataformas o entornos.
- **Seguridad:** La capacidad del software para protegerse contra accesos no autorizados, ataques y garantizar la integridad y confidencialidad de los datos.

Por lo tanto, la calidad del software, es medido en cada proceso del desarrollo de manera que satisfaga las expectativas del usuario.

2.4 Ingeniería de software basada en UML

La ingeniería de software basada en UML se refiere al uso del Lenguaje de Modelado Unificado (UML) como una herramienta fundamental en el proceso de diseño, desarrollo y documentación de sistemas de software. UML es un lenguaje estándar de modelado gráfico que proporciona una notación visual para describir distintos aspectos de un sistema, lo que facilita la comprensión y comunicación entre los miembros del equipo de desarrollo y otras partes interesadas.

El proceso de ingeniería de software basada en UML implica varias etapas, que incluyen:

2.4.1 *Análisis de requisitos:*

En esta etapa, se recopilan y analizan los requisitos del sistema, es decir, las necesidades y expectativas de los usuarios y las partes interesadas. UML se utiliza para representar los casos de uso, diagramas de actividades y otros modelos que ayudan a entender las interacciones del sistema con los usuarios y cómo se llevarán a cabo las funcionalidades.

2.4.2 *Diseño*

En esta etapa, se desarrolla el diseño del sistema utilizando UML para representar la arquitectura, la estructura y el comportamiento del software. Los diagramas de clases, diagramas de secuencia, diagramas de estado, diagramas de componentes y otros diagramas UML son útiles para describir cómo se organizarán los componentes del sistema y cómo interactuarán entre sí.

2.4.3 *Implementación*

La fase de implementación implica la traducción del diseño del sistema en código fuente real. Los modelos UML, especialmente los diagramas de clases, a menudo se utilizan como referencia durante el desarrollo para guiar a los programadores y asegurar que el sistema se construya según las especificaciones.

2.4.4 *Pruebas y validación*

En esta etapa, se realizan pruebas para asegurar que el software cumple con los requisitos establecidos. Los modelos UML, como los diagramas de casos de uso y los diagramas de secuencia, también pueden ayudar en la creación y ejecución de pruebas funcionales.

2.4.5 *Mantenimiento y evolución*

Después de que el software ha sido desplegado, es necesario realizar mantenimiento y realizar cambios para adaptarse a nuevas necesidades o corregir errores. Los modelos UML actuales se pueden utilizar para entender el sistema y facilitar las actualizaciones.

La utilización de UML en la ingeniería de software proporciona una forma común y clara de representar y comunicar ideas y conceptos a través de diagramas gráficos. Sin embargo, es importante destacar que UML es solo una herramienta y no reemplaza una metodología de desarrollo sólida ni el juicio y experiencia de los ingenieros de software. Además, la elección de qué diagramas UML utilizar y su nivel de detalle dependerá de la complejidad y el alcance del proyecto.

Craing Larman considera que UML es una herramienta valiosa en el análisis, diseño y comunicación de sistemas de software, especialmente cuando se combina con enfoques ágiles y el uso de patrones de diseño (Larman, 2004).

2.5 Patrones de diseño

Los patrones de diseño en el desarrollo de software son soluciones probadas y reutilizables para problemas comunes que surgen al diseñar y desarrollar sistemas de software. Fueron introducidos por primera vez por los arquitectos de software Erich Gamma, Richard Helm, Ralph Johnson y John Vlissides en su libro "Design Patterns: Elements of Reusable Object-Oriented Software", también conocido como el libro "Gang of Four" (GoF), publicado en 1994.

Los patrones de diseño ofrecen una forma estandarizada y eficiente de abordar desafíos recurrentes en el diseño de software, permitiendo a los desarrolladores comunicarse de manera más efectiva sobre soluciones comunes. Cada patrón es una descripción detallada de un problema

específico que ocurre frecuentemente en el desarrollo de software y cómo se puede resolver de manera óptima.

2.5.1 Ventajas de usar patrones de diseño

Los patrones de diseño proporcionan una serie de ventajas en el desarrollo de software:

- **Mejoran la reutilización de código:** Los patrones de diseño promueven la reutilización de soluciones probadas y evitan la duplicación de código, lo que ahorra tiempo y esfuerzo en el desarrollo.
- **Facilitan el mantenimiento:** Al utilizar patrones de diseño, el código tiende a ser más estructurado y fácil de entender, lo que facilita su mantenimiento y evolución.
- **Estándares de diseño:** Los patrones de diseño se han convertido en estándares reconocidos en la industria, lo que permite a los desarrolladores hablar un lenguaje común y compartir buenas prácticas.
- **Mejoran la flexibilidad y la escalabilidad:** Los patrones de diseño ayudan a crear sistemas más flexibles y escalables, lo que permite adaptarse más fácilmente a cambios futuros en los requerimientos del software.

Es importante mencionar que, aunque los patrones de diseño son herramientas valiosas, no siempre son la solución adecuada para cada problema. Deben utilizarse con prudencia y siempre considerando el contexto y las necesidades específicas del proyecto en el que se está trabajando, por lo que no es obligatorio utilizar los patrones, solo es aconsejable en el caso de tener el mismo problema o similar que soluciona el patrón, siempre teniendo en cuenta que en un caso particular puede no ser aplicable. "Abusar o forzar el uso de los patrones puede ser un error".

2.6 Patrones de arquitectura

Los patrones de arquitectura en ingeniería de software son patrones de diseño de alto nivel que proporcionan soluciones generales para la organización y estructura de sistemas de software a nivel arquitectónico. A diferencia de los patrones de diseño que se centran en la implementación de clases y objetos, los patrones de arquitectura se enfocan en la distribución, comunicación y organización de componentes y subsistemas en un sistema completo.

Estos patrones de arquitectura abordan problemas de diseño a nivel macro, como la división de responsabilidades, la escalabilidad, la seguridad, la flexibilidad y la modularidad del sistema. Al utilizar patrones de arquitectura, los arquitectos de software pueden tomar decisiones fundamentales sobre la estructura del sistema y establecer una base sólida para el desarrollo futuro.

Algunos ejemplos comunes de patrones de arquitectura en ingeniería de software incluyen:

- Patrón Modelo-Vista-Controlador (MVC): Es un patrón de arquitectura que separa la lógica de negocio y la interfaz de usuario en tres componentes distintos: Modelo, Vista y Controlador. Esto mejora la modularidad y facilita el mantenimiento y la evolución del sistema.
- Patrón Arquitectura en Capas: Divide el sistema en capas lógicas, como capa de presentación, capa de lógica de negocio y capa de almacenamiento de datos. Cada capa se comunica con la capa adyacente a través de interfaces bien definidas, lo que mejora la escalabilidad y la mantenibilidad.
- Patrón Arquitectura Orientada a Servicios (SOA): Es un enfoque que organiza el sistema en servicios independientes y autónomos que se comunican a través de

interfaces estándar. Esto mejora la reutilización de componentes y permite una integración más flexible con otros sistemas.

- Patrón Microservicios: Es una variante de SOA en la que el sistema se descompone en pequeños servicios independientes que se implementan y despliegan de forma independiente. Esto mejora la escalabilidad y facilita el desarrollo ágil.
- Patrón Modelo-Actores: Es un patrón que se centra en el modelo de concurrencia y la comunicación entre componentes utilizando el modelo de actores. Es especialmente útil para sistemas altamente concurrentes y distribuidos.

Estos patrones de arquitectura proporcionan una guía sólida para diseñar sistemas de software robustos, escalables y mantenibles. Al implementar estos patrones, los arquitectos y desarrolladores pueden aprovechar las mejores prácticas y experiencias acumuladas en la industria para construir sistemas de software de alta calidad.

2.7 MVC (Modelo-Vista-Controlador)

El patrón de diseño MVC de las siglas en inglés (Model-Views-Controllers) fue propuesto en 1994 por cuatro autores Erich Gamma, Richard Helm, Ralph Johnson y John Vlissides en su libro titulado "Design Patterns - Elements of Reusable Object-Oriented Software.

MVC es un patrón de arquitectura de software ampliamente utilizado en el desarrollo de aplicaciones para separar la lógica de negocio, la presentación de la interfaz de usuario y la gestión de eventos en tres componentes distintos. Esta separación mejora la modularidad, la reusabilidad y la mantenibilidad del sistema.

El patrón MVC se ha creado para abordar varios problemas comunes que surgen en el desarrollo de software, por ejemplo, el crear código espagueti como varios lo han denominado, la

cual consiste en la combinación de código HTML, PHP y JAVASCRIPT en un mismo archivo el cual presenta muchas inconvenientes al realizar modificaciones o actualizaciones, especialmente en aplicaciones que tienen una interfaz de usuario. Algunas de las razones principales para la creación del patrón MVC son:

- Separación de preocupaciones: El patrón MVC permite separar claramente la lógica de negocio (Modelo), la presentación de la interfaz de usuario (Vista) y el manejo de eventos y la coordinación (Controlador). Esta separación mejora la organización y la estructura del código, facilitando la comprensión y el mantenimiento del sistema.
- Reusabilidad y mantenibilidad: Al tener componentes claramente definidos y separados, como el Modelo y la Vista, es más fácil reutilizarlos en diferentes partes del sistema o en proyectos futuros. Además, si se necesitan realizar cambios o mejoras, es más sencillo modificar un componente sin afectar a los demás, lo que mejora la mantenibilidad del sistema.
- Facilita el desarrollo en equipo: El patrón MVC promueve la colaboración en el desarrollo de software. Los desarrolladores pueden trabajar en diferentes componentes (Modelo, Vista o Controlador) de forma independiente, lo que permite un desarrollo paralelo y eficiente.
- Flexibilidad en la presentación: La separación entre el Modelo y la Vista permite que un mismo Modelo sea presentado de diferentes maneras, según las necesidades de la aplicación. Por ejemplo, un mismo conjunto de datos puede ser presentado en una interfaz gráfica, una interfaz de línea de comandos o en un servicio web.
- Adaptabilidad a cambios en la interfaz de usuario: La arquitectura MVC facilita la modificación o sustitución de la Vista sin afectar la lógica del Modelo o el flujo de

control del sistema. Esto permite adaptarse a cambios en los requisitos de la interfaz de usuario sin tener que realizar cambios significativos en otras partes del sistema.

Por estas razones, el patrón Modelo-Vista-Controlador se ha creado para mejorar la organización, flexibilidad y mantenibilidad de las aplicaciones de software con interfaz de usuario. Al separar claramente las responsabilidades y funciones en componentes distintos, el desarrollo de software se vuelve más estructurado, eficiente y colaborativo. Esta arquitectura ha demostrado ser ampliamente efectiva y ha sido ampliamente adoptada en el desarrollo de aplicaciones en diferentes plataformas y lenguajes de programación.

El patrón MVC está basado al desarrollo Orientado a Objetos donde destacan actualmente 23 Modelos MVC que se encuentran clasificados en 3 categorías principales: Estructural, Creacional y Comportamiento.

Tabla 1

Patrones de diseño del Modelo-Vista-Controlador (MVC).

CATEGORIA	DESCRIPCIÓN
Patrón creacional	Este patrón de diseño proporciona al programador crear instancias de objetos de una forma más flexible y de esta forma administre que tipo de objetos deben crearse para cada caso.
Patrón estructural	Su diseño esta principalmente basado en la herencia aplicada a la creación de interfaces para la creación de nuevos objetos y obtener nuevas funcionalidades.
Patrón de comportamiento	Se caracteriza por ser un patrón de diseño que permite la comunicación entre objetos.

Nota. Describe en 3 categorías principales el patrón de diseño Modelo-Vista-Controlador (MVC).

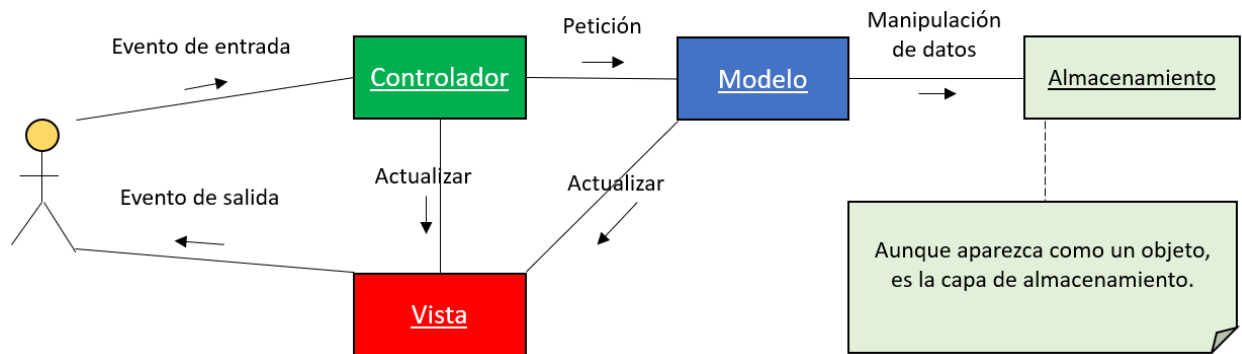
Elaboración propia.

2.7.1 Elementos del patrón MVC

La figura ilustra los elementos del patrón y la interrelación entre estos.

Figura 1

Interrelación entre los elementos del patrón MVC.



Nota. Ilustración de los elementos que componen el patrón MVC. Tomada de (Fernández & Díaz, 2012).

Los elementos que componen el patrón MVC son:

- Modelo: datos y reglas de negocio.
- Vistas: muestra la información del modelo al usuario.
- Controlador: gestiona las entradas del usuario.

2.7.2 Modelo (Model)

Representa la lógica de negocio y los datos del sistema. Es responsable de manejar la manipulación y la gestión de los datos, así como de la realización de las operaciones y cálculos necesarios para el funcionamiento del sistema. El Modelo es independiente de la interfaz de usuario y se comunica con el Controlador para recibir y enviar datos.

2.7.3 Vista (View)

Representa la interfaz de usuario y es responsable de la presentación de los datos del Modelo al usuario. La Vista no realiza ningún procesamiento de datos; simplemente muestra la información y envía eventos de interacción del usuario al Controlador. Puede haber múltiples vistas para un mismo Modelo, lo que permite una mayor flexibilidad y adaptabilidad en la presentación de los datos.

2.7.4 Controlador (Controller)

Actúa como intermediario entre el Modelo y la Vista. Recibe eventos de interacción del usuario desde la Vista y toma decisiones en función de esos eventos. Luego, realiza las acciones apropiadas en el Modelo y actualiza la Vista con los cambios realizados. El Controlador se encarga de coordinar las acciones del Modelo y la Vista, manteniéndolos separados entre sí.

2.8 Bases de datos

Con la aparición de las computadoras y los programas de gestión de base de datos las empresas organizan la información sin demasiado esfuerzo, realizando un tratamiento eficiente de administración, organización, ordenación, control y presentación de la información de forma clara y precisa. Esta información es conocida en la estructura de una base de datos, y permite que los sistemas gestores de bases de datos almacenen y administren grandes volúmenes de información.

Una base de datos es una colección coherente de datos relacionados que representan algunos aspectos del mundo real. Por datos entendemos que son hechos conocidos que pueden registrarse y que tienen significado implícito como por ejemplo propiedades y atributos de un ejemplar.

Es posible abstraer una serie de características de la realidad a través del modelo de datos, que a través de un conjunto de conceptos, reglas y convenciones bien definidas permiten describir y manipular los datos de un cierto mundo real que se almacenará en una base de datos.

En general, el objetivo del diseño de una base de datos relacional es generar un conjunto de esquemas de relaciones que permitan almacenar la información con un mínimo de redundancia, pero que a la vez faciliten la recuperación de la información.

El autor del libro “Fundamentos de bases de datos” (Silberschatz, Korth, & Sudarshan, 2002) indica varias etapas de diseño de una base de datos que se verá a continuación, pero antes, es importante tener en cuenta que estas etapas son una guía general, y el proceso de diseño de bases de datos puede variar según el proyecto y las necesidades específicas.

2.8.1.1 Recolección de requisitos

En esta etapa, se identifican y documentan los requisitos del sistema de bases de datos. Se investiga y comprende qué datos se deben almacenar, cómo se utilizan y cuáles son las necesidades de los usuarios.

2.8.1.2 Diseño conceptual

Se crea un modelo de alto nivel que representa las principales entidades, atributos y relaciones del sistema. Usualmente, se utiliza el modelo Entidad-Relación (ER) para representar esta etapa.

2.8.1.3 Diseño lógico

En esta etapa, el modelo conceptual se traduce en un esquema relacional que consiste en tablas, atributos y relaciones en términos de claves primarias y foráneas.

2.8.1.4 Normalización

Es el proceso de aplicar reglas de normalización para eliminar la redundancia y las anomalías de actualización en el esquema relacional. El objetivo es mejorar la integridad y la eficiencia del sistema de bases de datos.

2.8.1.5 Diseño físico

En esta etapa, se definen los detalles específicos de implementación, como el tipo de datos para cada atributo, la estructura de almacenamiento y la creación de índices para mejorar el rendimiento.

2.8.1.6 Implementación

Aquí se crea la base de datos física en un sistema de gestión de bases de datos (DBMS) específico utilizando las definiciones del diseño físico.

2.8.1.7 Carga inicial de datos

Se realiza la inserción inicial de datos en la base de datos recién creada a partir de fuentes existentes.

2.8.1.8 Pruebas y ajustes

Se realizan pruebas exhaustivas para asegurar que la base de datos funcione correctamente y se ajusta según sea necesario.

2.8.1.9 Mantenimiento y evolución

Una vez en producción, se realiza el mantenimiento continuo de la base de datos para abordar cambios en los requisitos y mejorar el rendimiento.

2.8.2 Modelo Entidad-Relación (ER)

Un modelo Entidad-Relación (ER) es una herramienta conceptual utilizada en el diseño de bases de datos para representar y describir la estructura lógica de la base de datos de una manera visual y comprensible. Es una de las etapas del diseño de bases de datos, específicamente la etapa de diseño conceptual.

Según el autor Abraham Silberschatz y sus coautores en el libro "Fundamentos de bases de datos", un modelo Entidad-Relación se basa en los siguientes conceptos fundamentales:

2.8.2.1 Entidad

Representa un objeto o concepto en el mundo real que es distinguible y sobre el cual se necesita almacenar información. Puede ser una persona, lugar, objeto, evento, etc. En el modelo ER, las entidades se representan mediante rectángulos.

2.8.2.2 Atributo

Es una propiedad o característica que describe una entidad. Los atributos definen qué tipo de información se almacenará para cada entidad. En el modelo ER, los atributos se representan mediante elipses conectados a las entidades a las que pertenecen.

2.8.2.3 Relación

Representa una asociación o conexión entre dos o más entidades. Las relaciones capturan cómo las entidades se relacionan entre sí en el mundo real. En el modelo ER, las relaciones se representan mediante rombos conectados a las entidades que participan en la relación.

2.8.2.4 Cardinalidad

Indica la cantidad de ocurrencias de una entidad que pueden estar relacionadas con la otra entidad a través de una relación. Por ejemplo, una relación puede ser uno a uno, uno a muchos o muchos a muchos.

2.8.2.5 Clave primaria

Es un atributo (o conjunto de atributos) que identifica de manera única a cada instancia de una entidad. Sirve para garantizar la unicidad y la identificación de las entidades.

2.8.2.6 Clave foránea

Es un atributo (o conjunto de atributos) en una entidad que hace referencia a la clave primaria de otra entidad y establece una relación entre ellas.

El modelo Entidad-Relación es una herramienta gráfica que permite a los diseñadores de bases de datos visualizar y comunicar la estructura lógica del sistema de bases de datos de una manera fácil de entender. A partir del modelo ER, se puede proceder a la fase de diseño lógico, donde se traduce el modelo conceptual a un esquema relacional que puede ser implementado en un Sistema de Gestión de Bases de Datos (DBMS)³.

2.9 Herramientas para la implementación

2.9.1 MariaDB

MariaDB es un sistema de gestión de bases de datos relacional de código abierto. Es una bifurcación (fork) de MySQL, lo que significa que se desarrolló a partir del código fuente de MySQL y ha evolucionado de manera independiente. Fue creada por Michael "Monty" Widenius,

³ Database Management System, y actúa como una capa intermedia entre las aplicaciones de software y los datos almacenados en la base de datos física.

uno de los desarrolladores originales de MySQL, después de que MySQL fuera adquirido por Oracle en 2010.

Al ser una bifurcación de MySQL, MariaDB comparte muchas características y comandos con este último, lo que permite una fácil migración entre ambos sistemas. Sin embargo, también ha introducido mejoras y nuevas características, y se ha centrado en ofrecer una base de datos relacional de alto rendimiento, confiable y de código abierto.

Algunas de las características notables de MariaDB incluyen:

- **Almacenamiento de motor:** MariaDB admite varios motores de almacenamiento, incluyendo el motor InnoDB (que también se utiliza en MySQL) y otros como Aria, MyRocks y XtraDB.
- **Seguridad:** Ofrece opciones avanzadas de seguridad, incluyendo el cifrado de datos en reposo y en tránsito, y la autenticación mediante complementos de autenticación como PAM (Pluggable Authentication Modules) y LDAP.
- **Rendimiento:** MariaDB ha implementado mejoras de rendimiento y optimizaciones en comparación con MySQL, lo que lo convierte en una opción popular para aplicaciones de alto rendimiento.
- **Compatibilidad con MySQL:** Es compatible con aplicaciones y herramientas diseñadas para MySQL, lo que facilita la migración de sistemas existentes.
- **Características nuevas:** MariaDB ha agregado características adicionales, como los tipos de datos JSON y Sequences, que no están presentes en versiones antiguas de MySQL.

MariaDB se ha convertido en una opción popular para aplicaciones web, sistemas de gestión de contenido (CMS), aplicaciones empresariales y muchas otras aplicaciones que requieren una base de datos relacional robusta y escalable. Además, varias distribuciones de Linux, como CentOS, Fedora y openSUSE, han reemplazado MySQL con MariaDB como el sistema de gestión de bases de datos predeterminado.

2.9.2 PHP

PHP es un lenguaje de programación de código abierto especialmente diseñado para el desarrollo web. Su nombre es un acrónimo recursivo que significa "PHP: Hypertext Preprocessor" ("PHP: Preprocesador de Hipertexto"). Fue creado originalmente por Rasmus Lerdorf en 1994 y ha evolucionado desde entonces gracias a una comunidad activa de desarrolladores.

