

**UNIVERSIDAD CENTRAL  
VICERRECTORÍA ACADÉMICA**

**ESCUELA DE INGENIERÍA INFORMÁTICA**

**Desarrollo de un Sistema para la Gestión y el Control de  
Inventario, para una Microempresa**

**MODALIDAD DE TESIS PARA OPTAR POR EL GRADO DE BACHILLERATO EN  
INGENIERÍA INFORMÁTICA**

**ESTUDIANTE: SOFÍA YULIANA GAMBOA NAVARRO**

**TUTOR: ING. MAURICIO TORRES**

**SEDE CENTRAL SAN JOSÉ, COSTA RICA**

**DICIEMBRE, 2024**

## Tabla de Contenido

Tabla de Contenido .....	ii
Dedicatoria y agradecimiento .....	xx
Resumen.....	xxi
Abstract .....	xxii
Capítulo I. Problema .....	1
Planteamiento del Problema .....	2
Objetivos .....	2
Objetivo General.....	2
Objetivos Específicos.....	2
Justificación .....	3
Antecedentes .....	4
Antecedentes Internacionales.....	4
Antecedentes Nacionales .....	6
Proyecciones .....	8
Alcances .....	8
Limitaciones.....	9
Capítulo II. Marco Teórico .....	10
Historia de la Programación.....	11

Primeros lenguajes de programación .....	11
Fundamentos de la Programación.....	15
¿Qué es un lenguaje de programación? .....	15
Tipos de Programación .....	15
Tipos de Lenguajes de Programación.....	17
Paradigmas de Programación.....	21
Algoritmos de Programación .....	25
Sistema de Inventarios .....	26
¿Qué son los Sistemas de Inventarios? .....	27
¿Qué son los Métodos Para el Control de Inventarios? .....	27
Tipos de Inventarios.....	29
Sistemas de Información.....	32
Sistemas de Procesamiento de Transacciones .....	34
Sistemas de Información Gerencial .....	34
Sistemas de Control de Procesos de Negocio.....	35
Sistemas de Información de Marketing .....	36
Sistemas de Colaboración Empresarial.....	37
Sistema de Soporte de Toma de Decisiones .....	37
Sistemas de Información Ejecutiva.....	38
El Ciclo de Vida de un Sistema de Información.....	39

Planificación .....	39
Análisis de Requerimientos .....	39
Diseño .....	40
Desarrollo.....	40
Prueba .....	40
Integración y Ejecución .....	40
Operación y Mantenimiento .....	40
Requerimientos de Sistema.....	41
Requerimientos Funcionales .....	41
Requerimientos no Funcionales .....	41
Diseño Integral de Sistemas.....	42
Arquitectura .....	42
Estructura de Datos .....	43
Interface .....	44
Componentes.....	45
Diagrama UML.....	45
Casos de Uso.....	46
Base de Datos.....	46
Historia de las Bases de Datos .....	46
Qué son las Bases de datos de Código Abierto (Open Source) .....	47

Metodologías de Desarrollo.....	50
Metodologías Tradicionales.....	50
Cascada .....	50
Incremental .....	50
Prototipo.....	51
Metodologías Agiles .....	51
SCRUM.....	51
Kanban .....	51
Programación Extrema (Extreme programming, XP) .....	52
¿Qué es el Desarrollo Web?.....	52
¿Qué Hace un Desarrollador Web?.....	53
Tipos de Desarrollo web .....	54
Front-end (interfaz).....	54
Backend (servidor).....	55
Full Stack .....	56
Ventajas y Desventajas del Desarrollo Web.....	57
Ventajas.....	57
Desventajas .....	58
Herramientas Para Desarrollo del Proyecto .....	58
IDE.....	58

Editores de Código.....	58
Visual Studio Code .....	59
Java .....	60
Spring Boot .....	61
Thymeleaf .....	62
HTML .....	62
CSS .....	62
JavaScript.....	63
Bootstrap.....	63
Capítulo III. Marco Metodológico.....	65
¿Qué es un Marco Metodológico? .....	66
Importancia de un Marco Metodológico.....	67
Desarrollo de un Marco Metodológico.....	69
Definición de la Investigación .....	69
Revisión de la Literatura.....	69
Diseño de Investigación .....	69
Población y Muestra .....	69
Instrumentos de Recopilación de Datos.....	69
Procedimiento .....	70
Análisis de Datos .....	70

Aspectos Éticos .....	70
Plan de Trabajo .....	70
Presupuesto .....	70
Validación del Marco Metodológico .....	70
Redacción del Marco Metodológico.....	70
Revisión Continua.....	71
Enfoque de la Investigación.....	71
Tipos de Enfoques.....	72
Enfoque Cualitativo .....	72
Tipos de Investigación Cualitativa.....	73
Enfoque Cuantitativo .....	74
Tipos de Investigación Cuantitativa.....	76
Enfoque Mixto .....	76
Tipos de Investigación Mixta.....	78
Método de Investigación.....	78
Fuentes de Información.....	78
Fuentes Primarias .....	79
Fuentes Secundarias .....	79
Fuentes Terciarias .....	80
Otros Tipos de Fuente de Información .....	80

Local, Nacional o Internacional .....	81
Textual o Digital .....	81
General o Especializada .....	81
Formales o Informales .....	81
¿Qué es una Variable? .....	83
Requisitos Funcionales .....	84
Tipos de Requisitos Funcionales .....	85
Requerimientos no Funcionales .....	85
Tipos de Requisitos No Funcionales.....	86
Ventajas Requisitos No Funcionales .....	86
Desventajas de los Requisitos No Funcionales.....	86
Ejemplos de Requisitos No Funcionales.....	87
Plan de Pruebas .....	88
¿Cómo se Realiza un Plan de Pruebas? .....	88
Tipos de Pruebas de Software.....	88
Casos de uso.....	89
Aplicaciones de Casos de Uso .....	90
Componentes del Diagrama de Caso de uso.....	91
¿Cómo hacer un diagrama de caso de uso? .....	91
Instrumentos.....	92

Capítulo IV. Análisis de Resultados .....	93
Plan de Gestión de Riesgos.....	94
Componentes Principales.....	95
Tareas de la gestión de riesgos.....	96
Matriz de riesgos.....	97
Tipos de riesgos: .....	97
Estudio de Factibilidad .....	98
Análisis de factibilidad .....	99
Recursos tecnológicos.....	100
Optimización de recursos.....	101
Gestión del proyecto .....	102
Metodología Utilizada .....	104
Capítulo V. Conclusiones y Recomendaciones .....	107
Conclusiones .....	108
Recomendaciones .....	109
Capítulo VI. Análisis de Requerimientos .....	110
Análisis de Requerimientos .....	111
Requerimientos de usuario.....	111
Requerimientos Funcionales.....	119
Requerimientos No Funcionales .....	131

Modelos UML.....	133
Casos de Uso.....	133
Caso de Uso 1. Login.....	133
Caso de Uso 2. Registro de Usuario .....	135
Caso de Uso 3. Registro de Productos .....	137
Caso de Uso 4. Registro de Proveedores .....	139
Caso de Uso 5. Registro de Clientes .....	141
Caso de Uso 6. Registro de Categorías .....	143
Caso de Uso 7. Registro de Pedidos .....	145
Caso de Uso 8. Registro de Inventarios .....	147
Caso de Uso 9. Registro de Detalles Pedidos .....	149
Caso de Uso 10. Registro de suministros.....	151
Caso de Uso 11. Control de Bitácora Sesiones.....	153
Caso de Uso 12. Dashboard .....	155
Diagramas de flujo.....	157
Login.....	157
Menú .....	158
Registro Usuario .....	159
Mantenimiento .....	160
Diagramas de secuencia.....	161