PHP Es un lenguaje de programación interpretado en servidor y embebido entre el código HTML, que actualmente se encuentra en la versión estable PHP 8.3.0, con el objetivo de desarrollar páginas web dinámicas, además de ser fácil de aprender con respecto a otros lenguajes utilizados para el mismo propósito.

Características principales de PHP:

- **Lenguaje de servidor:** PHP es un lenguaje de servidor, lo que significa que se ejecuta en el servidor web y genera contenido dinámico antes de ser enviado al navegador del usuario.
- **Integración con HTML:** PHP está diseñado para integrarse fácilmente con el código HTML, lo que permite a los desarrolladores mezclar lógica de programación y contenido visual en el mismo archivo.

- **Soporte para bases de datos:** PHP tiene una amplia compatibilidad con diferentes sistemas de gestión de bases de datos, lo que facilita la creación de aplicaciones web que interactúan con bases de datos.
- **Código abierto:** PHP es de código abierto y se distribuye bajo la licencia PHP License, lo que significa que cualquiera puede usarlo, modificarlo y distribuirlo libremente.
- **Facilidad de aprendizaje:** PHP tiene una curva de aprendizaje relativamente suave, lo que lo convierte en una opción popular para principiantes en el desarrollo web.
- **Gran comunidad de desarrollo:** La comunidad de desarrolladores de PHP es muy activa y ofrece una amplia gama de recursos, bibliotecas y frameworks que facilitan el desarrollo rápido de aplicaciones web.
- **Plataforma independiente:** PHP es compatible con diferentes sistemas operativos, lo que significa que se puede ejecutar en múltiples plataformas, incluyendo Windows, macOS y varias distribuciones de Linux.

2.9.3 *JavaScript*

JavaScript, es el lenguaje interpretado orientado a objetos desarrollado por Netscape que se utiliza en millones de páginas web y aplicaciones de servidor en todo el mundo. JavaScript de Netscape es un superconjunto del lenguaje de scripts estándar de la edición de ECMA262 3 (ECMAScript) que presenta sólo leves diferencias respecto a la norma publicada.

JavaScript es un lenguaje de programación de alto nivel y de scripting que se utiliza principalmente para crear contenido interactivo en páginas web. Es un lenguaje interpretado, lo que significa que se ejecuta directamente en el navegador web del usuario, lo que permite que las páginas web respondan a las acciones del usuario sin necesidad de recargar la página.

Una de las partes esenciales del desarrollo web moderno es JavaScript y se utiliza en conjunto con HTML y CSS para crear experiencias interactivas y dinámicas en la web. Con el auge de tecnologías como Node.js, JavaScript también se utiliza para desarrollar aplicaciones del lado del servidor, lo que lo convierte en un lenguaje versátil y poderoso que abarca todo el espectro del desarrollo web.

2.9.3.1 AJAX

AJAX (Asynchronous JavaScript and XML) es una técnica de programación que permite enviar y recibir datos de manera asíncrona entre el navegador y el servidor web sin tener que recargar la página completa. Esta técnica utiliza JavaScript para enviar solicitudes al servidor, procesar los datos y actualizar partes específicas de una página web sin interrumpir la experiencia del usuario.

Características principales de AJAX:

- **Asincronía:** Como su nombre indica, AJAX permite realizar operaciones de manera asíncrona, lo que significa que las solicitudes y respuestas no bloquean la ejecución del código en el navegador.
- **Comunicación con el servidor:** AJAX utiliza el objeto XMLHttpRequest (XHR) en JavaScript para enviar y recibir datos del servidor en segundo plano.
- **Actualización parcial de la página:** En lugar de recargar toda la página, AJAX permite actualizar solo partes específicas de la página, lo que puede mejorar significativamente la velocidad y la eficiencia de las interacciones del usuario.
- **Respuestas en diferentes formatos:** Aunque el nombre original menciona XML, las respuestas del servidor pueden estar en diferentes formatos, como JSON (JavaScript

Object Notation), HTML o texto plano, lo que proporciona una mayor flexibilidad en el intercambio de datos.

- **Mejora la experiencia del usuario:** La utilización de AJAX permite crear interfaces de usuario más rápidas y receptivas, lo que mejora la experiencia del usuario al interactuar con aplicaciones web.
- **Bibliotecas y Frameworks:** Existen muchas bibliotecas y frameworks de JavaScript, como jQuery y Axios, que facilitan la implementación de AJAX en aplicaciones web.

2.9.3.2 JSTREE

JSTREE es una popular biblioteca de JavaScript que permite crear fácilmente árboles de jerarquía interactivos y dinámicos en aplicaciones web. Esta biblioteca proporciona una forma sencilla de representar información jerárquica en forma de árbol, lo que es especialmente útil para organizar y visualizar datos complejos o estructuras de datos anidadas.

Características principales de jstree:

- **Representación de árboles:** jstree ofrece una forma eficiente de crear y mostrar árboles jerárquicos en páginas web, lo que permite a los usuarios navegar y explorar la información de manera intuitiva.
- **Interactividad:** Los árboles creados con jstree son interactivos, lo que significa que los nodos se pueden expandir y contraer para mostrar o ocultar información secundaria.
- **Personalización:** jstree es altamente personalizable y permite ajustar la apariencia y el comportamiento del árbol a través de opciones de configuración y estilos CSS.
- **Eventos y acciones:** La biblioteca proporciona una API que permite agregar eventos y acciones a los nodos del árbol, lo que permite interactuar con los datos y ejecutar acciones específicas cuando se hace clic en un nodo.

- **Soporte para diferentes fuentes de datos:** jstree puede cargar datos desde diferentes fuentes, como un objeto JSON, una URL que devuelve datos en formato JSON, o incluso desde una estructura HTML existente.
- **Plug-ins:** jstree tiene una amplia colección de plug-ins disponibles que agregan funcionalidades adicionales al árbol, como arrastrar y soltar, búsqueda, edición en línea y más.

La popularidad de jstree se debe a su facilidad de uso y versatilidad para crear árboles de jerarquía en aplicaciones web. Es ampliamente utilizado en aplicaciones que necesitan representar datos en forma de árbol, como sistemas de archivos, organización de categorías y etiquetas, menús desplegables, y más.

La biblioteca jstree está escrita en JavaScript puro y no tiene dependencias externas, lo que facilita su integración en proyectos web existentes. Además, está bien documentada y cuenta con una comunidad activa que ofrece soporte y desarrollo continuo de nuevas características y mejoras.

2.9.3.3 DataTables

DataTables es una biblioteca de JavaScript que proporciona funcionalidades avanzadas para la visualización y manipulación de tablas de datos en páginas web. Permite convertir tablas HTML simples en tablas interactivas y dinámicas con características como paginación, búsqueda, ordenamiento, filtrado, y más.

Esta herramienta es ampliamente utilizada en el desarrollo web para mejorar la presentación y la interactividad de las tablas de datos en aplicaciones y sitios web. Facilita la manipulación y visualización de grandes cantidades de información de manera eficiente y elegante.

Además, cuenta con una comunidad activa y una documentación completa que hace que sea fácil de implementar y personalizar según las necesidades específicas de cada proyecto.

2.9.4 CSS

CSS (Cascading Style Sheets) es un lenguaje de hojas de estilo utilizado en desarrollo web para controlar la presentación y el diseño de páginas escritas en HTML y XHTML. Permite definir cómo se verá visualmente el contenido de una página, incluyendo aspectos como el tipo y tamaño de fuente, colores, espaciado, posicionamiento de elementos y otros estilos visuales. CSS permite separar la estructura y el contenido del estilo, facilitando la tarea de los diseñadores al modificar la apariencia de un sitio web sin cambiar su contenido. Es compatible con todos los navegadores modernos y es fundamental para crear sitios web atractivos, responsivos y visualmente agradables. Con CSS, los desarrolladores pueden aplicar reglas de estilo a elementos específicos, hacer uso de la herencia y crear animaciones e interacciones sin necesidad de usar JavaScript.

La relación entre HTML y CSS es muy fuerte. Dado que HTML es un lenguaje de marcado (es decir, constituye la base de un sitio) y CSS enfatiza el estilo. En resumen, CSS es una herramienta esencial para mejorar la presentación y el diseño de páginas web.

2.9.5 HTML

HTML (HyperText Markup Language) es un lenguaje de marcado utilizado en desarrollo web para crear y estructurar el contenido de páginas en Internet. Consta de etiquetas y elementos que definen diferentes secciones y elementos de una página, como títulos, párrafos, enlaces, imágenes y formularios. Los navegadores web interpretan el código HTML para mostrar el contenido visualmente, permitiendo la creación de sitios web y la navegación a través de hipervínculos. HTML es una tecnología fundamental para el desarrollo web moderno y trabaja en conjunto con CSS para la presentación visual y JavaScript para la interacción y funcionalidad en

la web. Es esencial para construir sitios web atractivos y funcionales y es compatible con todos los navegadores modernos, lo que garantiza una experiencia consistente para los usuarios en diferentes plataformas. La versión más reciente es HTML5, que ha introducido nuevas características y mejoras para adaptarse a las necesidades cambiantes de la web. En resumen, HTML es el lenguaje que permite la creación de contenido en la web y es esencial para el desarrollo de sitios y aplicaciones en línea.

Características principales de HTML:

- **Estructura de páginas:** HTML proporciona una estructura básica para organizar el contenido de una página web. Utiliza etiquetas y elementos para definir diferentes secciones como encabezados, párrafos, listas, imágenes, enlaces y más.
- **Hipervínculos:** Permite crear enlaces (hipervínculos) que conectan diferentes páginas web, lo que permite la navegación entre ellas y la creación de una red de información en la web.
- **Contenido multimedia:** HTML permite insertar y mostrar diversos tipos de contenido multimedia, como imágenes, audio y video, en una página web.
- **Formularios:** HTML proporciona elementos de formulario que permiten a los usuarios ingresar datos y enviar información al servidor, como en formularios de contacto o registros.
- **Compatibilidad con navegadores:** HTML es compatible con todos los navegadores modernos, lo que garantiza que el contenido se visualice correctamente en diferentes plataformas.

- **Responsive Web Design:** HTML es esencial para implementar Responsive Web Design, lo que permite que una página web se adapte automáticamente a diferentes tamaños de pantalla y dispositivos.
- **Versión:** HTML ha pasado por diferentes versiones, siendo HTML5 la versión más reciente, que ha introducido muchas características nuevas y mejoras para el desarrollo web moderno.

HTML es la columna vertebral del contenido en la web y es esencial para la creación de sitios web. Junto con CSS para la presentación y JavaScript para la interacción, HTML forma el núcleo de la mayoría de los sitios web modernos y aplicaciones en línea.

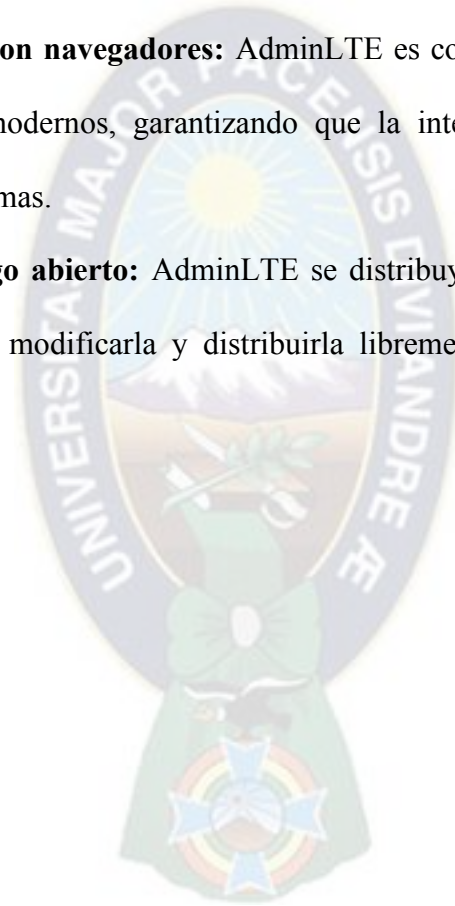
2.9.6 *AdminLTE*

AdminLTE es una plantilla de interfaz de usuario (UI) de código abierto y gratuita, basada en HTML, CSS y JavaScript. Está diseñada para crear rápidamente paneles de administración y sistemas de administración para aplicaciones web y sitios web. AdminLTE ofrece un diseño limpio, moderno y receptivo que es fácil de personalizar y adaptar a diferentes proyectos.

Características principales de AdminLTE:

- **Diseño receptivo:** AdminLTE es completamente receptivo, lo que significa que se ajusta automáticamente a diferentes tamaños de pantalla, como computadoras de escritorio, tabletas y dispositivos móviles.
- **Componentes predefinidos:** La plantilla incluye una amplia variedad de componentes y elementos de IU predefinidos, como tablas, gráficos, formularios, botones, barras de navegación, paneles y más, lo que facilita la creación de interfaces de usuario completas y atractivas.

- **Integración de plugins:** AdminLTE es compatible con varios plugins populares de JavaScript y CSS, lo que permite agregar funcionalidades adicionales como calendarios, gráficos interactivos, tablas avanzadas y más.
- **Personalización sencilla:** AdminLTE utiliza una estructura modular y está bien documentado, lo que facilita la personalización y adaptación de estilos y funcionalidades para satisfacer las necesidades específicas de cada proyecto.
- **Compatibilidad con navegadores:** AdminLTE es compatible con una amplia gama de navegadores modernos, garantizando que la interfaz se vea correctamente en diferentes plataformas.
- **Licencia de código abierto:** AdminLTE se distribuye bajo la licencia MIT, lo que permite utilizarla, modificarla y distribuirla libremente en proyectos personales y comerciales.



3 Estado del arte

El estado del arte, también conocido como estado de la cuestión o estado de conocimiento, representa el nivel actual de avance, conocimiento o desarrollo en un campo específico. En la investigación, implica la revisión y análisis exhaustivo de la literatura académica, publicaciones científicas y otras fuentes relevantes para identificar los avances recientes, tendencias y lagunas en el área de estudio. El estado del arte proporciona un contexto sólido para situar la nueva investigación y demostrar el conocimiento previo sobre el tema, lo que es fundamental para el desarrollo de proyectos y la contribución al conocimiento existente.

La contextualización, la clasificación y la categorización son términos importantes en la investigación para el Estado del Arte. A continuación, se proporciona estos aspectos importantes:

3.1 Contextualización

El primer software de la historia fue creado en la década de 1940. Uno de los ejemplos más destacados es el "ENIAC" (Electronic Numerical Integrator and Computer), que fue una de las primeras computadoras electrónicas de propósito general. Fue desarrollada durante la Segunda Guerra Mundial y se completó en 1945. ENIAC fue diseñada para realizar cálculos complejos y fue programada mediante el uso de cables y tableros de conexiones físicas.

Otro ejemplo relevante es el "Colossus", una máquina diseñada y construida en Gran Bretaña durante la misma época para descifrar códigos secretos alemanes durante la guerra. Colossus es considerada la primera computadora electrónica digital programable.

Es importante destacar que, en esa época, los programas se escribían en lenguaje de máquina, lo que implicaba trabajar directamente con el código binario que la computadora

entendía. No existían lenguajes de programación de alto nivel como los que se utilizan en la actualidad.

Desde entonces, el desarrollo del software ha avanzado significativamente y ha revolucionado la forma en que interactuamos con las computadoras y el mundo digital en general. Actualmente, el software es una parte esencial de la vida moderna y ha impulsado el progreso en diversas áreas, desde la investigación científica hasta la administración empresarial y la comunicación global.

3.2 Clasificación

Para una mejor comprensión de las áreas de enfoque y desarrollo de software para sistemas de calzado se clasifican 5 proyectos similares.

3.2.1 Diseño de un sistema de gestión de inventario de materia prima de la empresa fábrica de calzado ECCO de Nicaragua, S.A. (Hurtado, Molina, & Vega, 2018).

El proyecto de grado se enfoca en el control de inventario de materia prima en una empresa, considerando su importancia para la eficacia y la obtención de beneficios. Se analizó el proceso de inventario mediante la familiarización con la división del almacén, el clima organizacional y entrevistas al personal involucrado. Se utilizó Microsoft Excel para obtener información digital del inventario y se aplicó el principio de Pareto para clasificar los productos más representativos monetariamente.

Se realizaron pronósticos a corto plazo para los productos más representativos mediante el estudio de costos ABC. Se utilizó CurveExpert Professional y Microsoft Excel para el procesamiento de datos. Se determinó el modelo adecuado para las particularidades de los artículos de la empresa, considerando la demanda, los costos y el tiempo entre pedidos. Los resultados

mostraron una cantidad óptima a ordenar, tiempo de reorden y un ahorro significativo en costos al utilizar el modelo propuesto en comparación con el modelo actual.

Las recomendaciones se basaron en los resultados obtenidos y la información de la empresa. Los pronósticos de la demanda fueron útiles para elaborar el modelo matemático de control de inventario, permitiendo ajustarlo de manera más precisa a la realidad del mercado de los productos. En general, el proyecto se centró en mejorar la gestión de inventario de materia prima para optimizar costos y eficiencia en la empresa.

3.2.2 *Diseño de un sistema de ventas en Calzados Gorillaz (Guerra, 2022).*

El objetivo principal de este proyecto es diseñar un sistema de ventas para la empresa Gorillaz, ubicada en el Jirón de la Unión en el Cercado de Lima. La empresa se dedica a la venta de calzados para damas, niños y niñas. El sistema tiene como finalidad centralizar y organizar la información de ventas y compras, ya que actualmente estos procesos son manuales y presentan un control inadecuado de la información.

Para desarrollar el sistema, se recopiló información detallada sobre el funcionamiento del área de ventas y compras de la tienda, analizando los flujos de estos procesos para identificar áreas de mejora. Se utilizó la metodología RUP (Proceso Unificado Relacional) y UML (Lenguaje Unificado para Modelado) en el desarrollo del sistema.

El proyecto concluye que el uso adecuado de herramientas como el lenguaje de programación C#, ASP.NET, CSS y la base de datos Microsoft SQL Server permite crear un software confiable y seguro, mejorando el control y manejo de los procesos en la tienda de calzados Gorillaz.

Como resultado, se logró desarrollar una solución automatizada que cambia el método de trabajo en el área de ventas y mejora la función del área de compras de la tienda de calzados Gorillaz.

3.2.3 Desarrollo de un sistema de gestión de inventarios para el control de materias primas y productos terminados (Gayón & Ospina, 2019).

El proyecto se centra en la empresa FIDENCI Y COMPAÑÍA LTDA, dedicada a la fabricación de calzado. La gerencia general reconoce la importancia de asegurar la fidelidad de los clientes y alcanzar un crecimiento sostenible, por lo que solicitaron una propuesta para mejorar los procesos de planeación, control y seguimiento de inventarios debido a las falencias presentadas en este aspecto.

Para desarrollar la propuesta, se realizó una revisión bibliográfica sobre sistemas de inventarios, manejo de almacenes, pronósticos y políticas de reabastecimiento, entre otros temas. Luego, se indagó en los problemas críticos de la empresa a través de encuestas y observación, identificando que el manejo inadecuado de las compras y reposiciones de materiales agotados afectaba la disponibilidad de productos y la rotación de productos de baja demanda, generando una inversión alta en productos almacenados.

Se propusieron acciones para mejorar el control de requerimientos de materia prima y productos terminados, incluyendo el diseño del área de almacenamiento, la implementación de procedimientos y formatos estandarizados, y el desarrollo del modelo de inventario más adecuado basado en las 40 referencias más importantes según su utilización.

Los resultados del proyecto se basan en una evaluación de costos positiva, una simulación que muestra un nivel de servicio óptimo sin importar la demanda y mejoras mensuales en los

indicadores de gestión gracias a las acciones implementadas. En general, la propuesta ha permitido mejorar la gestión de inventarios en la empresa, lo que contribuye a la fidelización de clientes y un crecimiento sostenible.

3.2.4 Gestión, control de materia prima, venta de productos e inventarios usando Supply Chain Management (Vasquez, 2020).

Este proyecto implica el desarrollo de un sistema integral que abarque todas las etapas de la producción de calzado en una fábrica. Incluye módulos para la gestión de pedidos de clientes, programación de la producción, control de la cadena de suministro, gestión de inventarios de materias primas y productos terminados, y seguimiento de las etapas de producción.

La empresa INGENIERÍA Y TECNOLOGÍA BOLIVIANA INTEB SRL., con sede en El Alto, ofrece servicios de diseño, construcción y mantenimiento de equipos industriales a nivel nacional. Conforme crece, la cantidad de información en sus distintos departamentos aumenta, por lo que necesitan un Sistema Web para el control de materia prima, venta de productos e inventarios. Para el desarrollo de este sistema, utilizarán herramientas como C# con .Net Core 2.2 para el Back-End, SQL Server para la Base de Datos y Vue.js para el Front-End. El modelado se realizará utilizando diagramas UML aplicados en una variación de la ingeniería de software llamada UWE.

3.2.5 Sistema web de administración y control del almacenamiento de artículos en el almacén central y en subalmacenes (Espejo, 2020).

El proyecto se enfoca en desarrollar un sistema de información y administración para gestionar los artículos de un almacén central y subalmacenes de manera eficiente. El objetivo es agilizar los movimientos de los artículos y mantener una información actualizada en el Gobierno Autónomo Municipal de El Alto.

Para el desarrollo del proyecto, se utilizó la metodología SCRUM, que divide el proceso en etapas llamadas Sprints basadas en historias de usuario recopiladas a través de entrevistas con el personal del almacén central. También se empleó la metodología UML para el modelado y la metodología UWE para el desarrollo web, lo que proporcionó un modelado ordenado y preciso.

El aspecto de calidad del software se basó en los factores de calidad del ISO 9126, y para estimar el costo del sistema se aplicó el modelo de costos COCOMO II.

En resumen, el proyecto tiene como objetivo desarrollar un sistema web que permita administrar y controlar el almacenamiento de artículos en el almacén central y los subalmacenes, optimizando los procesos y facilitando la gestión de inventarios para el Gobierno Autónomo Municipal de El Alto.

3.3 Categorización

Con lo descrito anteriormente podemos decir que un Sistema de Control de Producción y Administrativo de Calzado debería tener una serie de características y funcionalidades clave para asegurar una gestión eficiente y efectiva de todos los aspectos relacionados con la producción y administración de calzado. A continuación, se presenta algunas de las características que se consideró importantes y que debería incluir:

- **Gestión de Inventarios:** Un módulo para administrar el inventario de materias primas, componentes y productos terminados. Debe permitir registrar las entradas y salidas de inventario, así como realizar seguimiento de las existencias disponibles.
- **Gestión de Producción:** Capacidad para planificar y gestionar las órdenes de producción. Esto incluye el registro de pedidos, programación de la producción, seguimiento del progreso y la capacidad de generar reportes de producción.

- **Personalización de Productos:** Si el negocio ofrece productos personalizados, el sistema debe permitir la configuración de opciones de personalización para los clientes y asegurar que las especificaciones se incluyan en el proceso de producción.
- **Gestión de Proveedores:** Un módulo para administrar la información de los proveedores de materias primas y componentes, incluyendo datos de contacto, historial de compras y seguimiento de pagos.
- **Ventas y Facturación:** Un módulo de ventas y facturación que permita registrar pedidos, generar cotizaciones y facturas, y llevar un seguimiento de los pagos realizados.
- **Gestión de Clientes:** Un sistema para gestionar la información de los clientes, incluyendo datos de contacto, historial de compras y preferencias.
- **Reportes y Análisis:** Capacidad para generar reportes y análisis sobre el rendimiento de la producción, las ventas, el inventario y otros aspectos relevantes del negocio.
- **Seguridad y Permisos de Acceso:** Un sistema seguro que limite el acceso a la información según los roles y permisos de los usuarios, asegurando que la información sensible esté protegida.
- **Escalabilidad:** El sistema debe tener la capacidad de crecer y adaptarse a medida que el negocio se expande y aumenta su producción y operaciones.

Cada empresa de calzado puede tener necesidades específicas, por lo que es importante personalizar el sistema según los requerimientos particulares del negocio. Un sistema de control de producción y administración de calzado bien diseñado y adaptado puede ayudar a mejorar la eficiencia, reducir costos y garantizar una gestión más efectiva del negocio.

4 Objetivos y metodología

4.1 Objetivos

4.1.1 *Objetivo general*

Implementar un Sistema de Control de Producción y Administración de Calzado que contribuya al manejo de la información estratégica de la fábrica de calzado Omar Alex, de manera que facilite la administración y seguimiento de los productos.

4.1.2 *Objetivos específicos*

- Favorecer el control del sistema de calzado a tal punto que una persona sin conocimientos de programación sea capacitada para la administración del sistema.
- Optimizar el proceso de obtención de información de manera rápida y flexible de todas las actividades realizadas.
- Desarrollar un módulo que registre todos los pedidos y pueda generar consolidaciones de pedidos.
- Implementar un módulo de registro de pedidos de calzado que contenga toda la información del detalle del pedido.
- Desarrollar módulos que permitan dar seguimiento a la elaboración de cada pedido realizado por el cliente.
- Brindar compatibilidad con dispositivos móviles y diferentes plataformas de dispositivos utilizando Diseño Web Adaptable (Responsive Web Design).

4.2 Metodología

El análisis y diseño de sistemas según Kendall es una metodología clásica que se utiliza para el desarrollo de sistemas de información. Esta metodología fue propuesta por Kenneth Kendall y Julie Kendall y se ha utilizado ampliamente en el campo de la ingeniería de software y la gestión de proyectos de TI (Kendall & Kendall, 2011).

El enfoque de análisis y diseño de sistemas Kendall se basa en un enfoque estructurado y secuencial para el desarrollo de sistemas. Se divide en varias etapas, que se describen a continuación:

- **Investigación Preliminar:** En esta etapa, se identifican y definen los problemas y oportunidades que pueden resolverse con un nuevo sistema de información. Se realizan entrevistas y análisis de las necesidades de los usuarios y se establecen los objetivos del sistema.
- **Estudio de Factibilidad:** Se lleva a cabo un análisis de factibilidad técnica, económica, operativa, legal y de planificación para determinar si el desarrollo del sistema es viable y justificado.
- **Análisis de Requisitos:** En esta etapa, se recopilan y documentan los requisitos del sistema, incluyendo las funciones que debe realizar, las restricciones y las expectativas de los usuarios.
- **Diseño del Sistema:** Se crea un diseño detallado del sistema, que incluye la arquitectura, la estructura de datos, el diseño de la interfaz de usuario y los algoritmos necesarios para llevar a cabo las funciones requeridas.
- **Implementación:** Se realiza la codificación del sistema de acuerdo con el diseño realizado en la etapa anterior.
- **Pruebas:** Se llevan a cabo pruebas exhaustivas para asegurar que el sistema funcione correctamente y cumpla con los requisitos definidos.
- **Instalación y Puesta en Marcha:** Se implementa el sistema en el entorno de producción y se lleva a cabo una transición suave del sistema anterior al nuevo sistema.

- **Mantenimiento:** Después de la puesta en marcha, se realizan ajustes y correcciones para mantener y mejorar el sistema a lo largo del tiempo.

El enfoque de análisis y diseño de sistemas Kendall se basa en una metodología secuencial, lo que significa que cada etapa debe completarse antes de pasar a la siguiente. Sin embargo, es importante tener en cuenta que, en la práctica, el desarrollo de sistemas a menudo puede ser iterativo y adaptarse a los cambios y requisitos a lo largo del proceso.

La elección de utilizar la metodología de análisis y diseño de sistemas de Kendall dependerá de la naturaleza del proyecto. Cada metodología tiene sus ventajas y desventajas, y se decidió utilizar esta metodología.

4.2.1 Descripción gráfica de los sistemas

Es posible representar visualmente un sistema o subsistema dentro de una organización corporativa de diversas maneras. Mediante diferentes modelos gráficos, podemos delinear los límites del sistema y la información que utiliza.

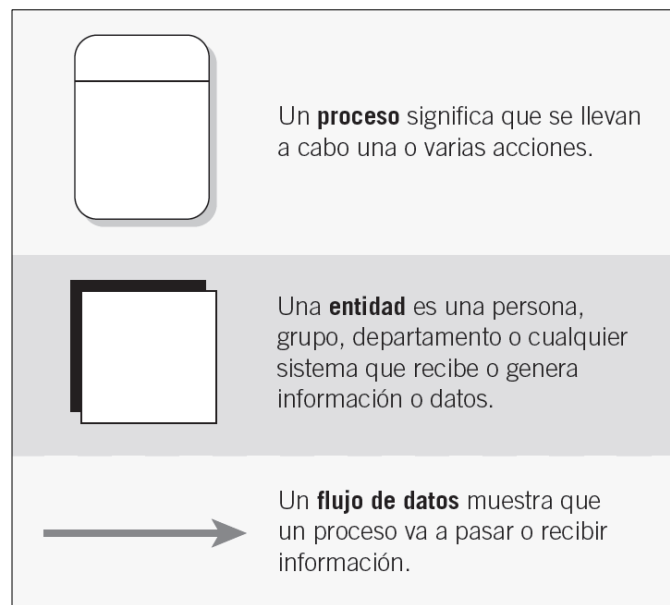
4.2.1.1 Los sistemas y el diagrama de flujo de datos a nivel de contexto

El primer modelo, conocido como el diagrama de flujo de datos a nivel de contexto o modelo ambiental, se enfoca en representar visualmente el flujo de datos que ingresan y salen del sistema, así como el procesamiento de estos datos. En este modelo, se utilizan tres símbolos principales: un rectángulo con esquinas redondas, un cuadrado con dos bordes sombreados y una flecha. Los procesos transforman los datos entrantes en información de salida, y el nivel de contenido tiene solo un proceso, que representa todo el sistema completo. Las entidades externas, que pueden ser personas, grupos, departamentos corporativos u otros sistemas, se conectan mediante líneas de flujo de datos. El diagrama de flujo de datos a nivel de contexto define los

límites del sistema, dejando fuera a las entidades externas, sobre las cuales el sistema no tiene control. Aunque este modelo no muestra detalles precisos, sirve como punto de partida para desarrollar otros diagramas más detallados y complejos en el análisis de sistemas.

Figura 2

Los símbolos básicos de un diagrama de flujo de datos.



Nota. Representa los símbolos básicos que se usa en los diagramas de flujo de datos a nivel de contexto. Tomada de (Kendall & Kendall, 2011).

4.2.2 Modelo de casos de uso

El modelo de caso de uso, que inicialmente fue presentado como un diagrama en UML orientado a objetos, ha evolucionado y se utiliza actualmente independientemente de la metodología de desarrollo de sistemas. Puede ser empleado tanto en el SDLC (Ciclo de Vida del Desarrollo de Sistemas) como en el modelado ágil. Este modelo describe las funcionalidades que realiza un sistema sin entrar en detalles de cómo lo hace, siendo un enfoque lógico del sistema desde la perspectiva de un usuario externo (por ejemplo, los requerimientos del sistema).

El desarrollo de casos de uso implica una colaboración entre el analista y los expertos de negocios para definir los requerimientos del sistema. Es una herramienta efectiva para comunicarse entre el equipo de negocios y el equipo de desarrollo. Los casos de uso representan las distintas interacciones o comportamientos que son significativos para los usuarios del sistema, dividiendo la funcionalidad del sistema en servicios, respuestas y comportamientos relevantes.

Desde la perspectiva del usuario, un caso de uso debe producir un valor significativo. Por lo tanto, el analista debe identificar qué es importante para el usuario e incluirlo en el diagrama del caso de uso. Por ejemplo, la introducción de una contraseña puede ser considerada valiosa si es esencial para la seguridad o el éxito del proyecto y preocupa al usuario. El modelo de caso de uso se enfoca en los resultados que brindan valor al usuario final del sistema.

4.2.2.1 Símbolos de los casos de uso

Un diagrama de caso de uso está compuesto por símbolos de actores, casos de uso y líneas conectoras. Los actores representan roles específicos de los usuarios del sistema y pueden ser personas, sistemas externos o dispositivos. Estos actores interactúan con el sistema desde fuera de él y pueden iniciar uno o varios casos de uso. Los casos de uso, por otro lado, representan las funcionalidades o interacciones del sistema que producen un valor significativo para los actores.

Los actores pueden agruparse en categorías, como actores principales y actores de soporte. Los actores principales son aquellos que interactúan directamente con el sistema y pueden proporcionar información detallada sobre lo que se espera que haga cada caso de uso. También pueden dar prioridades y objetivos relacionados con el sistema. Por otro lado, los actores de soporte, también conocidos como actores secundarios, mantienen el sistema en funcionamiento o proveen otros servicios para el sistema, como el personal de soporte técnico, analistas o programadores.

Es importante representar a los actores según sus roles específicos, incluso si se trata de la misma persona en el mundo real, pero interactúa con el sistema en diferentes roles. Los actores y casos de uso pueden interactuar de manera compleja, involucrando múltiples actores en un caso de uso o permitiendo que un actor interactúe con varios casos de uso. En general, el diagrama de caso de uso es una herramienta visual que permite identificar de manera clara y concisa las interacciones entre los usuarios y el sistema, lo que facilita la comprensión de los requerimientos y objetivos del sistema.

4.2.2.2 Relaciones de los casos de uso

Las relaciones activas, también conocidas como relaciones de comportamiento, son

Figura 3

Símbolos de los casos de uso.

Relación	Símbolo	Significado
Comunica	—————	Para conectar un actor con un caso de uso se utiliza una línea sin puntas de flecha.
Incluye	<< Incluye >> ←-----	Un caso de uso contiene un comportamiento común para más de un caso de uso. La flecha apunta al caso de uso común.
Extiende	<< Extiende >> ----->	Un caso de uso distinto maneja las excepciones del caso de uso básico. La flecha apunta del caso de uso extendido al básico.
Generaliza	—————▷	Una “cosa” de UML es más general que otra “cosa”. La flecha apunta a la “cosa” general.

Nota. Descripción y significado de los símbolos de casos de uso. Tomada de (Kendall & Kendall, 2011).

utilizadas principalmente en los diagramas de casos de uso. Existen cuatro tipos básicos de relaciones de comportamiento: comunica, incluye, extiende y generaliza. Todos estos términos son

verbos y se representan con flechas y líneas en el diagrama de casos de uso. A continuación, describiremos cada una de estas cuatro relaciones.

4.2.2.3 Niveles de los casos de uso

El método propuesto por Alistair Cockburn para crear casos de uso utiliza metáforas de altitud para clasificarlos en distintos niveles:

1. Blanco: Representa el nivel más alto, el nivel empresarial. En este nivel, se definen casos de uso generales para toda la organización, y su cantidad es limitada (alrededor de cuatro o cinco). Ejemplos podrían ser "Anunciar los productos", "Vender los productos a los clientes", "Administrar el inventario", "Administrar la cadena de suministro" y "Optimizar los embarques".
2. Cometa: Es un nivel inferior al blanco, pero aún de alto nivel, que ofrece una visión general. Los casos de uso del cometa pueden estar enfocados en una unidad de negocio o departamento específico y resumen los objetivos de dicho nivel. Ejemplos incluyen "Registrar estudiantes" o "Hacer reservaciones de aerolínea, hotel, automóvil o crucero" en el caso de una agencia de viajes.
3. Azul: Este nivel se encuentra a nivel del mar y se centra en los objetivos de los usuarios. Los casos de uso a nivel azul son de mayor interés para los usuarios y son más fáciles de comprender desde una perspectiva de negocios. Generalmente, se escriben para actividades de negocios específicas, y cada persona debe poder realizar una actividad de nivel azul en un lapso de 2 a 20 minutos. Ejemplos incluyen "Registrar un estudiante que desea continuar", "Agregar un nuevo cliente", "Colocar un artículo en un carrito de compras" y "Realizar el pago".

4. Índigo o pez: Representa un caso de uso que muestra muchos detalles, a menudo a un nivel funcional o subfuncional. Aquí se incluyen actividades más específicas y detalladas. Ejemplos podrían ser "Elegir una clase", "Pagar las cuotas académicas", "Buscar el código de aeropuerto para una ciudad" y "Producir una lista de clientes después de introducir un nombre".
5. Negro o almeja: Representa los casos de uso más detallados, a un nivel de subfunción. Aquí se incluyen actividades muy específicas y detalladas, generalmente enfocadas en tareas técnicas. Ejemplos podrían ser "Validar un inicio de sesión seguro", "Agregar un nuevo campo mediante HTML dinámico" o "Usar Ajax para realizar una pequeña parte de una página Web".

En el escenario de caso de uso, se incluyen áreas opcionales, y las principales son: el encabezado del área con identificadores e iniciadores del caso, los pasos realizados y el área del pie de página con precondiciones, suposiciones, preguntas y otra información relevante.

El área inicial del caso de uso proporciona una orientación al lector al contener el nombre y un ID único del caso de uso, el sistema o área de aplicación al que pertenece, los actores involucrados y los interesados con alto nivel de interés. Algunos interesados, como accionistas o directores, no interactúan directamente con el sistema. Los actores principales son considerados interesados, pero no se enumeran nuevamente. Además, se incluye el nivel del caso de uso (azul, cometa, etc.) y una breve descripción de los objetivos logrados por el caso de uso.

Figura 4

Escenario de caso de uso.

Nombre del caso de uso:	Registrarse para la conferencia	ID Único: Conf RG 003
Área:	Planeación de la conferencia	
Actor(es):	Participante	
Interesados:	Patrocinador de la conferencia, oradores de la conferencia	
Nivel:	Azul	
Descripción:	Permitir que el participante de la conferencia se registre en línea mediante un sitio Web seguro.	
Evento desencadenador:	El participante utiliza el sitio Web de registro para la conferencia, introduce su ID de usuario y su contraseña, y hace clic en el botón de inicio de sesión.	
Tipo de desencadenador:	<input type="checkbox"/> Externo <input checked="" type="checkbox"/> Temporal	
Pasos realizados (ruta principal)	Información para los pasos	
1. El participante inicia sesión mediante el servidor Web seguro.	ID de usuario, contraseña	
2. Se lee el registro del participante y se verifica su contraseña.	Registro del participante, ID de usuario, contraseña	
3. Se muestra la información del participante y la sesión en la página Web de registro.	Registro del participante, registro de la sesión	
4. El participante introduce su información en el formulario Web de registro y hace clic en el botón Enviar .	Formulario Web de registro	
5. Se valida la información de registro en el servidor Web.	Formulario Web de registro	
6. Se muestra la página Confirmación de registro para confirmar la información de registro.	Página Web de confirmación	
7. Se hace un cargo a la tarjeta de crédito equivalente a las cuotas de registro.	Página Web segura para tarjeta de crédito	
8. Se escribe el registro en el Diario de agregar registros.	Página Web de confirmación	
9. Se actualiza el registro en el Archivo maestro de registros.	Página Web de confirmación, registro del proceso de registro	
10. Se actualiza el registro de la sesión para cada sesión seleccionada en el Archivo maestro de sesiones.	Página Web de confirmación, registro de la sesión	
11. Se actualiza el registro para el participante en el Archivo maestro de participantes.	Página Web de confirmación, registro de participantes	
12. Se envía la página Web de Confirmación de registro exitoso al participante.	Número de confirmación del registro en el proceso de registro	
Precondiciones:	El participante ya se registró y creó una cuenta de usuario.	
Postcondiciones:	El participante se registró con éxito para la conferencia.	
Suposiciones:	El participante tiene un navegador Web, además de un ID de usuario y contraseña válidos.	
Garantía de éxito:	El participante se registró para la conferencia y está inscrito en todas las sesiones seleccionadas.	
Garantía mínima:	El participante pudo iniciar sesión.	
Requerimientos cumplidos:	Permitir que los participantes de la conferencia se registren mediante un sitio Web seguro.	
Cuestiones pendientes:	¿Cómo se debe manejar una tarjeta de crédito rechazada?	
Prioridad:	Alta	
Riesgo:	Medio	

Nota. Se muestra un escenario de caso de uso que se divide en tres secciones: identificación e iniciación; pasos realizados; y condiciones, suposiciones y preguntas. Tomada de (Kendall & Kendall, 2011).

El encabezado del caso de uso concluye con el evento iniciador, es decir, lo que provoca que el caso de uso comience, junto con el tipo de desencadenador, ya sea externo o temporal. Los eventos externos son aquellos que inician un actor, como una persona o un sistema, solicitando información al sistema, como cuando un sistema de reservaciones de aerolíneas solicita detalles sobre vuelos. Los eventos temporales son aquellos que se activan en momentos específicos, como enviar correos electrónicos con ofertas especiales una vez a la semana, los domingos por la tarde.

La segunda área del caso de uso contiene los pasos realizados y la información necesaria para cada uno de ellos. Estos pasos representan el flujo estándar de eventos y describen las acciones necesarias para completar el caso de uso con éxito. Es recomendable escribir un caso de uso para la ruta principal y luego crear casos de uso separados para las rutas alternativas, en lugar de utilizar instrucciones IF...THEN. Los pasos se enumeran con números enteros y pueden surgir de entrevistas con los usuarios o de historias de modelado ágil. Es esencial revisar estos pasos con los usuarios para garantizar la claridad del proceso.

El analista debe revisar detalladamente cada paso del caso de uso y determinar la información necesaria para llevarlos a cabo. Si hay información faltante, se debe programar una entrevista de seguimiento con los usuarios. Algunas descripciones de casos de uso incluyen extensiones o escenarios alternativos, que se enumeran como 3.1, 3.2, 3.3, etc., siguiendo el flujo estándar de eventos. Los analistas y usuarios pueden realizar lluvias de ideas para identificar posibles fallos en la ruta principal y descubrir detalles importantes. Trabajar con los usuarios para definir cómo actuar en diferentes condiciones ayuda a detectar errores en las etapas iniciales del ciclo de vida del proyecto.

El área de Precondiciones en el pie de página del escenario del caso de uso muestra una lista de condiciones que deben cumplirse antes de que el participante pueda realizar cierta acción,

como registrarse para una conferencia. En el ejemplo dado, el participante debe haber iniciado sesión previamente como miembro de la sociedad y contar con un ID de usuario y contraseña válidos.

El área de Postcondiciones muestra una lista de lo que se logra al completar el caso de uso. En este caso, se indicaría lo que se logra al registrarse para la conferencia.

El área de Suposiciones presenta una lista de premisas básicas que el analista asume que se han cumplido previamente por parte del actor involucrado en el caso de uso.

El área de Requerimientos cumplidos muestra por qué este caso de uso es importante y necesario para el éxito del área de negocios.

La prioridad es una indicación de la importancia relativa del caso de uso y su necesidad de desarrollo. Algunos casos de uso pueden tener una mayor prioridad que otros y deben ser desarrollados primero.

El riesgo es una evaluación aproximada de la probabilidad de que surjan problemas o dificultades durante el desarrollo del caso de uso. En el ejemplo mencionado, el riesgo es calificado como medio debido a que el caso de uso de registro requiere un servidor seguro y acepta información de tarjetas de crédito, lo que podría presentar desafíos en términos de seguridad y cumplimiento.

5 Marco aplicativo

Al establecer un sistema se debe tener la certeza que cumpla dos objetivos; que sea un sistema correcto en el análisis y diseño, y que este correctamente implementado y funcionando, siendo útil y mejorando así la toma de decisiones de la empresa. La función del análisis y diseño es desarrollar un producto que pueda cumplir con los requerimientos establecidos por el usuario y no presente fallas posteriores en su uso.

En el presente capítulo se describe y desarrolla el análisis y diseño del "Sistema de Control de Producción y Administrativo de Calzado. Zapatillas Omar Alex".

5.1 Planificación del Proyecto

En esta fase identificaremos y estableceremos la obtención de los requerimientos, es decir, obtener y relevar la mayor cantidad de información sobre el dominio del sistema. En esta fase se lleva a cabo la definición, análisis y validación de los requisitos a partir de la información recolectada y facilitada por el usuario. Esta primera fase contempla cuatro subfases.

5.1.1 Ejecución de Entrevistas

En esta subfase se aplica el modelo de entrevistas planteado por el método. En la tabla 1 se puede ver un resumen en un modelo final después de aplicar el modelo de entrevistas y cuestionario que se detalla en el anexo. Este modelo está organizado en tres partes donde se indica de manera breve y concisa; el propósito del sistema, la información que se va a publicar en el sistema y las funciones que brindará el sistema.

Tabla 2

Modelo final de cuestionario y entrevista.