Login.....	161
Menú Principal.....	162
Mantenimientos y Búsqueda.....	163
Datos Dashboard.....	173
Modelo Entidad Relación .....	174
Diccionario de datos .....	175
Pantalla del Login .....	180
Pantalla de ingreso al sistema .....	181
Pantallas de Categoría.....	182
Pantallas de Pedidos.....	184
Pantallas de Detalles de los Pedidos .....	185
Pantallas de Clientes .....	187
Pantallas de Productos .....	188
Pantallas de Suministros .....	190
Pantallas de Inventario.....	191
Pantallas de Proveedores.....	193
Pantallas de Usuarios .....	194
Pantallas de Bitácora Sesión .....	196
Referencias Bibliográficas .....	197

## Índice de Figuras

Figura 1 <i>Historia de la programación</i> .....	12
Figura 2 <i>Paradigmas de la programación</i> .....	13
Figura 3 <i>Mapa conceptual fundamentos de programación</i> .....	15
Figura 4 <i>Lenguaje de programación</i> .....	17
Figura 5 <i>Lenguajes de alto nivel vs de bajo nivel</i> .....	18
Figura 6 <i>Lenguajes tipados estático</i> .....	19
Figura 7 <i>Lenguajes tipados dinámicos</i> .....	20
Figura 8 <i>Lenguajes de programación</i> .....	21
Figura 9 <i>¿Qué son los paradigmas de programación?</i> .....	22
Figura 10 <i>Programación orientada a objetos</i> .....	23
Figura 11 <i>Programación funcional</i> .....	24
Figura 12 <i>Programación imperativa (PHP)</i> .....	24
Figura 13 <i>Programación declarativa</i> .....	25
Figura 14 <i>Los algoritmos en la programación</i> .....	26
Figura 15 <i>Sistemas de inventarios</i> .....	27
Figura 16 <i>Métodos para el Control de Inventarios</i> .....	29
Figura 17 <i>Sistemas de Información</i> .....	33
Figura 18 <i>Ciclo de vida de un sistema</i> .....	41
Figura 19 <i>Requerimientos del sistema</i> .....	42
Figura 20 <i>Diagrama UML</i> .....	45
Figura 21 <i>Casos de uso</i> .....	46

Figura 22 <i>Base de datos</i> .....	47
Figura 23 <i>Que es MySQL</i> .....	48
Figura 24 <i>Bases de datos open source</i> .....	50
Figura 25 <i>Metodologías ágiles y tradicional</i> .....	52
Figura 26 <i>Que Hace un Desarrollador web</i> .....	54
Figura 27 <i>¿Qué es frontend y backend?</i> .....	57
Figura 28 <i>Editor vs IDE</i> .....	59
Figura 29 <i>Visual Studio Code</i> .....	60
Figura 30 <i>Java</i> .....	61
Figura 31 <i>spring boot</i> .....	61
Figura 32 <i>Thymeleaf</i> .....	62
Figura 33 <i>HTML, CSS, JS</i> .....	63
Figura 34 <i>Bootstrap</i> .....	64
Figura 35 <i>Marco Metodológico</i> .....	67
Figura 36 <i>Enfoques de la Investigación</i> .....	72
Figura 37 <i>Métodos de Investigación Cualitativa</i> .....	73
Figura 38 <i>Proceso de Investigación Cuantitativa</i> .....	75
Figura 39 <i>Investigación Mixta</i> .....	77
Figura 40 <i>Fuentes de Información</i> .....	80
Figura 41 <i>Cuadro de Variables</i> .....	84
Figura 42 <i>Ejemplo Casos de Uso Registrar Cliente</i> .....	90
Figura 43 <i>Diagrama de Casos de Usos</i> .....	92
Figura 44 <i>Gestión de Riesgos</i> .....	94

Figura 45 <i>Matriz de Riesgos</i> .....	98
Figura 46 <i>Estudio de Factibilidad</i> .....	99
Figura 47 <i>Tipos de Factibilidad</i> .....	104
Figura 48 <i>Sistema Para la Gestión y el Control de Inventario Kanban</i> .....	105
Figura 49 <i>Tarea Terminadas Sistema Para la Gestión y el Control de Inventario Kanban</i> .....	106
Figura 50 <i>Diagrama de caso de uso - Login</i> .....	133
Figura 51 <i>Diagrama de caso de uso - Registro Usuario</i> .....	135
Figura 52 <i>Diagrama de caso de uso - Registro Producto</i> .....	137
Figura 53 <i>Diagrama de caso de uso - Registro Proveedores</i> .....	139
Figura 54 <i>Diagrama de caso de uso - Registro Clientes</i> .....	141
Figura 55 <i>Diagrama de caso de uso - Registro Categorías</i> .....	143
Figura 56 <i>Diagrama de caso de uso - Registro Pedidos</i> .....	145
Figura 57 <i>Diagrama de caso de uso - Registro Inventarios</i> .....	147
Figura 58 <i>Diagrama de caso de uso - Registro Detalles Pedidos</i> .....	149
Figura 59 <i>Diagrama de caso de uso - Registro Suministro</i> .....	151
Figura 60 <i>Diagrama de caso de uso - Control de Bitácora Sesión</i> .....	153
Figura 61 <i>Diagrama de caso de uso - Control de Datos Dashboard</i> .....	155
Figura 62 <i>Diagrama de Flujo Login</i> .....	157
Figura 63 <i>Diagrama de Flujo Menú</i> .....	158
Figura 64 <i>Diagrama de Flujo Registro Usuario</i> .....	159
Figura 65 <i>Diagrama de Flujo Mantenimiento</i> .....	160
Figura 66 <i>Diagrama de secuencia Login</i> .....	161

Figura 67 <i>Diagrama de Secuencia Menú Principal</i> .....	162
Figura 68 <i>Diagrama de Secuencia Mantenimiento Registro Usuarios</i> .....	163
Figura 69 <i>Diagrama de Secuencia Mantenimiento de Categorías</i> .....	164
Figura 70 <i>Diagrama de Secuencia Mantenimiento Pedidos</i> .....	165
Figura 71 <i>Diagrama de Secuencia Mantenimiento Detalles Pedidos</i> .....	166
Figura 72 <i>Diagrama de Secuencia Mantenimiento Clientes</i> .....	167
Figura 73 <i>Diagrama de Secuencia Mantenimiento Productos</i> .....	168
Figura 74 <i>Diagrama de Secuencia Mantenimiento Inventario</i> .....	169
Figura 75 <i>Diagrama de Secuencia Mantenimiento Suministros</i> .....	170
Figura 76 <i>Diagrama de Secuencia Mantenimiento Proveedores</i> .....	171
Figura 77 <i>Diagrama de Secuencia Bitácora Sesión</i> .....	172
Figura 78 <i>Diagrama de Secuencia Dashboard</i> .....	173
Figura 79 <i>Modelo Relacional</i> .....	174
Figura 80 <i>Login</i> .....	180
Figura 81 <i>Pantalla Principal</i> .....	181
Figura 82 <i>Pantalla Listar Categorías</i> .....	182
Figura 83 <i>Pantalla Agregar Categorías</i> .....	183
Figura 84 <i>Pantalla Actualizar Categorías</i> .....	183
Figura 85 <i>Pantalla Listar pedidos</i> .....	184
Figura 86 <i>Pantalla Agregar Pedidos</i> .....	184
Figura 87 <i>Pantalla Actualizar Pedidos</i> .....	185
Figura 88 <i>Pantalla Listar Detalles pedidos</i> .....	185
Figura 89 <i>Pantalla Agregar Detalles pedidos</i> .....	186