Especificación de propósito y bondades del sistema.
Propósito general del sistema.
¿Cuál es el propósito general del sistema? Manejo de la información estratégica para el control de producción, la administración y seguimiento de los productos para evitar pérdidas económicas en la fábrica de calzado Omar Alex.
Información visible en el sistema.
¿Qué información se verá en el sistema? <ul style="list-style-type: none">- Estado y seguimiento de producción de pedidos.- Líneas y estilos de calzado con mayor producción.- Tiempos de entrega de calzado a clientes.
Funciones del sistema.
¿Qué tipo de funciones brinda el sistema? <ul style="list-style-type: none">- El usuario podrá dar de alta los pedidos con la numeración correspondiente del calzado asignando una fecha de entrega.- Se podrá realizar cotizaciones de pedidos de calzado realizados por el cliente.- En caso de no contar con una nueva línea, estilo de calzado u alguna otra característica, el usuario podrá crear una nueva línea o estilo requerido por el cliente.- Contará con módulos de consolidado de forro y corte (material de calzado).- Se tendrá un módulo para el pedido de materiales para distintos proveedores.- Se contará con un módulo de notas de venta.- El administrador del sistema podrá contar con el permiso para registrar nuevos usuarios, clientes, proveedores.- El administrador podrá asignar roles y perfiles a cada usuario.

Nota: Modelo de entrevista que detalla de manera breve y concisa toda la información y las funciones que brindará el sistema. Elaboración propia.

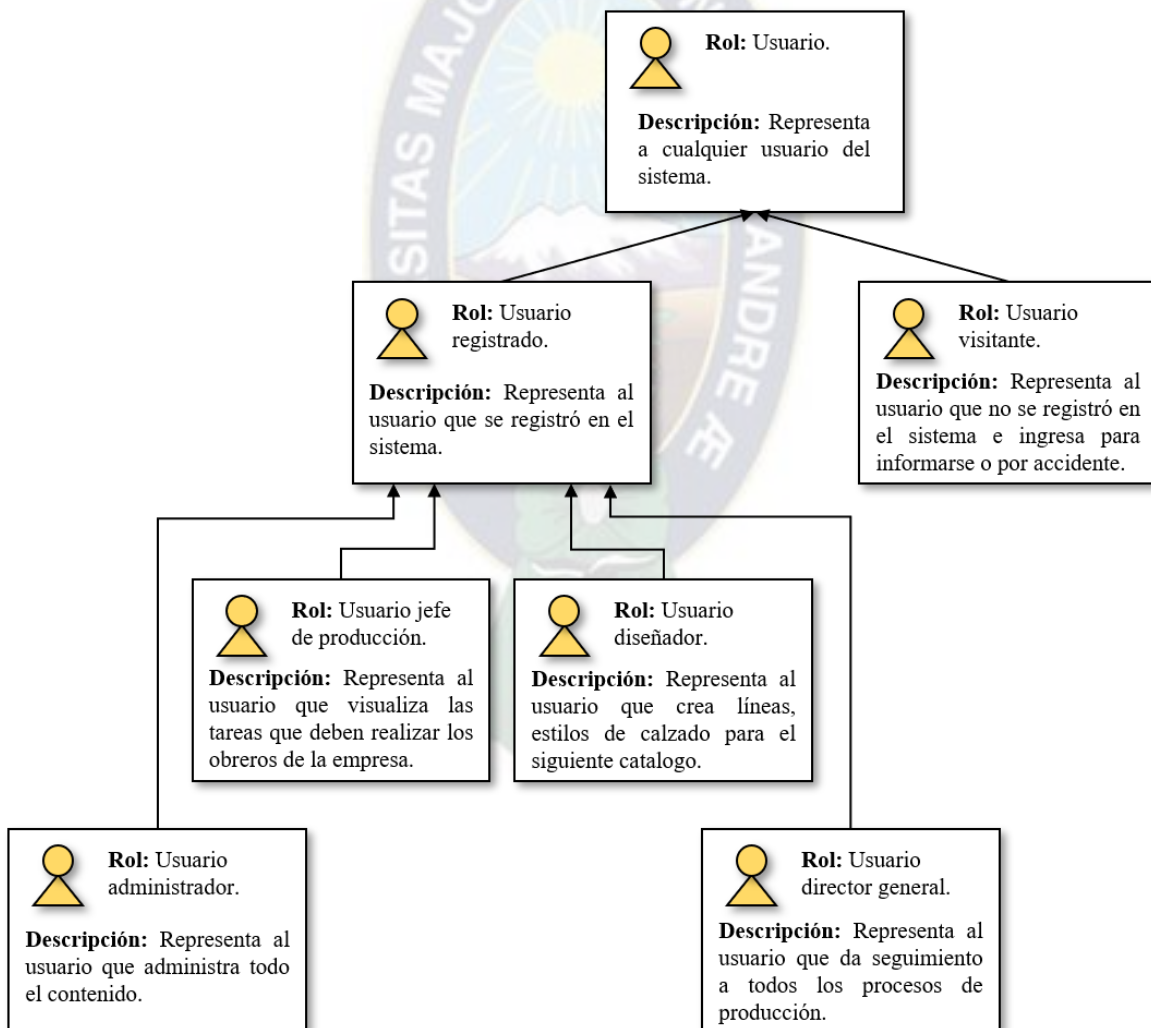
5.1.2 Especificación de roles y tareas

5.1.2.1 Especificación de roles y perfiles.

Luego de obtener la información del dominio del sistema y aplicando la especificación que plantea el método refinado, se identifican los diferentes tipos o clases de usuarios que intervendrán en el manejo y uso del sistema. Son tres los usuarios que se han identificado (ver figura 6):

Figura 5

Identificación de roles.



Nota. Roles y perfiles para los usuarios que intervienen en el sistema. Elaboración propia.

- **Usuario administrador.** Podrá dar de alta nuevos pedidos, realizar cotizaciones, administrar los contenidos del sistema en general.
- **Usuario director general.** Da seguimiento a los pedidos, etapas de producción, contenidos de información en general para la toma de decisiones.
- **Usuario jefe de producción.** Con privilegios reducidos y solo podrá ver y analizar las tareas que deben realizar los obreros de la empresa.
- **Usuario diseñador.** Con privilegios reducidos y podrá crear nuevas líneas, estilos de calzado y generar un catálogo por temporada.

5.1.2.2 Especificación de tareas

Teniendo identificado los roles y perfiles de usuarios que intervendrán en el sistema, se describe las tareas que realiza cada usuario dentro del sistema siguiendo la especificación planteada en el método, el cual se muestra a continuación en una plantilla que identifica la tarea que corresponde a cada actor.

Tabla 3

Identificación de tareas.

Rol	Tareas
Rol/Actor: Usuario Visitante	No realiza ninguna tarea, debido a que no tiene acceso al sistema.
Rol/Actor: Usuario diseñador	T1. Administrar líneas y estilos de calzado. T2. Crea catalogo por temporada.
Rol/Actor: Usuario jefe de producción	T3. Visualiza tareas a realizar en producción. T4. Seguimiento del estado de producción de pedidos.
Rol/Actor: Usuario director general	T5. Visualiza tablero principal de información y estadísticas de ventas. T6. Administrar clientes.

	<p><i>T7.</i> Genera cotizaciones.</p> <p><i>T8.</i> Seguimiento del estado en el que se encuentra un pedido y el tiempo de entrega.</p>
<p>Rol/Actor: Usuario administrador</p>	<p><i>T9.</i> Administra pedidos de clientes.</p> <p><i>T10.</i> Administra consolidaciones de forro y corte.</p> <p><i>T11.</i> Administra pedido de materiales de planta y suela.</p> <p><i>T12.</i> Administra y genera tarjetas de producción.</p> <p><i>T13.</i> Administra y genera numeración de calzado para los pedidos.</p> <p><i>T14.</i> Genera notas de venta.</p> <p><i>T15.</i> Genera reportes en general.</p> <p><i>T16.</i> Administrar usuarios, roles y perfiles.</p>

Nota. Tareas que pueden realizar los usuarios según el rol asignado en el sistema. Elaboración propia.

5.1.3 Diagrama de casos de uso

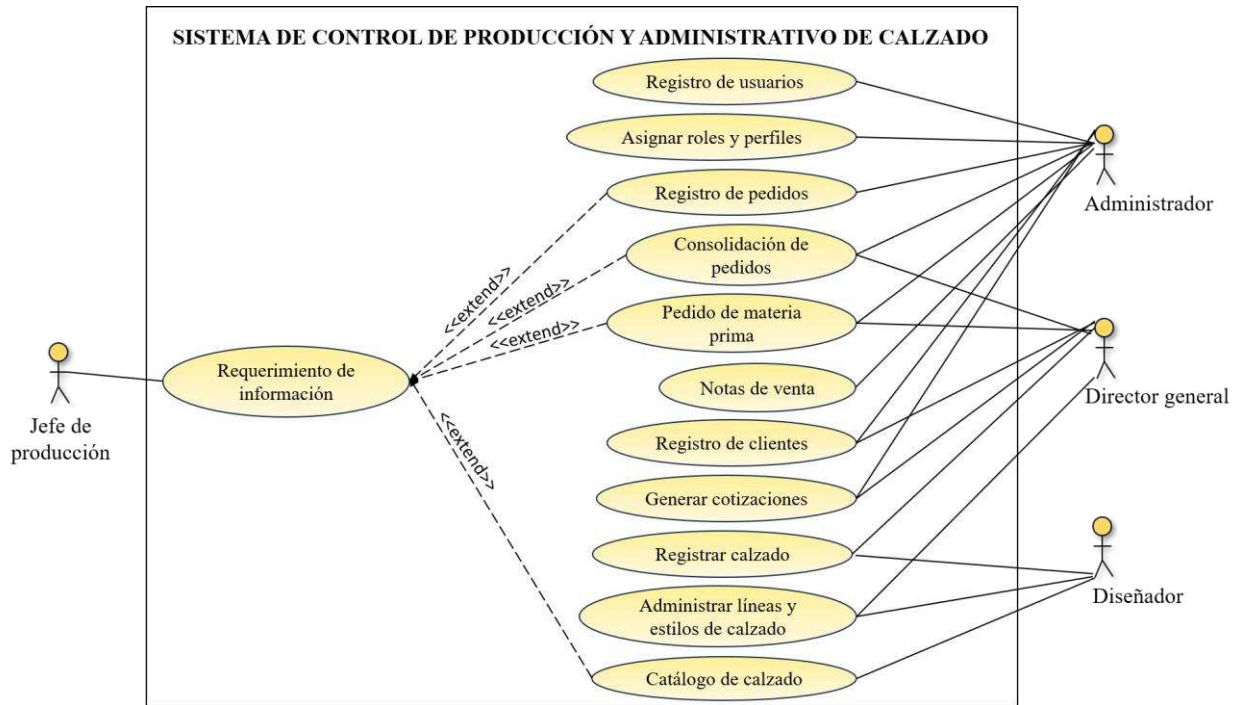
En esta subfase se mostrará la interacción del usuario con el sistema de forma gráfica, para ello se hace uso del diagrama de caso de uso mismos que se representan de manera formal, sencilla, rápida y clara.

El diagrama de flujo describe la relación que tienen los actores con el Sistema de Control de Producción y Administración de Calzado.

A continuación en la figura 6, se muestre el diagrama de caso de uso de negocio para los cuatro actores registrados en el sistema propuesto, se tiene 12 casos de uso que serán detallados en los escenarios de casos de uso.

Figura 6

Diagrama de caso de uso de negocio.



Nota. Interacción del usuario con el sistema de manera gráfica. Elaboración propia.

5.1.4 Escenario de casos de uso.

Una vez completada y teniendo la especificación de las tareas, se establece la funcionalidad propia del sistema por medio de los escenarios de los casos de uso, que muestra la relación que existe entre el usuario y el sistema, para cada una de las tareas, aplicando la plantilla reestructurada del método. En dicha plantilla se muestra la división de los pasos realizados o acciones que realiza el usuario, las respuestas que da la aplicación y la información de cada acción, además permiten especificar las precondiciones y postcondiciones que deben cumplirse para que el curso normal de las acciones se dé.

A continuación, se muestra y se detalla los escenarios de casos de uso:

Tabla 4

Escenario de caso de uso, requerimiento de información.

Nombre del caso de uso	Requerimiento de información	ID	1
Área	Sistema de consulta e información.		
Actor(es)	Jefe de producción, administrador, director Gral., diseñador.		
Interesado	Jefe de producción.		
Descripción	Permitir informar al jefe de producción acerca de los pedidos, tareas a realizar y el estado de producción (montaje, adorno, etc.) de los calzados.		
Evento desencadenador	El administrador ingresa al sistema con usuario y contraseña y visualiza información de su interés.		
Tipo de desencadenador	<input type="checkbox"/> Externo <input checked="" type="checkbox"/> Temporal		
Pasos realizados		Información de los pasos	
1. El jefe de producción inicia sesión en el sistema.		Se requiere usuario y contraseña del jefe de producción.	
2. Se lee el registro del jefe de producción y se verifica la contraseña.		Registro del jefe de producción, usuario y contraseña.	
3. El sistema devuelve las tareas a realizar con la información necesaria.		Listado de tareas a realizar.	
4. El jefe de producción ingresa a la sección de pedidos.		Lista de pedidos, plazos de entrega y estado de pedido.	
5. El jefe de producción elige pedido y hace clic en ver más detalles.		Información detallada del pedido.	
Precondiciones	El jefe de producción ya está registrado en el sistema.		
Postcondiciones	El jefe de producción obtiene información de su interés de manera exitosa.		
Suposiciones	El jefe de producción tiene un navegador y un nombre de usuario y contraseña válidos.		
Garantía de éxito	El jefe de producción tiene una lista de datos actualizados.		
Garantía mínima	El jefe de producción ingresa al sistema exitosamente.		
Requerimientos cumplidos	Permitir que el jefe de producción pueda visualizar la información necesaria mediante un sistema de forma segura.		
Prioridad	Alta		
Riesgo	Bajo		

Nota. Muestra la relación que existe entre el jefe de producción y el sistema, los pasos a realizar para las tareas encomendadas. Elaboración propia.

Tabla 5

Escenario de caso de uso, registro de usuarios.

Nombre del caso de uso	Registro de usuarios.	ID	2
Área	Administración de usuarios.		
Actor(es)	Administrador.		
Interesados	Administrador.		
Descripción	Permite al Administrador el registro, eliminación y/o actualización de usuarios.		
Evento desencadenador	El administrador ingresa al sistema con usuario y contraseña y administra usuarios.		
Tipo de desencadenador	<input type="checkbox"/> Externo <input checked="" type="checkbox"/> Temporal		
Pasos realizados		Información de los pasos	
1. El administrador inicia sesión en el sistema.		Se requiere usuario y contraseña del administrador.	
2. Se lee el registro del administrador y se verifica la contraseña.		Registro de administrador, usuario y contraseña.	
3. El administrador ingresa a la sección de usuarios.		Listado actualizado de usuarios registrados en el sistema.	
4. El administrador introduce datos a un formulario de registro o actualización y hace clic en el botón Guardar.		Formulario de registro o actualización de usuarios.	
5. Los datos de registro o actualización se valida en el servidor.		Formulario de registro o actualización de usuarios.	
6. Se escribe el registro en la base de datos de usuarios.		Formulario de registro o actualización de usuarios.	
7. El sistema muestra un mensaje de registro realizado con éxito.		Listado actualizado de usuarios registrados en el sistema.	
Precondiciones	El administrador ya está registrado en el sistema.		
Suposiciones	El administrador registró o actualizó un usuario con éxito.		
Postcondiciones	El administrador tiene un navegador y un nombre de usuario y contraseña válidos.		
Garantía de éxito	El administrador tiene una lista con datos actualizados.		
Garantía mínima	El administrador ingresa al sistema exitosamente.		
Requerimientos cumplidos	Permitir que el administrador pueda registrar, eliminar y/o actualizar un usuario mediante un sistema de forma segura.		
Prioridad	Alta		
Riesgo	Alta		

Nota. Muestra la relación que existe entre el administrador y el sistema para la administración de usuarios, los pasos a realizar para las tareas encomendadas. Elaboración propia.

Tabla 6

Escenario de caso de uso, asignar roles y perfiles.

Nombre del caso de uso	Asignar roles y perfiles.	ID	3
Área	Administración de roles y perfiles.		
Actor(es)	Administrador.		
Interesados	Administrador.		
Descripción	Permitir al administrador el registro, eliminación y/o actualización de roles y perfiles.		
Evento desencadenador	El administrador ingresa al sistema con usuario y contraseña y administra roles y perfiles.		
Tipo de desencadenador	<input type="checkbox"/> Externo <input checked="" type="checkbox"/> Temporal		
Pasos realizados		Información de los pasos	
1. El administrador inicia sesión en el sistema.		Se requiere usuario y contraseña del administrador.	
2. Se lee el registro del administrador y se verifica la contraseña.		Registro de administrador, usuario y contraseña.	
3. El administrador ingresa a la sección de administración de roles y perfiles.		Listado de perfiles registrados en el sistema.	
4. El administrador registra o selecciona un perfil y concede permisos, hace clic en guardar cambios.		Listado de módulos con opción a seleccionar.	
5. Los datos de registro o actualización se valida en el servidor.		Listado de módulos con opción a seleccionar.	
6. Se escribe el registro en la base de datos de roles y perfiles.		Listado de módulos con opción a seleccionar.	
7. El sistema muestra un mensaje de registro realizado con éxito.		Listado actualizado con los módulos permitidos para el perfil seleccionado.	
Precondiciones	El administrador ya está registrado en el sistema.		
Suposiciones	El administrador registró o actualizó roles y perfiles con éxito.		
Postcondiciones	El administrador tiene un navegador y un nombre de usuario y contraseña válidos.		
Garantía de éxito	El administrador tiene una lista con datos actualizados.		
Garantía mínima	El administrador ingresa al sistema exitosamente.		

Requerimientos cumplidos	Permitir que el administrador pueda registrar, eliminar y/o actualizar roles y perfiles mediante un sistema de forma segura.
Prioridad	Alta
Riesgo	Alta

Nota. Muestra la relación que existe entre el administrador y el sistema para la administración de roles y perfiles, los pasos a realizar para las tareas encomendadas. Elaboración propia.

Tabla 7

Escenario de caso de uso, registro de pedidos.

Nombre del caso de uso	Registro de pedidos.	ID 4
Área	Administración de pedidos.	
Actor(es)	Administrador.	
Interesados	Administrador.	
Descripción	Permite al director general el registro, eliminación y/o actualización de pedidos.	
Evento desencadenador	El director general ingresa al sistema con usuario y contraseña y administra pedidos.	
Tipo de desencadenador	<input type="checkbox"/> Externo <input checked="" type="checkbox"/> Temporal	
Pasos realizados		Información de los pasos
1. El administrador inicia sesión en el sistema.	Se requiere usuario y contraseña del administrador.	
2. Se lee el registro del administrador y se verifica la contraseña.	Registro de administrador, usuario y contraseña.	
3. El administrador ingresa a la sección de administración de pedidos.	Formulario de registro de pedido.	
4. El administrador introduce datos del pedido, hace clic en el botón guardar.	Formulario de registro de pedido.	
5. Los datos de registro o actualización se valida en el servidor.	Formulario de registro de pedido.	
6. Se escribe el registro en la base de datos de pedidos.	Formulario de registro de pedido.	
7. El sistema muestra un mensaje de registro realizado con éxito.	Formulario de registro de pedido.	
Precondiciones	El administrador ya está registrado en el sistema.	
Suposiciones	El administrador registró un pedido con éxito.	

Postcondiciones	El administrador tiene un navegador y un nombre de usuario y contraseña válidos.
Garantía de éxito	El administrador tiene una lista con datos actualizados.
Garantía mínima	El administrador ingresa al sistema exitosamente.
Requerimientos cumplidos	Permitir que el administrador pueda registrar pedidos mediante un sistema de forma segura.
Prioridad	Alta
Riesgo	Alta

Nota. Muestra la relación que existe entre el administrador y el sistema para la administración de pedidos, los pasos a realizar para las tareas encomendadas. Elaboración propia.

Tabla 8

Escenario de caso de uso, consolidación de pedidos.

Nombre del caso de uso	Consolidación de pedidos.	ID	5
Área	Administración de pedidos.		
Actor(es)	Administrador, director general, jefe de producción.		
Interesados	Administrador.		
Descripción	Permite al Administrador seleccionar los pedidos que se desean consolidar en material de forro o corte y generar un archivo PDF para posteriormente imprimir si así lo desea.		
Evento desencadenador	El administrador ingresa al sistema con usuario y contraseña y genera el consolidado de forro y corte.		
Tipo de desencadenador	<input type="checkbox"/> Externo <input checked="" type="checkbox"/> Temporal		
Pasos realizados		Información de los pasos	
1. El administrador inicia sesión en el sistema.		Se requiere usuario y contraseña del administrador.	
2. Se lee el registro del administrador y se verifica la contraseña.		Registro de administrador, usuario y contraseña.	
3. El administrador ingresa a la sección de consolidados.		Lista de consolidados realizados anteriormente.	
4. El administrador presiona el botón añadir consolidado, selecciona los pedidos que desea consolidar y guarda cambios.		Lista de pedidos con opción de seleccionar.	
5. La consolidación se valida en el servidor.		Lista de pedidos con opción de seleccionar.	
6. Se escribe la consolidación en la base de datos de consolidaciones.		Lista de pedidos con opción de seleccionar.	

7. Los pedidos consolidados ya no se muestran más para ser consolidados.	Lista de consolidados cambia agregando la nueva consolidación.
8. El sistema muestra un mensaje de confirmación de consolidación exitosa.	Información de consolidaciones.
Precondiciones	El administrador ya está registrado en el sistema.
Postcondiciones	El administrador generó con éxito la consolidación.
Suposiciones	El administrador tiene un navegador y un nombre de usuario y contraseña válidos.
Garantía de éxito	El administrador tiene una lista con datos actualizados.
Garantía mínima	El administrador ingresa al sistema exitosamente.
Requerimientos cumplidos	Permitir que el administrador pueda generar consolidaciones mediante un sistema de forma segura.
Prioridad	Media
Riesgo	Baja

Nota. Muestra la relación que existe entre el administrador y el sistema para generar consolidaciones. Identificación e inicialización, pasos realizados y las condiciones, suposiciones, preguntas para las tareas encomendadas. Elaboración propia.

Tabla 9

Escenario de caso de uso, pedido de materia prima.

Nombre del caso de uso	Pedido de materia prima.	ID	6
Área	Administración de materia prima.		
Actor(es)	Administrador, director general, jefe de producción.		
Interesados	Administrador.		
Descripción	Permite al Administrador seleccionar pedidos para realizar el pedido de materia prima y generar un archivo PDF para posteriormente enviarlo al proveedor.		
Evento desencadenador	El administrador ingresa al sistema con usuario y contraseña y genera el pedido de materia prima.		
Tipo de desencadenador	<input type="checkbox"/> Externo <input checked="" type="checkbox"/> Temporal		
Pasos realizados		Información de los pasos	
1. El administrador inicia sesión en el sistema.		Se requiere usuario y contraseña del administrador.	
2. Se lee el registro del administrador y se verifica la contraseña.		Registro de administrador, usuario y contraseña.	

3. El administrador ingresa a la sección de materia prima.	Lista de pedidos de materia prima realizados anteriormente.
4. El administrador presiona el botón añadir pedido de materia prima, selecciona los pedidos que serán tomados en cuenta para el pedido de materia prima y guarda cambios.	Lista de pedidos con opción de seleccionar.
5. El pedido de materia prima se valida en el servidor.	Lista de pedidos con opción de seleccionar.
6. Se escribe el pedido de materia prima en la base de datos de materia prima.	Lista de pedidos con opción de seleccionar.
7. Los pedidos que se seleccionó para el pedido de materia prima ya no se muestran más para generar el siguiente pedido de materia prima.	Lista de pedidos de materia prima se actualiza agregando la nueva petición de materia prima.
8. El sistema muestra un mensaje de confirmación de pedido de materia prima exitosa.	Información de petición de materia prima.
Precondiciones	El administrador ya está registrado en el sistema.
Postcondiciones	El administrador generó con éxito la petición de materia prima.
Suposiciones	El administrador tiene un navegador y un nombre de usuario y contraseña válidos.
Garantía de éxito	El administrador tiene una lista con datos actualizados.
Garantía mínima	El administrador ingresa al sistema exitosamente.
Requerimientos cumplidos	Permitir que el administrador pueda generar el pedido de materia prima mediante un sistema de forma segura.
Prioridad	Media
Riesgo	Baja

Nota. Muestra la relación que existe entre el administrador y el sistema para generar el pedido de materia prima. Identificación e inicialización, pasos realizados y las condiciones, suposiciones, preguntas para las tareas encomendadas. Elaboración propia.

Tabla 10

Escenario de caso de uso, notas de venta.

Nombre del caso de uso	Notas de venta.	ID	7
Área	Administración de pedidos.		

Actor(es)	Administrador.
Interesados	Administrador.
Descripción	Permite al Administrador generar una nota de venta de un pedido en formato PDF y tener un registro de las ventas realizadas.
Evento desencadenador	El administrador ingresa al sistema con usuario y contraseña y genera la nota de venta del pedido seleccionado.
Tipo de desencadenador	<input type="checkbox"/> Externo <input checked="" type="checkbox"/> Temporal
Pasos realizados	
Información de los pasos	
1. El administrador inicia sesión en el sistema.	Se requiere usuario y contraseña del administrador.
2. Se lee el registro del administrador y se verifica la contraseña.	Registro de administrador, usuario y contraseña.
3. El administrador ingresa a la sección de notas de venta.	Listado de pedidos y de notas de venta realizados con anterioridad.
4. El administrador selecciona el pedido y los lotes que se desea incluir en la nota de venta, se presiona botón generar nota de venta.	Listado de pedidos con opción de seleccionar y listado de notas de venta.
5. La nota de venta se valida en el servidor.	Listado de pedidos con opción de seleccionar y listado de notas de venta.
6. Se escribe la nota de venta en la base de datos de materia prima.	Listado de pedidos con opción de seleccionar y listado de notas de venta.
7. Los pedidos y/o lotes de un pedido que ya se generó una nota de venta ya no se muestran en listado de pedidos.	Listado de pedidos y listado de notas de venta se actualizan.
8. El sistema muestra una ventana con el reporte de la nota de venta.	Información de la nota de venta.
Precondiciones	El administrador ya está registrado en el sistema.
Postcondiciones	El administrador generó con éxito la nota de venta.
Suposiciones	El administrador tiene un navegador y un nombre de usuario y contraseña válidos.
Garantía de éxito	El administrador tiene una lista con datos actualizados.
Garantía mínima	El administrador ingresa al sistema exitosamente.
Requerimientos cumplidos	Permitir que el administrador pueda generar la nota de venta mediante un sistema de forma segura.
Prioridad	Alta
Riesgo	Media

Nota. Muestra la relación que existe entre el administrador y el sistema para generar una nota de venta. Identificación e inicialización, pasos realizados y las condiciones, suposiciones, preguntas para las tareas encomendadas. Elaboración propia.

Tabla 11

Escenario de caso de uso, registro de clientes.

Nombre del caso de uso	Registro de clientes.	ID	8
Área	Administración de clientes.		
Actor(es)	Administrador, director general.		
Interesados	Director general.		
Descripción	Permite al director general el registro, eliminación y/o actualización de clientes.		
Evento desencadenador	El director general ingresa al sistema con usuario y contraseña y administra clientes.		
Tipo de desencadenador	<input type="checkbox"/> Externo <input checked="" type="checkbox"/> Temporal		
Pasos realizados		Información de los pasos	
1. El director general inicia sesión en el sistema.		Se requiere usuario y contraseña del director general.	
2. Se lee el registro del director general y se verifica la contraseña.		Registro de director general, usuario y contraseña.	
3. El director general ingresa a la sección de clientes.		Listado actualizado de clientes registrados en el sistema.	
4. El director general introduce datos a un formulario de registro o actualización y hace clic en el botón Guardar.		Formulario de registro o actualización de cliente.	
5. Los datos de registro o actualización se valida en el servidor.		Formulario de registro o actualización de cliente.	
6. Se escribe el registro en la base de datos de clientes.		Formulario de registro o actualización de cliente.	
7. El sistema muestra un mensaje de registro realizado con éxito.		Listado actualizado de clientes registrados en el sistema.	
Precondiciones	El director general ya está registrado en el sistema.		
Postcondiciones	El director general registró o actualizó un cliente con éxito.		
Suposiciones	El director general tiene un navegador y un nombre de usuario y contraseña válidos.		
Garantía de éxito	El director general tiene una lista con datos actualizados.		

Garantía mínima	El director general ingresa al sistema exitosamente.
Requerimientos cumplidos	Permitir que el director general pueda registrar, eliminar y/o actualizar un cliente mediante un sistema de forma segura.
Prioridad	Alta
Riesgo	Alta

Nota. Muestra la relación que existe entre el director general y el sistema para registrar, eliminar y/o actualizar un cliente. Identificación e inicialización, pasos realizados y las condiciones, suposiciones, preguntas para las tareas encomendadas. Elaboración propia.

Tabla 12

Escenario de caso de uso, generar cotizaciones.

Nombre del caso de uso	Generar cotizaciones.	ID	9
Área	Administración de cotizaciones.		
Actor(es)	Administrador, director general.		
Interesados	Director general.		
Descripción	Permitir al director general añadir una cotización de pedido de calzado mediante el uso de un sistema.		
Evento desencadenador	El director general ingresa al sistema con usuario y contraseña y administra las cotizaciones.		
Tipo de desencadenador	<input type="checkbox"/> Externo <input checked="" type="checkbox"/> Temporal		
Pasos realizados		Información de los pasos	
1. El director general inicia sesión en el sistema.		Se requiere usuario y contraseña del director general.	
2. Se lee el registro del director general y se verifica la contraseña.		Registro de director general, usuario y contraseña.	
3. El director general ingresa a la sección de cotizaciones.		Listado actualizado de cotizaciones registrados en el sistema.	
4. El director general hace clic en el botón añadir cotización, selecciona cliente, agrega calzados con su respectiva numeración y hace clic en el botón generar cotización.		Formulario de registro de cotización de calzado.	
5. Los datos introducidos se valida en el servidor.		Formulario de registro de cotización de calzado.	
6. Se escribe el registro en la base de datos de cotizaciones.		Formulario de registro de cotización de calzado.	

7. El sistema muestra una venta con la cotización y la opción de descargar en formato PDF.	Información de cotización de calzado.
Precondiciones	El director general ya está registrado en el sistema.
Postcondiciones	El director general registró la cotización con éxito.
Suposiciones	El director general tiene un navegador y un nombre de usuario y contraseña válidos.
Garantía de éxito	El director general tiene una lista con datos actualizados.
Garantía mínima	El director general ingresa al sistema exitosamente.
Requerimientos cumplidos	Permitir que el director general pueda registrar, eliminar y/o actualizar una cotización mediante un sistema de forma segura.
Prioridad	Alta
Riesgo	Alta

Nota. Muestra la relación que existe entre el director general y el sistema para registrar, eliminar y/o actualizar una cotización de calzado. Identificación e inicialización, pasos realizados y las condiciones, suposiciones, preguntas para las tareas encomendadas. Elaboración propia.

Tabla 13

Escenario de caso de uso, registrar calzado.

Nombre del caso de uso	Registrar Calzado.	ID	10
Área	Diseño de calzados.		
Actor(es)	Director general, diseñador.		
Interesados	Diseñador.		
Descripción	Permitir al diseñador el registro, eliminación y/o actualización de calzado.		
Evento desencadenador	El diseñador ingresa al sistema con usuario y contraseña y administra calzados.		
Tipo de desencadenador	<input type="checkbox"/> Externo <input checked="" type="checkbox"/> Temporal		
Pasos realizados		Información de los pasos	
1. El diseñador inicia sesión en el sistema.		Se requiere usuario y contraseña del diseñador.	
2. Se lee el registro del diseñador y se verifica la contraseña.		Registro de diseñador, usuario y contraseña.	
3. El diseñador ingresa a la sección de administrar calzado.		Listado actualizado de calzados registrados en el sistema.	

4. El diseñador registra, elimina y/o actualiza datos de calzado, y hace clic en guardar o modificar.	Formulario de registro de calzado.
5. Los datos introducidos se valida en el servidor.	Formulario de registro de calzado.
6. Se escribe el registro en la base de datos de calzado.	Formulario de registro de calzado.
7. El sistema muestra un mensaje de confirmación de registro exitoso de calzado.	Listado actualizado de calzados registrados en el sistema.
Precondiciones	El diseñador ya está registrado en el sistema.
Postcondiciones	El diseñador registró, elimino y/o actualizó un calzado con éxito.
Suposiciones	El diseñador tiene un navegador y un nombre de usuario y contraseña válidos.
Garantía de éxito	El diseñador tiene una lista con datos actualizados.
Garantía mínima	El diseñador ingresa al sistema exitosamente.
Requerimientos cumplidos	Permitir que el diseñador pueda registrar, eliminar y/o actualizar un calzado mediante un sistema de forma segura.
Prioridad	Media
Riesgo	Media

Nota. Muestra la relación que existe entre el diseñador y el sistema para registrar, eliminar y/o actualizar un calzado. Identificación e inicialización, pasos realizados y las condiciones, suposiciones, preguntas para las tareas encomendadas. Elaboración propia.

Tabla 14

Escenario de caso de uso, administrar líneas y estilos de calzado.

Nombre del caso de uso	Administrar líneas y estilos de calzado.	ID	11
Área	Diseño de calzados.		
Actor(es)	Director general, diseñador.		
Interesados	Diseñador.		
Descripción	Permitir al diseñador el registro, eliminación y/o actualización de línea, estilos de calzado.		
Evento desencadenador	El diseñador ingresa al sistema con usuario y contraseña y administra líneas y estilos de calzado.		
Tipo de desencadenador	<input type="checkbox"/> Externo <input checked="" type="checkbox"/> Temporal		

Pasos realizados	Información de los pasos
8. El diseñador inicia sesión en el sistema.	Se requiere usuario y contraseña del diseñador.
9. Se lee el registro del diseñador y se verifica la contraseña.	Registro de diseñador, usuario y contraseña.
10. El diseñador ingresa a la sección de administrar líneas y estilos.	Listado actualizado de líneas y estilos registrados en el sistema.
11. El diseñador registra, elimina y/o actualiza datos de línea y etilo, y hace clic en guardar o modificar.	Formulario de registro de líneas y estilos de calzado.
12. Los datos introducidos se valida en el servidor.	Formulario de registro de líneas y estilos de calzado.
13. Se escribe el registro en la base de datos de líneas y estilos.	Formulario de registro de líneas y estilos de calzado.
14. El sistema muestra un mensaje de confirmación de registro exitoso de línea o estilo.	Listado actualizado de líneas y estilos registrados en el sistema.
Precondiciones	El diseñador ya está registrado en el sistema.
Postcondiciones	El diseñador registró, elimino y/o actualizó una línea o estilo con éxito.
Suposiciones	El diseñador tiene un navegador y un nombre de usuario y contraseña válidos.
Garantía de éxito	El diseñador tiene una lista con datos actualizados.
Garantía mínima	El diseñador ingresa al sistema exitosamente.
Requerimientos cumplidos	Permitir que el diseñador pueda registrar, eliminar y/o actualizar líneas y estilos mediante un sistema de forma segura.
Prioridad	Alta
Riesgo	Alta

Nota. Muestra la relación que existe entre el diseñador y el sistema para registrar, eliminar y/o actualizar líneas y estilos. Identificación e inicialización, pasos realizados y las condiciones, suposiciones, preguntas para las tareas encomendadas. Elaboración propia.

Tabla 15

Escenario de caso de uso, catálogo de calzado.

Nombre del caso de uso	Catálogo de calzado.	ID	12
Área	Diseño de calzado.		

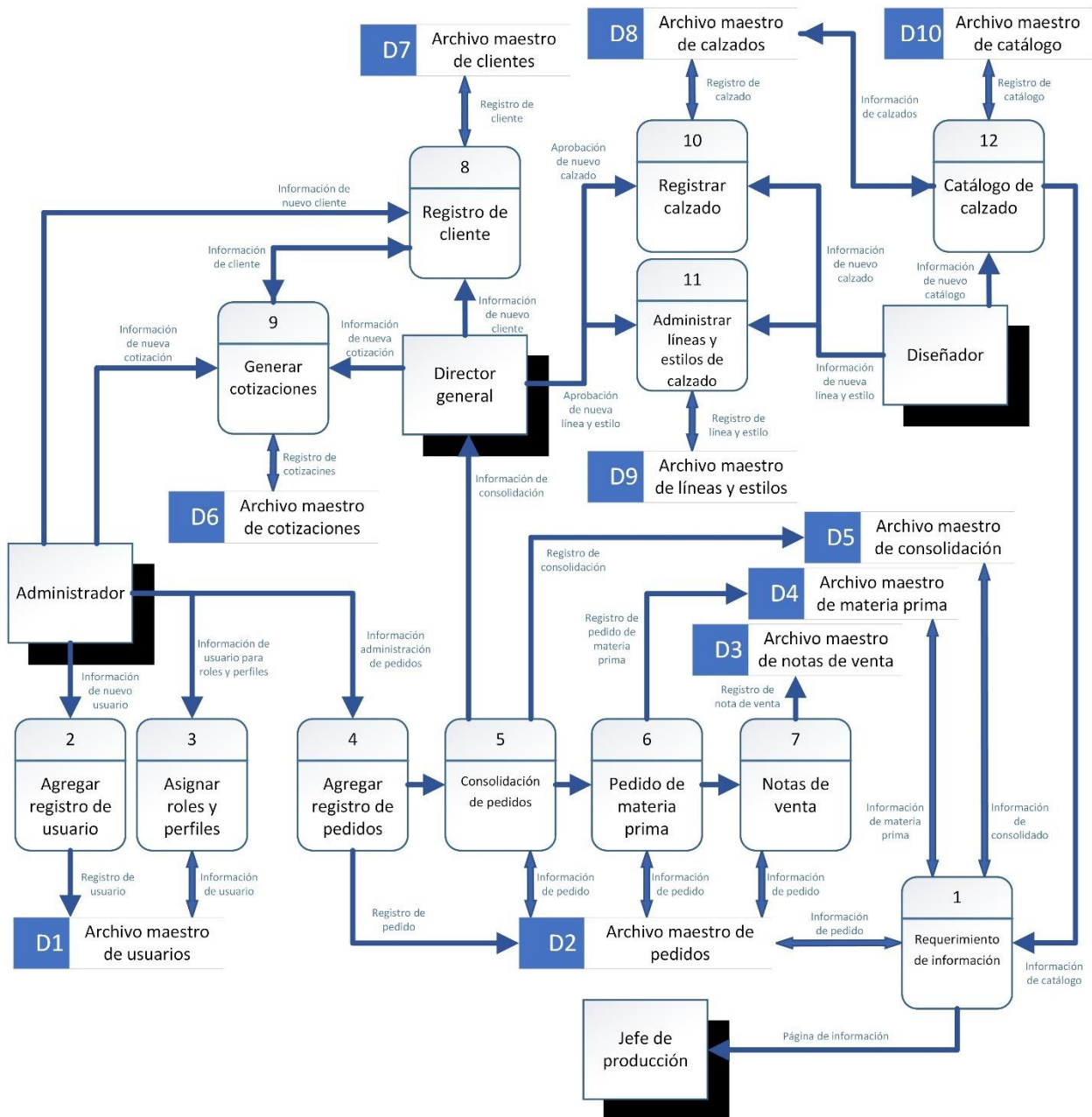
Actor(es)	Diseñador, jefe de producción.	
Interesados	Diseñador.	
Descripción	Permitir al diseñador generar un catálogo de calzado con sus especificaciones.	
Evento desencadenador	El diseñador ingresa al sistema con usuario y contraseña y administra catálogo de calzado.	
Tipo de desencadenador	<input type="checkbox"/> Externo <input checked="" type="checkbox"/> Temporal	
Pasos realizados		Información de los pasos
1. El diseñador inicia sesión en el sistema.	Se requiere usuario y contraseña del diseñador.	
2. Se lee el registro del diseñador y se verifica la contraseña.	Registro de diseñador, usuario y contraseña.	
3. El diseñador ingresa a la sección de administrar catálogo de calzado.	Listado actualizado de catálogos por temporada en el sistema.	
4. El diseñador introduce datos y genera un catálogo, hace clic en guardar.	Formulario de registro de catálogo de calzado.	
5. Los datos introducidos se valida en el servidor.	Formulario de registro de catálogo de calzado	
6. Se escribe el registro en la base de datos de catálogo.	Formulario de registro de catálogo de calzado	
7. El sistema muestra un mensaje de confirmación de registro exitoso con el catálogo.	Listado actualizado de catálogos por temporada registrados en el sistema.	
Precondiciones	El diseñador ya está registrado en el sistema.	
Postcondiciones	El diseñador registró y generó un catálogo con éxito.	
Suposiciones	El diseñador tiene un navegador y un nombre de usuario y contraseña válidos.	
Garantía de éxito	El diseñador tiene una lista con datos actualizados.	
Garantía mínima	El diseñador ingresa al sistema exitosamente.	
Requerimientos cumplidos	Permitir que el diseñador pueda registrar y generar un catálogo mediante un sistema de forma segura.	
Prioridad	Media	
Riesgo	Baja	

Nota. Muestra la relación que existe entre el diseñador y el sistema para registrar y generar un catálogo de calzado. Identificación e inicialización, pasos realizados y las condiciones, suposiciones, preguntas para las tareas encomendadas. Elaboración propia.

5.2 Diagrama de flujo de datos a nivel de contexto

Figura 7

Diagrama de flujo de datos a nivel de contexto del sistema.



Nota. Describe el flujo de la información de la base de datos a través de procedimientos solicitados por las entidades. Elaboración propia.

En esta etapa se identifica el flujo de datos del sistema, son los datos que fluyen hacia el sistema y que salen del sistema además de procesamiento de estos datos. Es también conocido como modelo ambiental el cual está constituido por procesos, entidades, flujo de datos y almacén de datos.

La figura 7 muestra el diagrama de flujo de datos a nivel de contexto del sistema. Se puede identificar las entidades (administrador, director general, diseñador y jefe de producción) que interactuarán en cada proceso generando y obteniendo información a través del flujo de datos y el almacenamiento de datos.

5.3 Diseño navegacional

En esta fase se describe la estructura de navegación de la aplicación. La navegación es definida primordialmente por el tipo de usuario, especialmente en el caso de la información restringida como documentos y artículos que solo son para los miembros de la empresa.

Se definirá la información que será presentada y la posible navegación entre ellas. A que estructuras el usuario tendrá acceso: nodos, enlaces y estructuras de acceso. La metodología refinada propone y admite la construcción de dos modelos o dos tipos de diagramas los cuales son el esquema de clases navegacionales no abstractas y esquema de contexto navegacional mejorado.

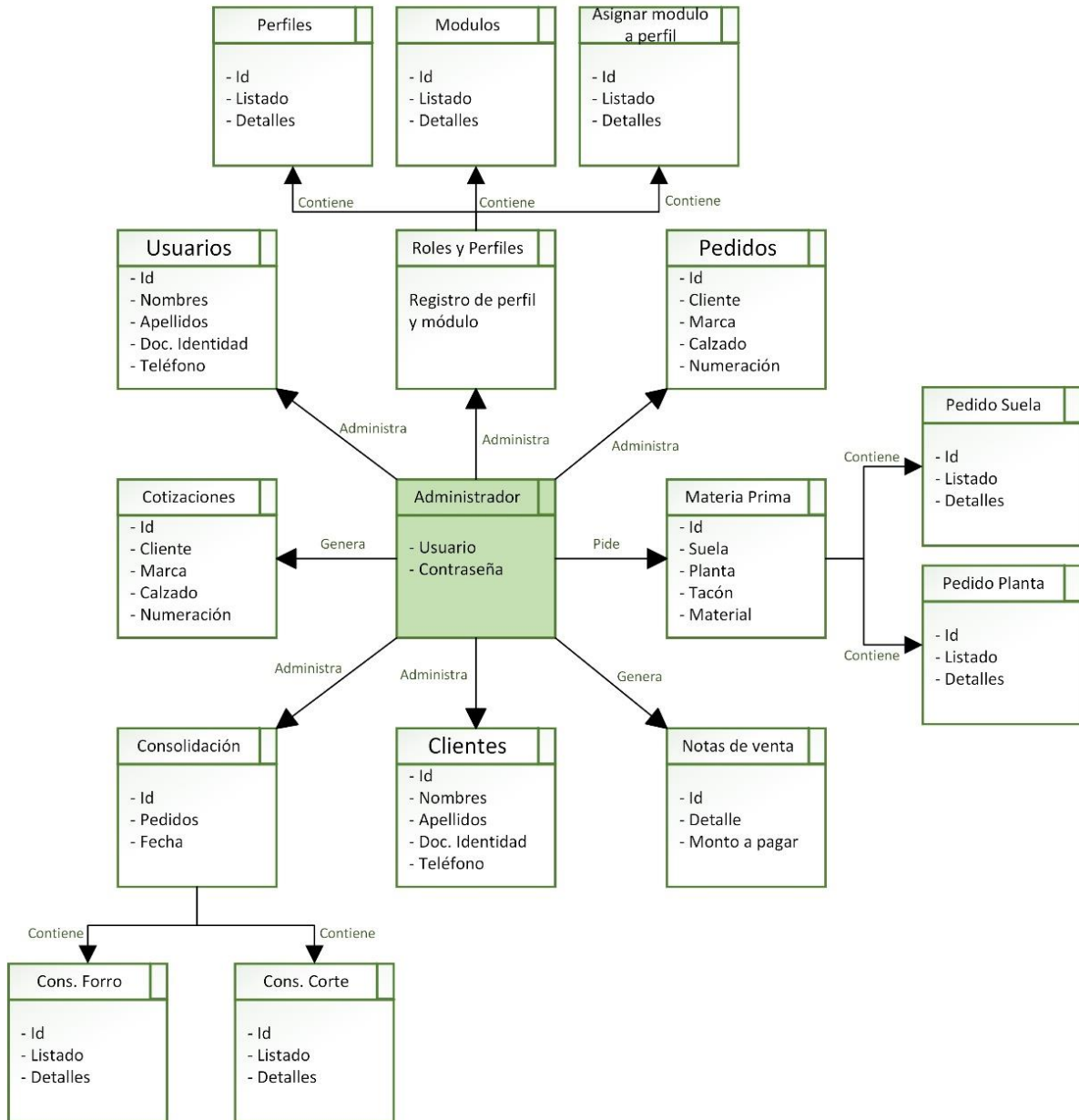
5.3.1 Esquema de clases navegacionales no abstractas

Un esquema de clases de navegacionales está compuesto por nodos y enlaces los cuales fueron obtenidos del modelo conceptual. Estas clases definidas muestran cómo será la navegación por el sistema web para los usuarios.

A continuación, se presenta los esquemas de clases navegacionales tomando en cuenta los diferentes perfiles de usuarios identificados.

Figura 8

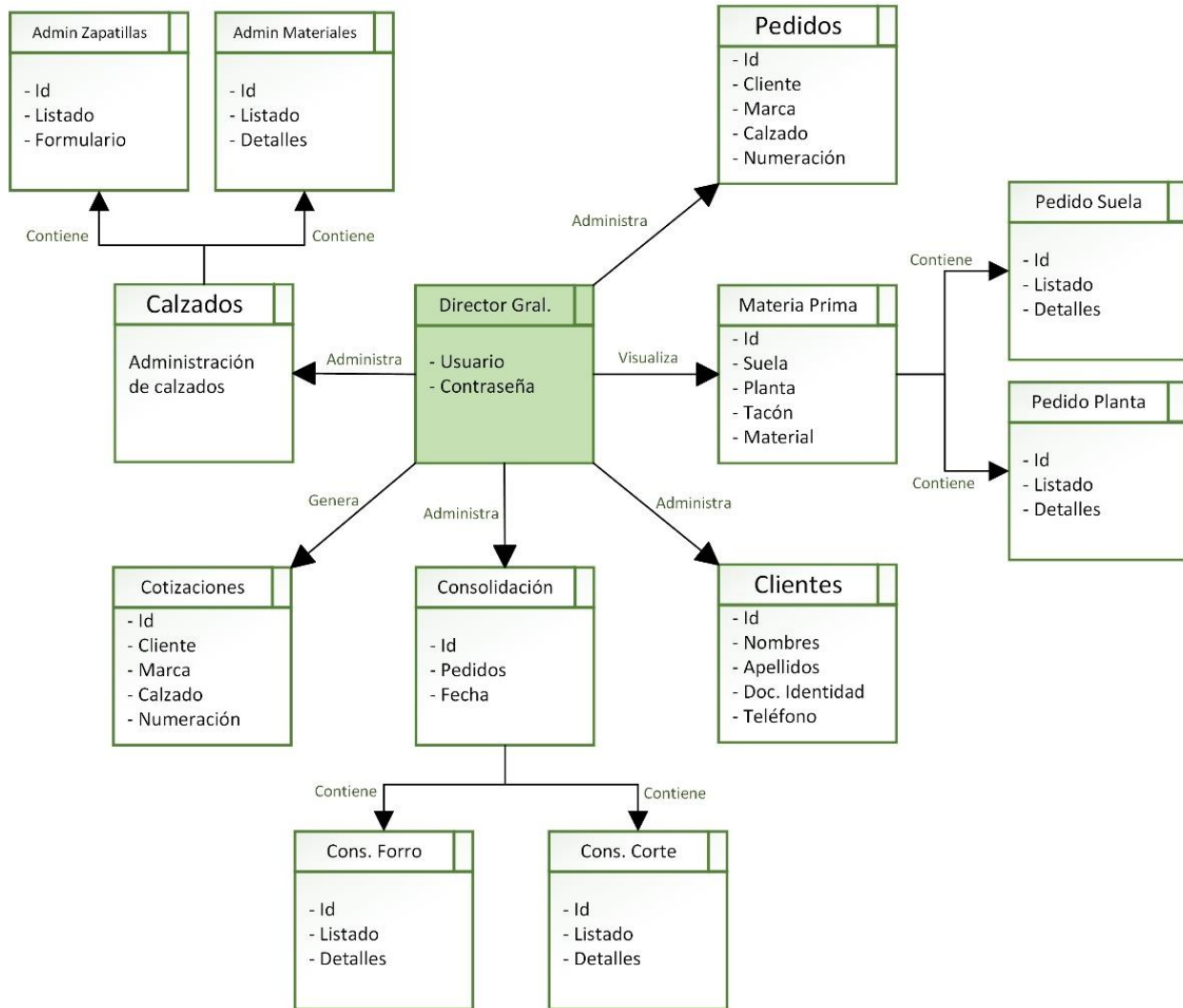
Esquema de clase navegacional no abstracta para el administrador.



Nota. Muestra la manera que el usuario administrador navegará por el sistema y la información que visualizará o administrará. Elaboración propia.

Figura 9

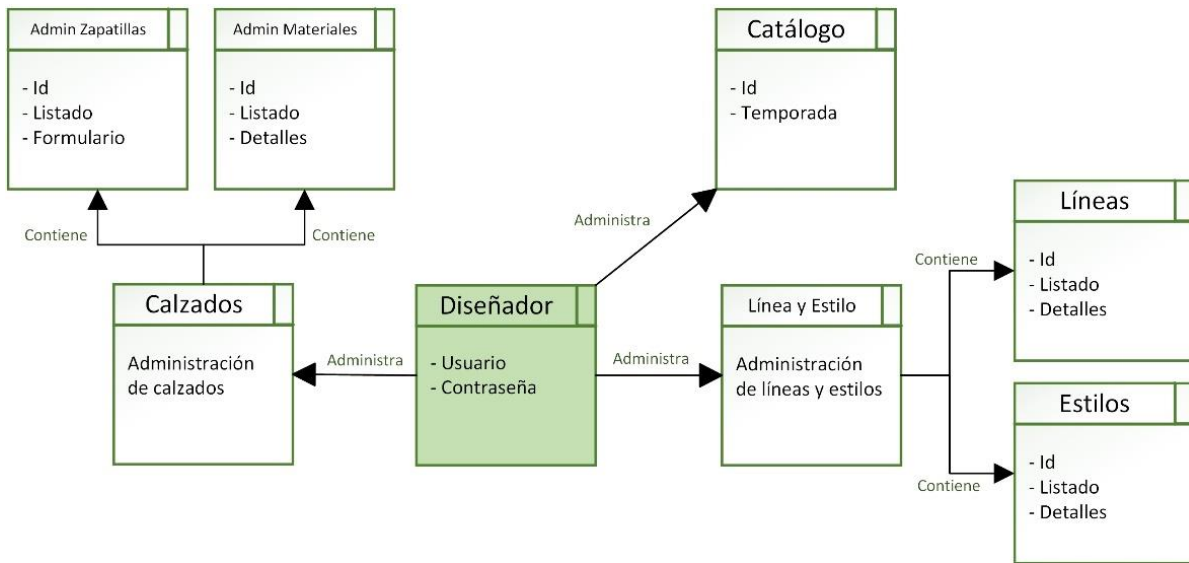
Esquema de clase navegacional no abstracta para el director general.



Nota. Muestra la manera que el usuario director general navegará por el sistema y la información que visualizará o administrará. Elaboración propia.

Figura 10

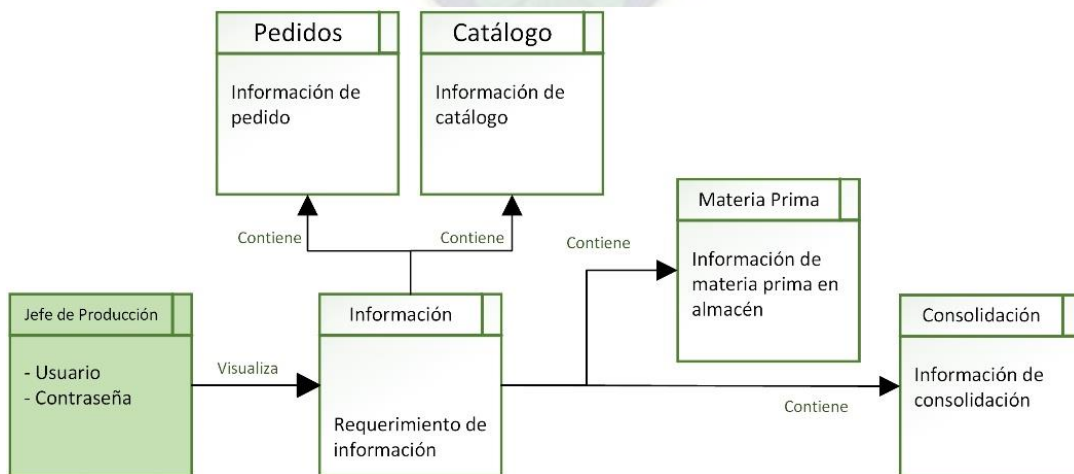
Esquema de clase navegacional no abstracta para el diseñador.



Nota. Muestra la manera que el usuario diseñador navegará por el sistema y la información que visualizará o administrará. Elaboración propia.

Figura 11

Esquema de clase navegacional no abstracta para el jefe de producción.



Nota. Muestra la manera que el usuario jefe de producción navegará por el sistema y la información que visualizará. Elaboración propia.

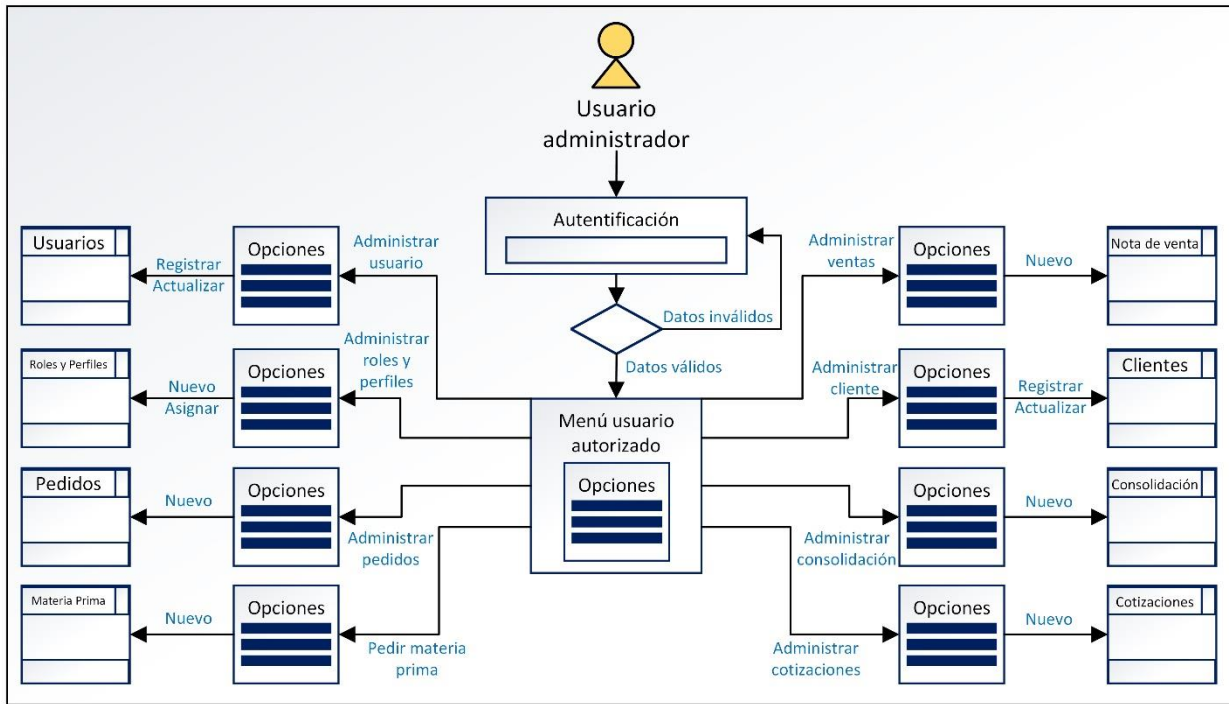
5.3.2 Esquema de contexto navegacional mejorado

En esta fase se presenta las diferentes maneras en la que un nodo puede ser accedido en un contexto. Es necesario definir el contexto en el cual se desenvolverá el usuario es decir la información a la que tendrá acceso y las posibilidades permitidas para su manejo.

Se ha definido un esquema de contexto para cada usuario del sistema que a continuación se puede observar.

Figura 12

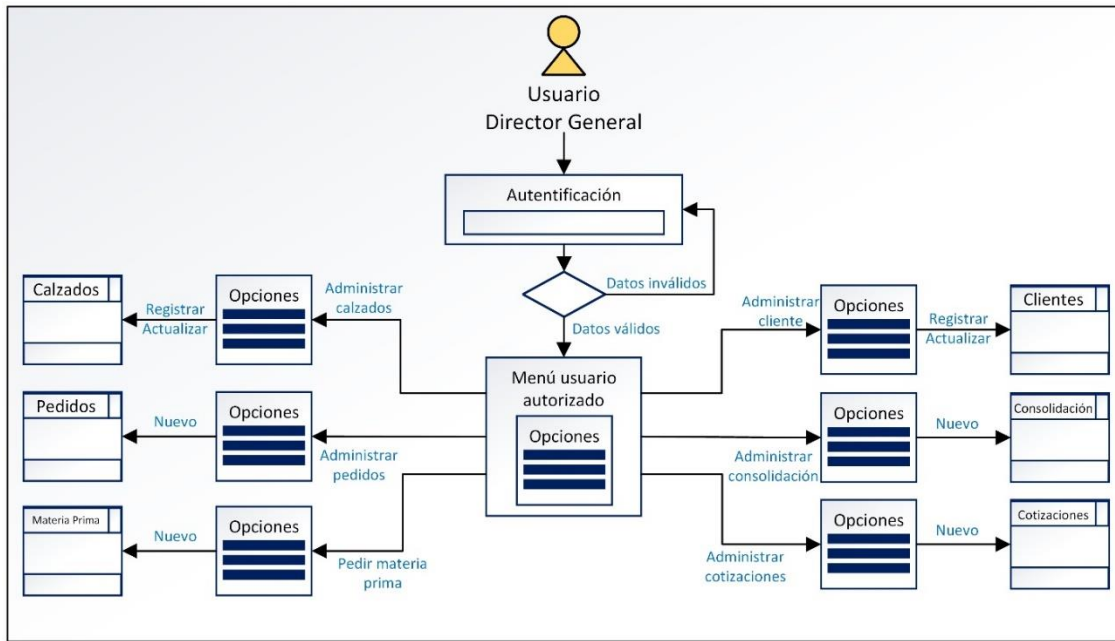
Esquema de contexto navegacional mejorado para el administrador.



Nota. Define el contexto en el cual el administrador se desenvolverá por el sistema y la información que tiene acceso y las posibilidades permitidas para su manejo. Elaboración propia.

Figura 13

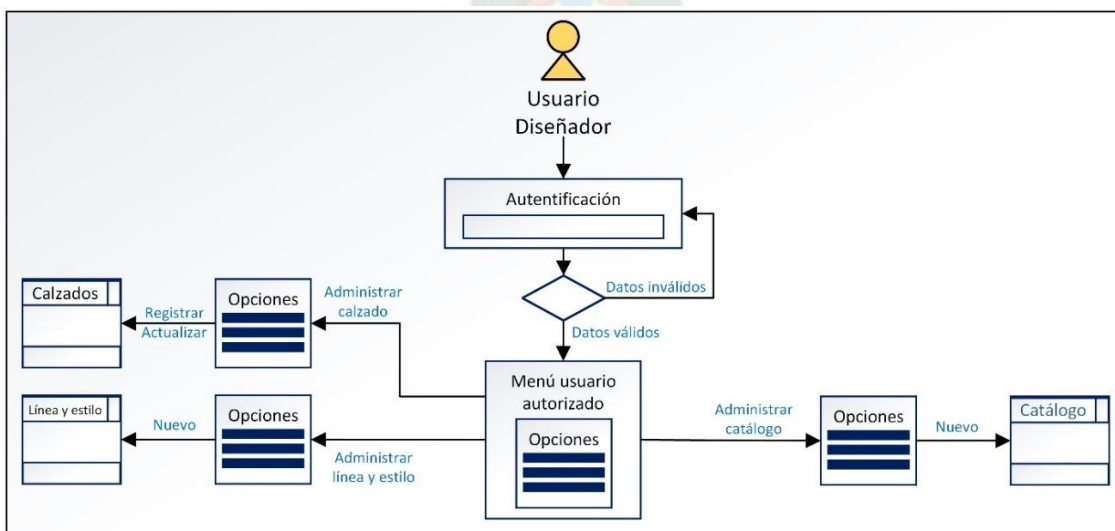
Esquema de contexto navegacional mejorado para el director general.



Nota. Define el contexto en el cual el director general se desenvolverá por el sistema y la información que tiene acceso y las posibilidades permitidas para su manejo. Elaboración propia.

Figura 14

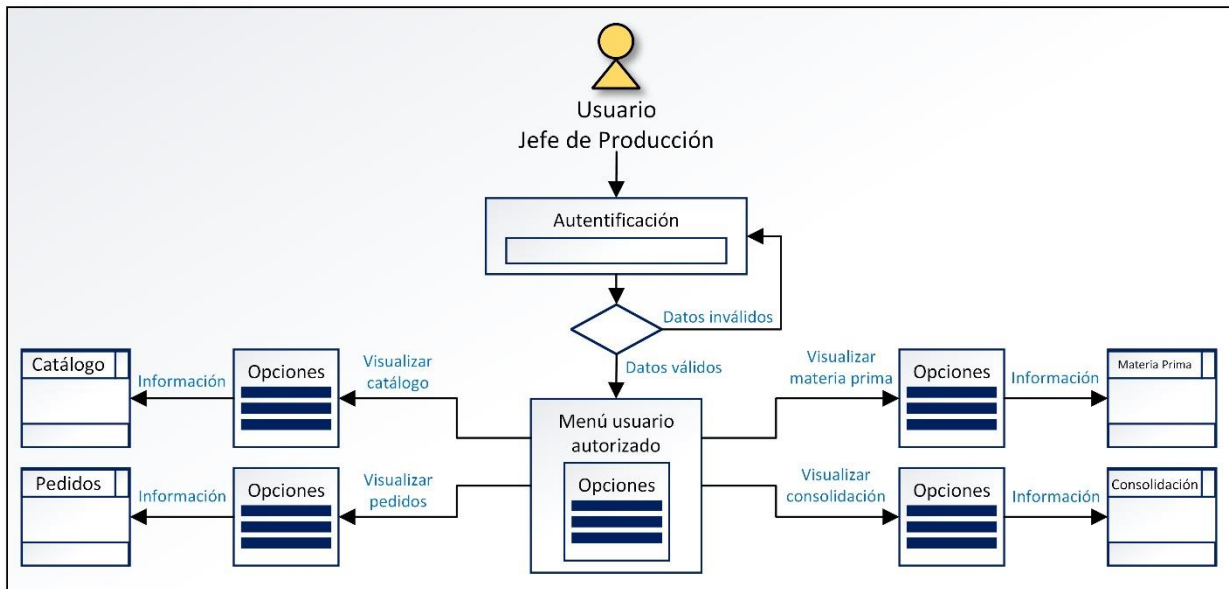
Esquema de contexto navegacional mejorado para el diseñador.



Nota. Defina el contexto en el cual el diseñador se desenvolverá por el sistema y la información que tiene acceso y las posibilidades permitidas para su manejo. Elaboración propia.

Figura 15

Esquema de contexto navegacional mejorado para el jefe de producción.



Nota. Defina el contexto en el cual el jefe de producción se desenvolverá por el sistema y la información que tiene acceso y las posibilidades permitidas para su manejo. Elaboración propia.

5.4 Diseño de interfaz no abstracta

Una vez que las aplicaciones de estructura navegacional han sido definidos, se debe especificar ahora aspectos de la interfaz. El diseño de una interfaz nos ayuda a decidir que objetos serán mostrados, para la definición de objetos de interfaz se utilizan enlaces, imágenes, etc.

La presente fase muestra los distintos prototipos de interfaz que serán visualizados por el usuario, así tener una idea clara del aspecto de las pantallas finales.

Figura 16

Diseño de interfaz no abstracta para pedidos de calzado.

Nota. Diseño de interfaz para dar de alta un pedido de calzado agregando el detalle de marca, estilo de calzado, numeración, etc. Paralelamente se observa el menú navegacional para el usuario autenticado. Elaboración propia

5.5 Implementación

La fase de implementación es el resultado del análisis y del diseño, es la responsable de traducir las anteriores fases y los alcances logrados hacia un ambiente de aplicación y generar así el código final. Esta se adapta al lenguaje de programación y/o base de datos.

Herramientas utilizadas

La aplicación web correrá dentro de un servidor web apache, el lenguaje que se utiliza es PHP que tiene soporte aceptable para sitios web dinámicos y el gestor de base de datos es MariaDB, de esta manera se maneja la política de software libre.

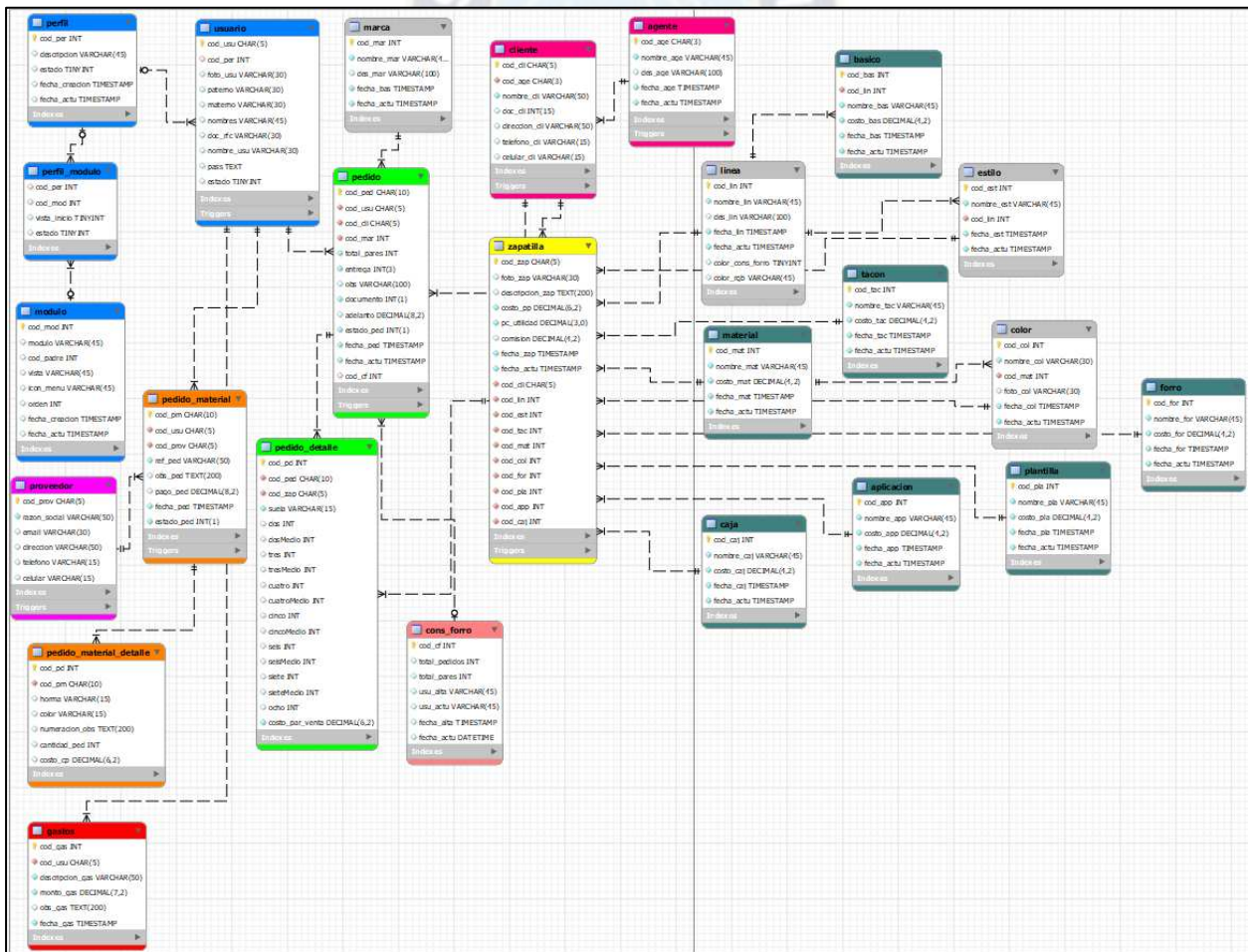
Como herramientas de programación y diseño se hace uso de Visual Studio Code para la programación y edición del código fuente del sistema, Photoshop para la creación y edición de imágenes.

5.5.1 Modelo de base de datos

Las clases conceptuales presentadas en la fase del modelo conceptual son las que dan origen a las tablas del modelo de datos relacional. En la figura se puede observar el modelo físico de la base de datos.

Figura 17

Modelo de base de datos del Sistema de Control de Producción y Administración de Calzado.



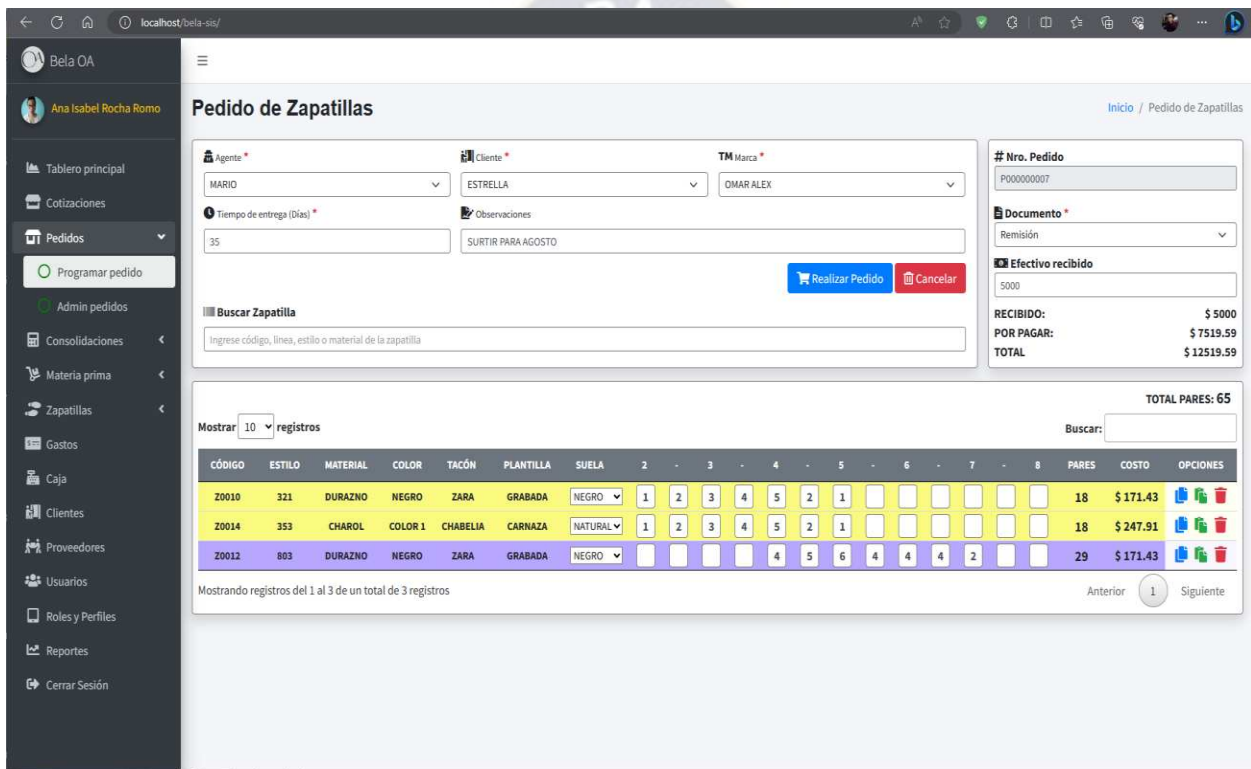
Nota. Diagrama físico del modelo relacional de base de datos. Elaboración propia

5.5.2 Interfaz de usuario

Al contar con distintos tipos de usuarios y estos a su vez tienen diferentes roles y perfiles dentro del sistema, el menú para un perfil A será distinto al menú del perfil B, a continuación, se muestra la diferencia que existe.

Figura 18

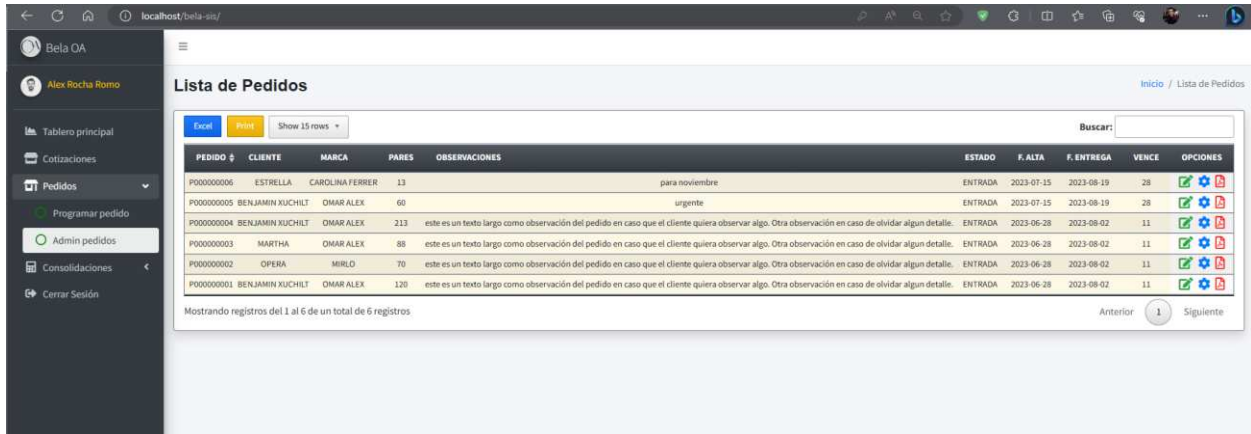
Interfaz gráfica del administrador.



Nota. Captura de la interfaz del Administrador al momento que se da de alta un pedido.
Elaboración propia.

Figura 19

Interfaz gráfica del director general.

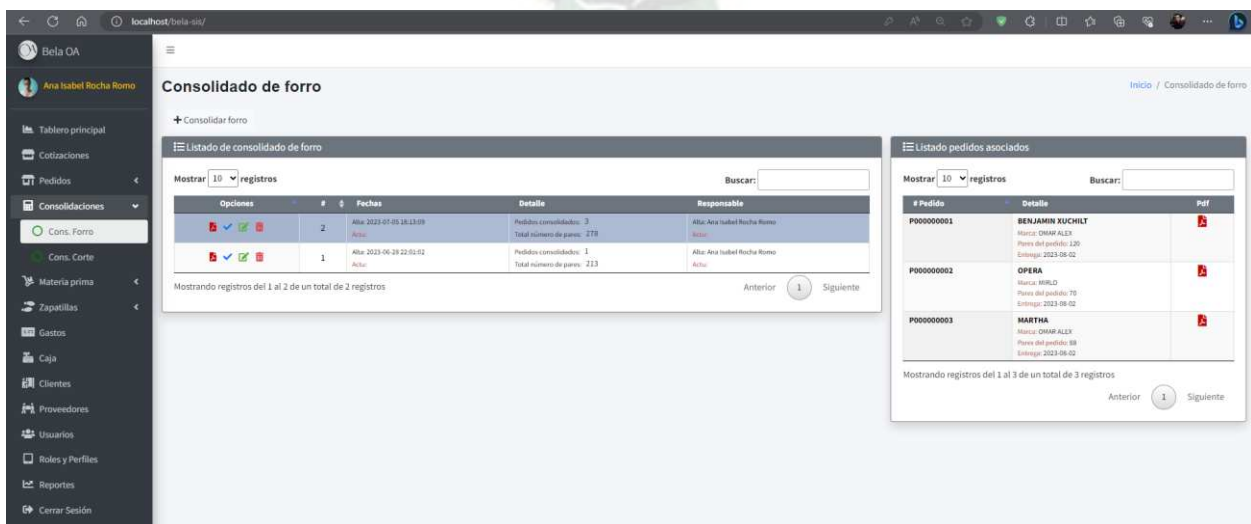


Nota. Captura de la interfaz del Director General al momento que visualiza los pedidos pedido. Elaboración propia.

A continuación, se muestran los módulos desarrollados para el cumplimiento de los objetivos trazados de este proyecto.

Figura 20

Interfaz gráfica del consolidado de forro.



Nota. Captura de la interfaz para la administración del consolidado de forro. Elaboración propia.

Figura 21

Reporte de consolidado de forro.

Consolidado: 2

OMAR ALEX
PARALLELO

CONSOLIDADO DE FORRO
P00000001 - P00000002 - P00000003

N	LINEA	PARES	COLOR	2	-	3	-	4	-	5	-	6	-	7	-	8
1	300	66	NUEZ	3	2	7	7	7	7	7	6	5	5	3	3	4
2	500	66	NUEZ	3	2	7	7	7	7	7	6	5	5	3	3	4
3	800	66	NUEZ	3	2	7	7	7	7	7	6	5	5	3	3	4
4	960	80	NEGRO	4	2	9	9	7	7	7	9	8	5	3	4	6
Total prs.:				278												

Nota. Captura del reporte de consolidado de forro. Elaboración propia.

Figura 22

Reporte de una hoja de pedido.

OMAR ALEX

HOJA DE PEDIDO

FECHA ALTA : 28/06/2023 CLIENTE : BENJAMIN KUCHILT
 FECHA ENTREGA : 02/08/2023 MARCA : OMAR ALEX PEDIDO : P00000001
 ID CLIENTE : C0003

#	ESTILO	PARES	MATERIAL	ORO ROSADO	TACON	PLATILLA	2	-	3	-	4	-	5	-	6	-	7	-	8	
1	321	30	DURAZNO	NEGRO	ZARA	GRABADA	2	2	2	2	4	4	4	4			3	3	2	2
2	500	30	DURAZNO	NEGRO	ZARA	GRABADA	2	2	2	2	4	4	4	4			3	3	2	2
3	803	30	DURAZNO	NEGRO	ZARA	GRABADA	2	2	2	2	4	4	4	4			3	3	2	2
4	954	30	DURAZNO	NEGRO	ZARA	GRABADA	2	2	2	2	4	4	4	4			3	3	2	2
Total prs.:		120																		

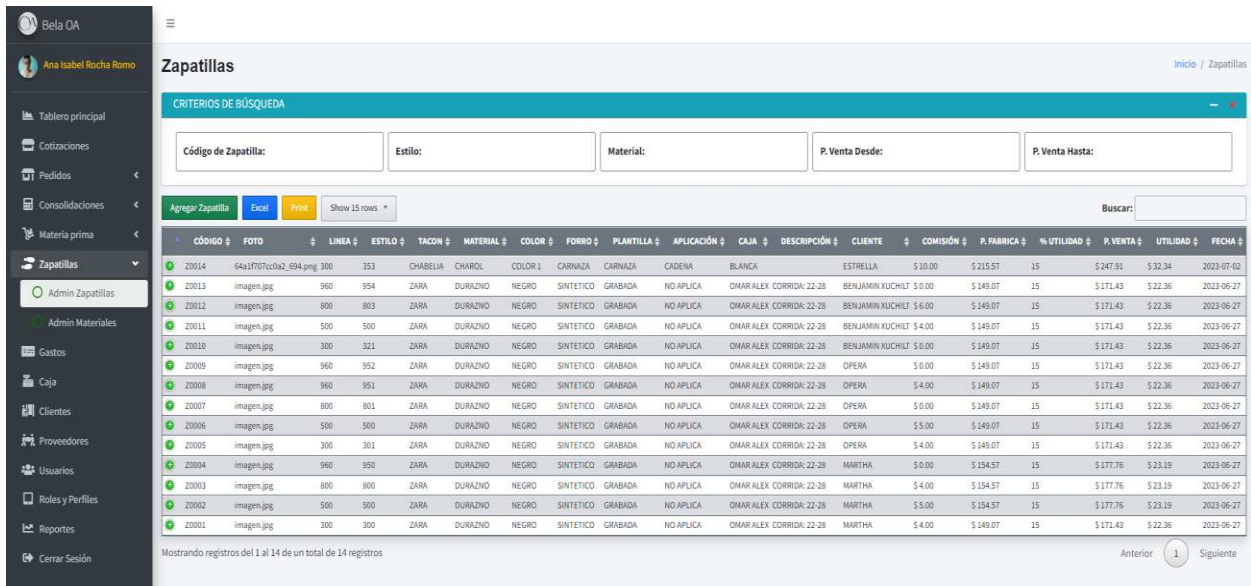
OBS.: este es un texto largo como observación del pedido en caso que el cliente quiera observar algo. Otra observación en caso de olvidar algun detalle.

Página 1 / 1

Nota. Captura del reporte de una hoja de pedido. Elaboración propia.

Figura 23

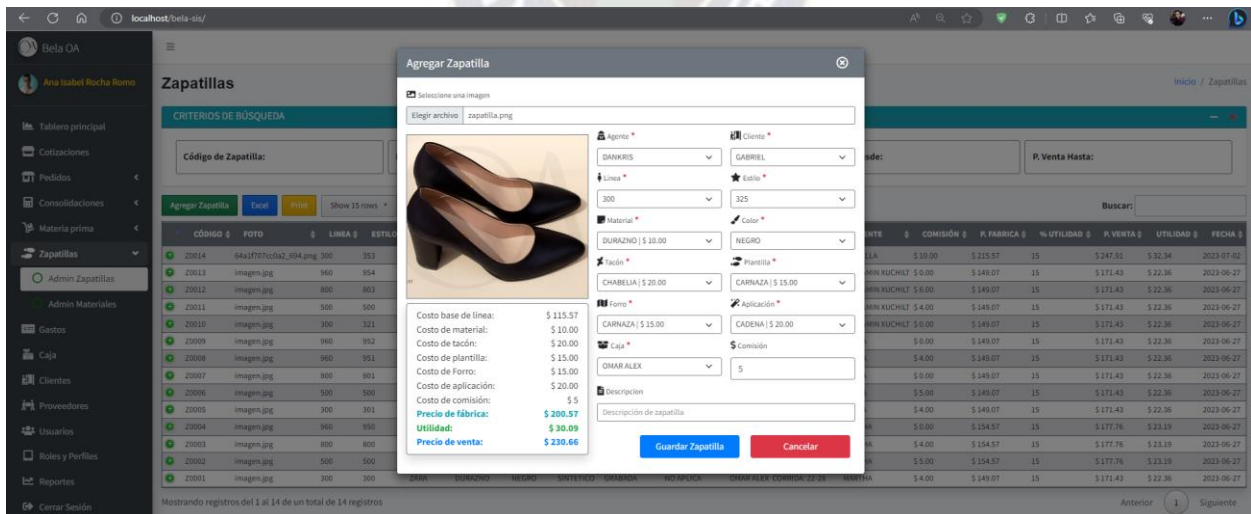
Interfaz gráfica de administración de zapatillas.



Nota. Captura de la interfaz para la administración de zapatillas. Elaboración propia.

Figura 24

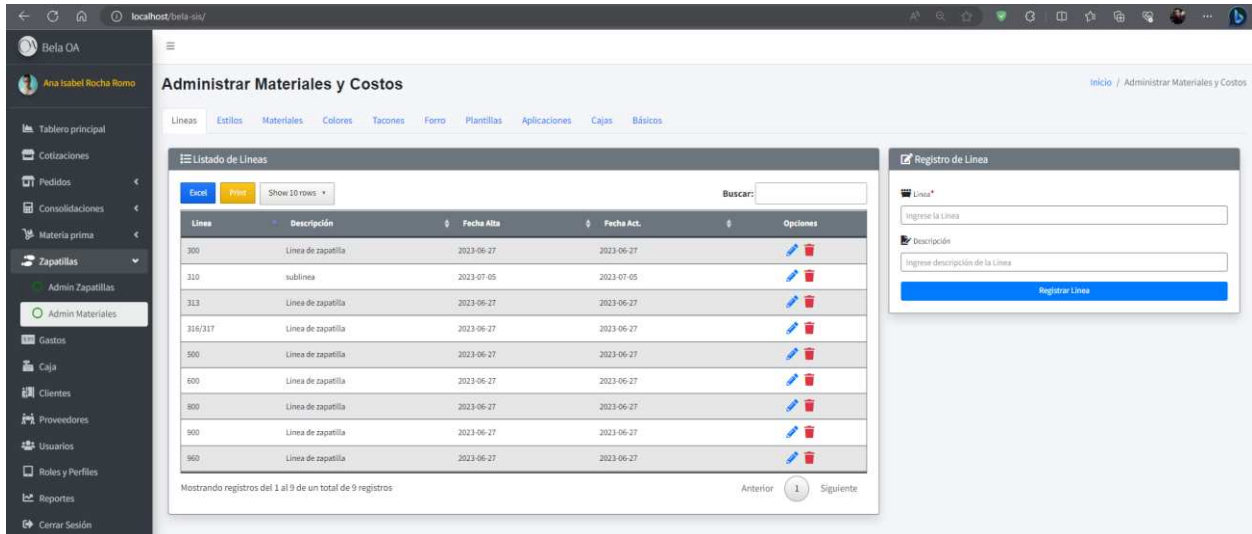
Interfaz gráfica para agregar una zapatilla.



Nota. Captura de la interfaz para agregar una nueva zapatilla. Elaboración propia.

Figura 25

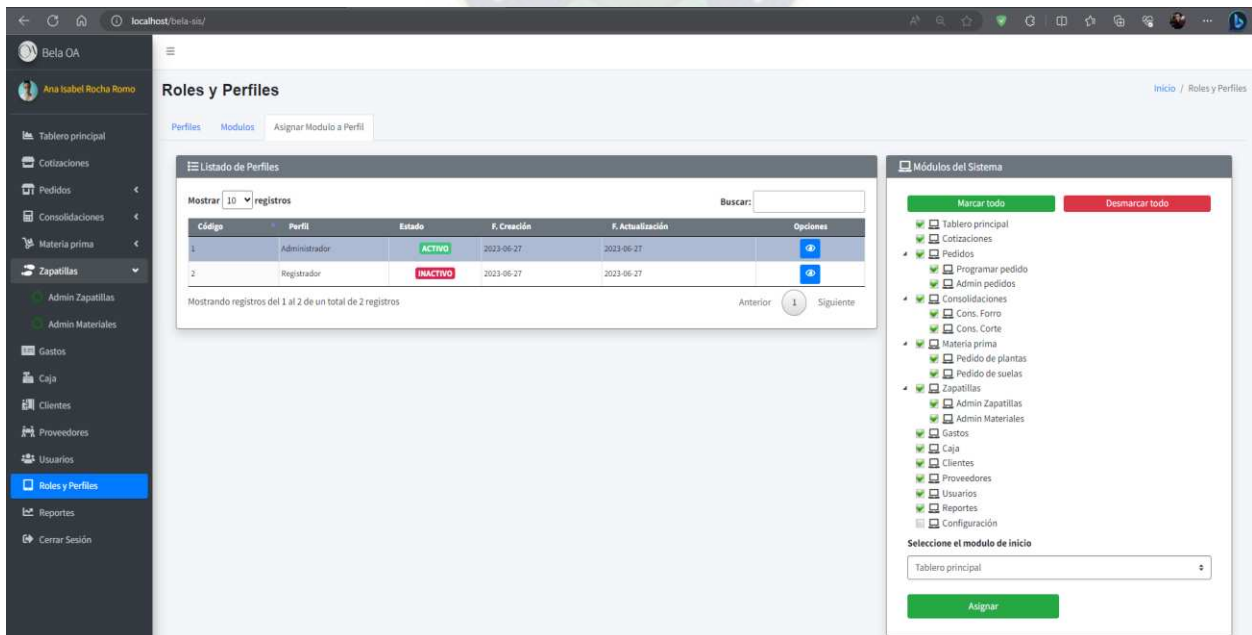
Interfaz gráfica para administrar materiales y costos.



Nota. Captura de la interfaz para la administración materiales y los costos de cada material empleado en la elaboración de un calzado. Elaboración propia.

Figura 26

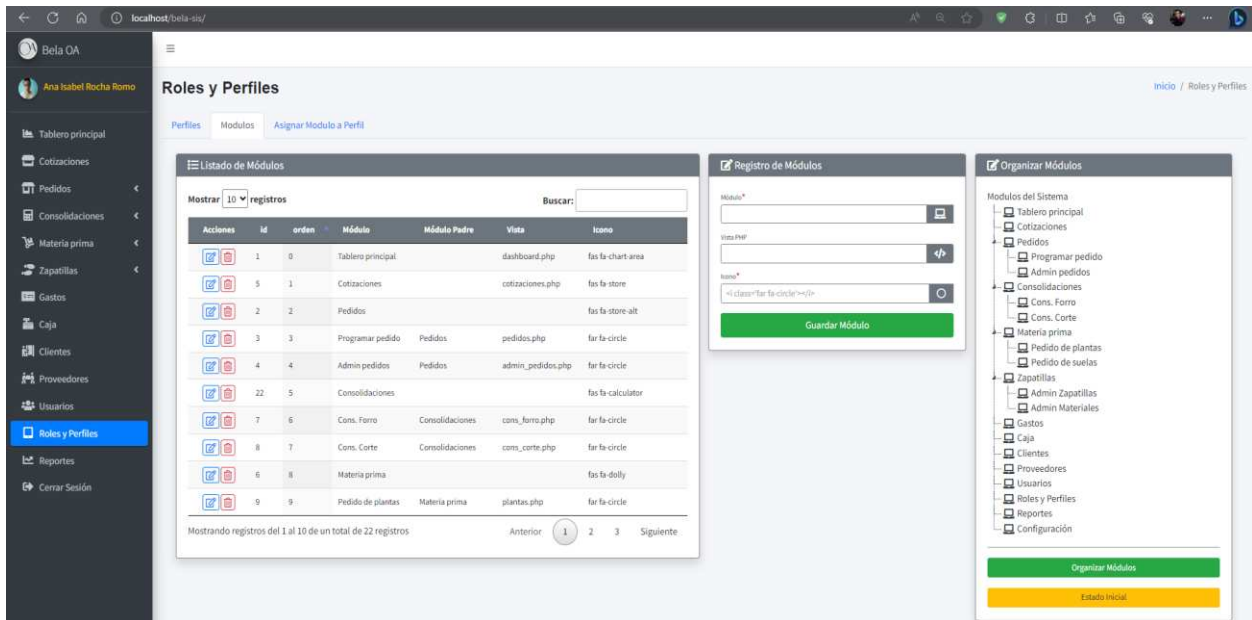
Interfaz gráfica para asignar módulos a un perfil.



Nota. Captura de la interfaz para asignar módulos a un perfil y dar permisos. Elaboración propia.

Figura 27

Interfaz gráfica para administrar módulos.



Nota. Captura de la interfaz para la administración de módulos. Elaboración propia.

6 Conclusión y Recomendaciones

6.1 Conclusiones

En conclusión, la implementación del Sistema de Control de Producción y Administrativo de Calzado en la fábrica Omar Alex ha sido un éxito en el logro de sus objetivos. El sistema ha contribuido de manera significativa al manejo eficiente de la información estratégica de la empresa, permitiendo una administración y seguimiento más efectivos de los productos.

En cuanto a los objetivos específicos, se ha logrado favorecer el control del sistema de calzado de manera excepcional, al punto de capacitar a personal sin conocimientos de programación para administrar el sistema de forma autónoma. La optimización del proceso de obtención de información ha brindado resultados notables, permitiendo acceder rápidamente a toda la información relevante sobre las actividades realizadas.

El desarrollo del módulo para el registro de pedidos y consolidaciones ha sido fundamental para agilizar y facilitar la gestión de los pedidos, permitiendo llevar un registro detallado y actualizado de cada pedido realizado por los clientes. Además, los módulos implementados para el seguimiento de la elaboración de cada pedido han permitido tener un control más preciso sobre el proceso productivo, garantizando tiempos de entrega adecuados y una mayor satisfacción del cliente.

La inclusión de compatibilidad con dispositivos móviles y diferentes plataformas mediante el Diseño Web Adaptable ha demostrado ser una decisión acertada, ya que ha permitido a los usuarios acceder al sistema desde cualquier lugar y en cualquier momento, aumentando la eficiencia y la flexibilidad en la administración de la fábrica de calzado.

En conjunto, la implementación exitosa de estos objetivos específicos ha generado un impacto positivo en la operatividad y productividad de la fábrica, mejorando la toma de decisiones estratégicas y contribuyendo al crecimiento sostenible del negocio. El Sistema de Control de Producción y Administración de Calzado ha demostrado ser una herramienta valiosa y confiable para optimizar los procesos internos y mantener una gestión eficiente de los productos, fortaleciendo así la posición competitiva de la empresa en el mercado.

6.2 Recomendaciones

Basándonos en los objetivos y conclusiones presentados, algunas recomendaciones para mejorar y mantener el Sistema de Control de Producción y Administración de Calzado en la fábrica Omar Alex:

- ✓ **Capacitación Continua:** Asegurarse de proporcionar capacitación y actualización periódica a todo el personal que interactúa con el sistema, incluyendo aquellos sin conocimientos de programación. Esto garantizará que todos los usuarios estén familiarizados con las funcionalidades del sistema y puedan utilizarlo de manera efectiva.
- ✓ **Monitoreo y Mantenimiento:** Implementar un programa de monitoreo y mantenimiento regular para asegurar el buen funcionamiento del sistema y prevenir posibles problemas técnicos o de rendimiento. Esto incluye actualizaciones de software, respaldos de datos y solución oportuna de cualquier incidencia.
- ✓ **Retroalimentación de Usuarios:** Fomentar un canal de comunicación para que los usuarios puedan proporcionar retroalimentación sobre el sistema. Esto permitirá identificar áreas de mejora, nuevas funcionalidades requeridas y posibles ajustes para adaptarse a las necesidades cambiantes del negocio.

- ✓ Seguridad y Privacidad: Garantizar la seguridad y privacidad de la información almacenada en el sistema. Implementar medidas de seguridad robustas para proteger los datos sensibles y restringir el acceso a la información solo a aquellos usuarios autorizados.
- ✓ Mejora Continua: Promover una cultura de mejora continua, donde se realicen análisis periódicos de la eficiencia del sistema y se busquen oportunidades para optimizar y expandir sus funcionalidades con el objetivo de mantenerse alineado con las necesidades del negocio y las tendencias tecnológicas.
- ✓ Soporte Técnico: Contar con un equipo de soporte técnico disponible para atender cualquier consulta o problema técnico que puedan enfrentar los usuarios. Brindar un servicio de atención al cliente eficiente y oportuno ayudará a mantener la confianza en el sistema.
- ✓ Adaptabilidad a Futuros Cambios: Anticipar futuras necesidades y cambios en el negocio para asegurar que el sistema pueda adaptarse y evolucionar junto con la empresa. Mantenerse actualizado con las tecnologías emergentes y las mejores prácticas de la industria.

Siguiendo estas recomendaciones, la fábrica Omar Alex podrá continuar aprovechando los beneficios del Sistema de Control de Producción y Administración de Calzado y mantener su posición competitiva en el mercado a través de una gestión eficiente y estratégica de sus productos.

7 Referencias bibliográficas

- A. P. A. (2019). *Publication Manual of the American Psychological Association (Séptima edición)* (7 ed.). American Psychological Association.
- Espejo, J. (2020). *3.2.5 Sistema web de administración y control del almacenamiento de artículos en el almacén central y en subalmacenes*. Universidad Mayor de San Andrés. Carrera de Informática.
- Fernández, Y., & Díaz, Y. (2012). *Patrón Modelo-Vista-Controlador*. Telematica.
- Gayón, J., & Ospina, L. (2019). *Desarrollo de un sistema de gestión de inventarios para el control de materias primas y productos terminados*. Universidad Libre - Facultad de Ingeniería.
- Guerra, D. (2022). *Diseño de un sistema de venta en Calzados Gorillaz*. Universidad Peruana de Las Américas.
- Hurtado, J., Molina, C., & Vega, M. (2018). *Diseño de un sistema de gestión de inventario de materia prima de la empresa fábrica de calzado ECCO de Nicaragua, S.A.* Universidad Nacional de Ingeniería - Facultad de Tecnología de la industria - Ingeniería Industrial.
- Kendall, K. E., & Kendall, J. E. (2011). *Análisis y Diseño de Sistemas* (8 ed.). PEARSON.
- Larman, C. (2004). *Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development* (3 ed.). Prentice Hall.
- Pressman, R. S. (2014). *Software Engineering: A Practitioner's Approach* (8 ed.). McGraw-Hill Education.
- Silberschatz, A., Korth, H. F., & Sudarshan, S. (2002). *Fundamentos de bases de datos* (4 ed.). McGraw-Hill Interamericana.

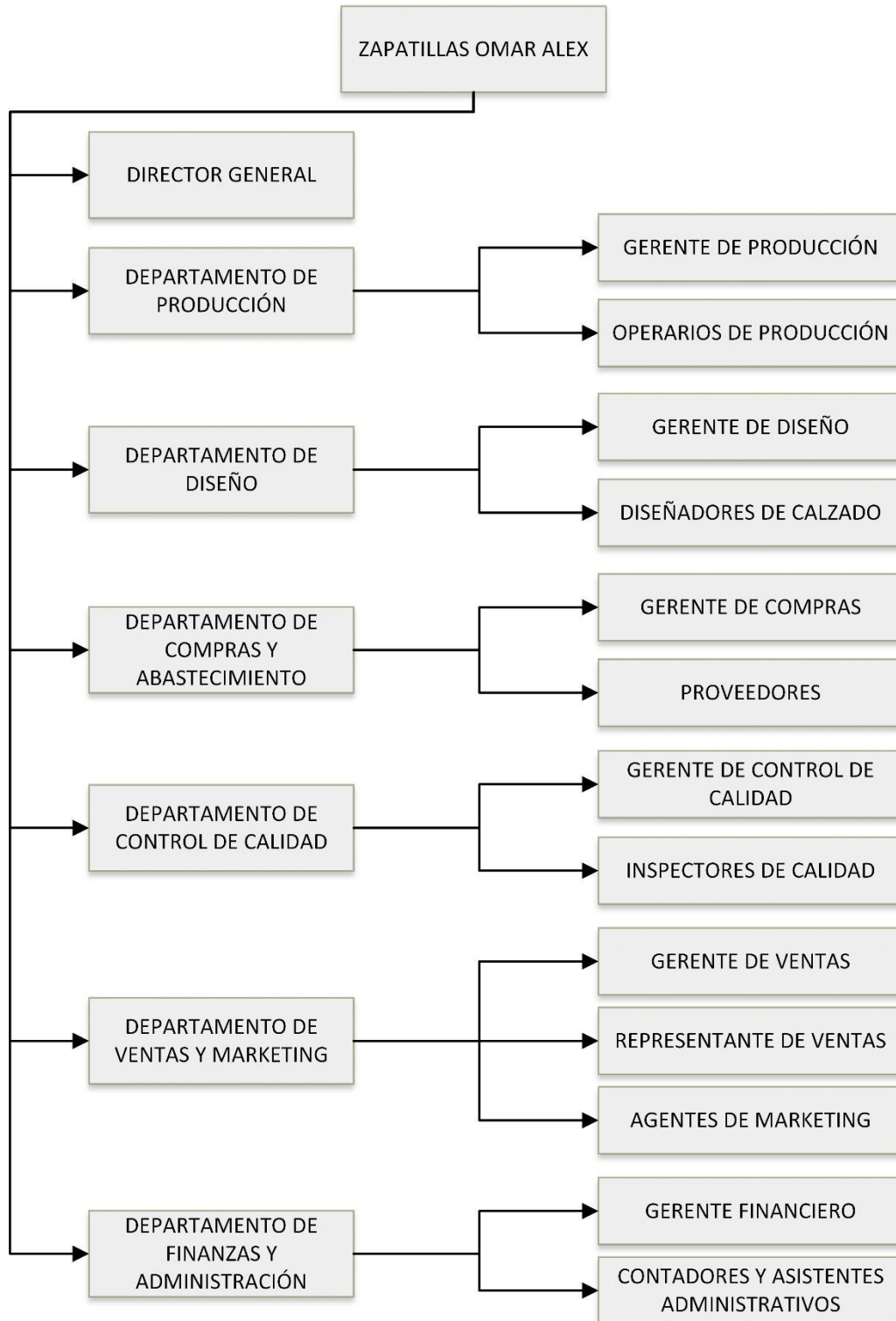
Society, I. C. (2014). *"IEEE Standard Glossary of Software Engineering Terminology" (IEEE Std 610.12-1990)*. Institute of Electrical and Electronics Engineers.

Sommerville, I. (2011). *Software Engineering* (9 ed.). Addison-Wesley.

Vasquez, I. (2020). *Gestión, control de materia prima, venta de productos e inventarios usando Supply Chain Management*. Universidad Mayor de San Andrés. Carrera de Informática.

ANEXOS

ORGANIGRAMA DE LA EMPRESA



Autor: Reynaldo Edwin Cuno Luque

Correo electrónico: recunoluque@gmail.com

Celular: 72050628



2023-TTES-1304-D-2

**DIRECCIÓN DE DERECHO DE AUTOR
Y DERECHOS CONEXOS**
RESOLUCIÓN ADMINISTRATIVA NRO. 1-3153/2023
La Paz, 25 de Octubre del 2023

VISTOS:

La solicitud de Inscripción de Derecho de Autor presentada en fecha **18 de Octubre del 2023**, por **REYNALDO EDWIN CUNO LUQUE** con C.I. N° **6798661 LP.**, con número de trámite **DA 1677/2023**, señala la pretensión de inscripción del Proyecto de Grado titulado: **"SISTEMA DE CONTROL DE PRODUCCIÓN Y ADMINISTRATIVO DE CALZADO"**, cuyos datos y antecedentes se encuentran adjuntos y expresados en el Formulario de Declaración Jurada.

CONSIDERANDO

Que, en observación al Artículo 4º del Decreto Supremo N° 27938 modificado parcialmente por el Decreto Supremo N° 28152 el *"Servicio Nacional de Propiedad Intelectual SENAPI, administra en forma desconcentrada e integral el régimen de la Propiedad Intelectual en todos sus componentes, mediante una estricta observancia de los regímenes legales de la Propiedad Intelectual, de la vigilancia de su cumplimiento y de una efectiva protección de los derechos de exclusiva referidos a la propiedad industrial, al derecho de autor y derechos conexos; constituyéndose en la oficina nacional competente respecto de los tratados internacionales y acuerdos regionales suscritos y adheridos por el país, así como de las normas y regímenes comunes que en materia de Propiedad Intelectual se han adoptado en el marco del proceso andino de integración"*.

Que, el Artículo 16º del Decreto Supremo N° 27938 establece *"Como núcleo técnico y operativo del SENAPI funcionan las Direcciones Técnicas que son las encargadas de la evaluación y procesamiento de las solicitudes de derechos de propiedad intelectual, de conformidad a los distintos regímenes legales aplicables a cada área de gestión"*. En ese marco, la Dirección de Derecho de Autor y Derechos Conexos otorga registros con carácter declarativo sobre las obras del ingenio cualquiera que sea el género o forma de expresión, sin importar el mérito literario o artístico a través de la inscripción y la difusión, en cumplimiento a la Decisión 351 Régimen Común sobre Derecho de Autor y Derechos Conexos de la Comunidad Andina, Ley de Derecho de Autor N° 1322, Decreto Reglamentario N° 23907 y demás normativa vigente sobre la materia.

Que, la solicitud presentada cumple con: el Artículo 6º de la Ley N° 1322 de Derecho de Autor, el Artículo 26º inciso a) del Decreto Supremo N° 23907 Reglamento de la Ley de Derecho de Autor, y con el Artículo 4º de la Decisión 351 Régimen Común sobre Derecho de Autor y Derechos Conexos de la Comunidad Andina.

Que, de conformidad al Artículo 18º de la Ley N° 1322 de Derecho de Autor en concordancia con el Artículo 18º de la Decisión 351 Régimen Común sobre Derecho de Autor y Derechos Conexos de la Comunidad Andina, referentes a la duración de los Derechos Patrimoniales, los mismos establecen que: *"la duración de la protección concedida por la presente ley será para toda la vida del autor y por 50 años después de su muerte, a favor de sus herederos, legatarios y cesionarios"*.



"2023 AÑO DE LA JUVENTUD HACIA EL BICENTENARIO"

Oficina Central - La Paz
Av. Montes, N° 515,
entre Esq. Uruguay y
C. Batallón Illimani.
Telfs.: 2195700
2195276 - 219251

Oficina - Santa Cruz
Av. Uruguay, Calle
prolongación Quijaro,
N° 29, Edif. Bicentenario.
Telfs.: 3121752 - 72042936

Oficina - Cochabamba
Calle Bolívar, N° 737,
entre 16 de Julio y Antezana.
Telfs.: 4144403 - 72042957

Oficina - El Alto
Av. Juan Pablo II, N° 2950
Edif. Multicentro El Ceibo
Ltda. Piso 2, Of. 58,
Zona 16 de Julio.
Telfs.: 2140001 - 72042029

Oficina - Chuquisaca
Calle Kilómetro 7, N° 366
casi esq. Uriolagoitia,
Zona Parque Bolívar.
Telf.: 72005873

Oficina - Tarija
Av. La Paz, entre
Calles Ciro Trigo y Avaroa
Edif. Santa Clara, N° 243.
Telf.: 72015286

Oficina - Oruro
Calle 6 de Octubre N° 5837
entre Ayacucho y Junín,
Galería Central, Of. 14.
Telf.: 67201288

Oficina - Potosí
Av. Villazón entre calles
Wenceslao Alba y San Alberto,
Edif. AM. Salinas N° 242,
Primer Piso, Of. 17.
Telf.: 72018160

Que, se deja establecido en conformidad al Artículo 4º de la Ley Nº 1322 de Derecho de Autor, y Artículo 7º de la Decisión 351 Régimen Común sobre Derecho de Autor y Derechos Conexos de la Comunidad Andina que: *"...No son objeto de protección las ideas contenidas en las obras literarias, artísticas, o el contenido ideológico o técnico de las obras científicas ni su aprovechamiento industrial o comercial"*.

Que, el artículo 4, inciso e) de la ley 2341 de Procedimiento Administrativo, instituye que: *"... en la relación de los particulares con la Administración Pública, se presume el principio de buena fe. La confianza, la cooperación y la lealtad en la actuación de los servidores públicos y de los ciudadanos ..."*, por lo que se presume la buena fe de los administrados respecto a las solicitudes de registro y la declaración jurada respecto a la originalidad de la obra.

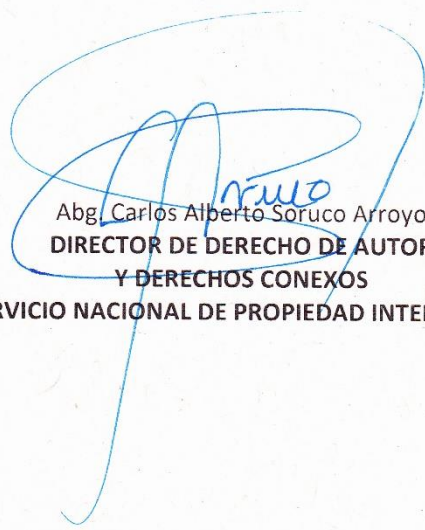
POR TANTO

El Director de Derecho de Autor y Derechos Conexos sin ingresar en mayores consideraciones de orden legal, en ejercicio de las atribuciones conferidas

RESUELVE:

INSCRIBIR en el Registro de Tesis, Proyectos de Grado, Monografías y Otras Similares de la Dirección de Derecho de Autor y Derechos Conexos, el Proyecto de Grado titulado: **"SISTEMA DE CONTROL DE PRODUCCIÓN Y ADMINISTRATIVO DE CALZADO"**, a favor del autor y titular: **REYNALDO EDWIN CUNO LUQUE** con **C.I. Nº 6798661 LP.**, quedando amparado su derecho conforme a Ley, salvando el mejor derecho que terceras personas pudieren demostrar.

Regístrese, Comuníquese y Archívese.



Abg. Carlos Alberto Soruco Arroyo
**DIRECTOR DE DERECHO DE AUTOR
Y DERECHOS CONEXOS**
SERVICIO NACIONAL DE PROPIEDAD INTELECTUAL



CASA/mxaq
c.c.Arch.



"2023 AÑO DE LA JUVENTUD HACIA EL BICENTENARIO"

Oficina Central - La Paz
Av. Montes, Nº 515,
entre Esq. Uruguay y
C. Batallón Illimani.
Telfs.: 2115700
2119276 - 2119251

Oficina - Santa Cruz
Av. Uruguay, Calle
prolongación Quijarro,
Nº 29, Edif. Bicentenario.
Telfs.: 3121752 - 72042936

Oficina - Cochabamba
Calle Bolívar, Nº 737,
entre 16 de Julio y Antezana.
Telfs.: 4141403 - 72042957

Oficina - El Alto
Av. Juan Pablo II, Nº 2560
Edif. Multicentro El Ceibo
Ltda. Piso 2, Of. 5B,
Zona 16 de Julio.
Telfs.: 2141001 - 72043029

Oficina - Chuquisaca
Calle Kilómetro 7, Nº 366
casi esq. Urriolagoitia,
Zona Parque Bolívar.
Telf.: 72015873

Oficina - Tarija
Av. La Paz, entre
Calles Ciro Trigo y Avaroa
Edif. Santa Clara, Nº 243.
Telf.: 72015286

Oficina - Oruro
Calle 6 de Octubre Nº 5837
entre Ayacucho y Junín,
Galería Central, Of. 14.
Telf.: 67201288

Oficina - Potosí
Av. Villazón entre calles
Wenceslao Alba y San Alberto,
Edif. AM. Salinas Nº 242,
Primer Piso, Of. 17.
Telf.: 72018160