Figura 90 <i>Pantalla Actualizar Detalles pedidos</i> .....	186
Figura 91 <i>Pantalla Listar Clientes</i> .....	187
Figura 92 <i>Pantalla Agregar Clientes</i> .....	187
Figura 93 <i>Pantalla Actualizar Clientes</i> .....	188
Figura 94 <i>Pantalla Listar Productos</i> .....	188
Figura 95 <i>Pantalla Agregar Productos</i> .....	189
Figura 96 <i>Pantalla Actualizar Productos</i> .....	189
Figura 97 <i>Pantalla Listar Suministros</i> .....	190
Figura 98 <i>Pantalla Agregar Suministros</i> .....	190
Figura 99 <i>Pantalla Actualizar Suministros</i> .....	191
Figura 100 <i>Pantalla Listar Inventario</i> .....	191
Figura 101 <i>Pantalla Agregar Inventario</i> .....	192
Figura 102 <i>Pantalla Actualizar Inventario</i> .....	192
Figura 103 <i>Pantalla Listar Proveedores</i> .....	193
Figura 104 <i>Pantalla Agregar Proveedores</i> .....	193
Figura 105 <i>Pantalla Actualizar Proveedores</i> .....	194
Figura 106 <i>Pantalla Listar Usuarios</i> .....	194
Figura 107 <i>Pantalla Agregar Usuarios</i> .....	195
Figura 108 <i>Pantalla Actualizar Usuarios</i> .....	195
Figura 109 <i>Pantalla Vista Bitácora Sesión</i> .....	196
Figura 110 <i>Pantalla Reporte de Bitácora Sesión</i> .....	196

**Índice de Tablas**

Tabla 1 <i>Recursos del Equipo</i> .....	100
Tabla 2 <i>Costo de Proyecto</i> .....	101
Tabla 3 <i>Costos del Proyecto</i> .....	101
Tabla 4 <i>Requerimiento de usuario RUM01</i> .....	112
Tabla 5 <i>Requerimiento de usuario RUM02</i> .....	112
Tabla 6 <i>Requerimiento de usuario RUM03</i> .....	113
Tabla 7 <i>Requerimiento de usuario RUM04</i> .....	113
Tabla 8 <i>Requerimiento de usuario RUM05</i> .....	114
Tabla 9 <i>Requerimiento de usuario RUM06</i> .....	114
Tabla 10 <i>Requerimiento de usuario RUM07</i> .....	115
Tabla 11 <i>Requerimiento de usuario RUM08</i> .....	115
Tabla 12 <i>Requerimiento de usuario RUM09</i> .....	116
Tabla 13 <i>Requerimiento de usuario RUM10</i> .....	116
Tabla 14 <i>Requerimiento de usuario RUM11</i> .....	117
Tabla 15 <i>Requerimiento de usuario RUM12</i> .....	117
Tabla 16 <i>Requerimiento de usuario RUM14</i> .....	118
Tabla 17 <i>Requerimiento funcional RFM01</i> .....	119
Tabla 18 <i>Requerimiento funcional RFM02</i> .....	120
Tabla 19 <i>Requerimiento funcional RFM03</i> .....	121
Tabla 20 <i>Requerimiento funcional RFM04</i> .....	122
Tabla 21 <i>Requerimiento funcional RFM05</i> .....	123
Tabla 22 <i>Requerimiento funcional RFM06</i> .....	124

Tabla 23 <i>Requerimiento funcional RFM07</i> .....	125
Tabla 24 <i>Requerimiento funcional RFM08</i> .....	126
Tabla 25 <i>Requerimiento funcional RFM09</i> .....	127
Tabla 26 <i>Requerimiento funcional RFM10</i> .....	128
Tabla 27 <i>Requerimiento funcional RFM11</i> .....	129
Tabla 28 <i>Requerimiento funcional RFM13</i> .....	130
Tabla 29 <i>Requerimiento no Funcionales del Sistema</i> .....	131
Tabla 30 <i>Requerimiento no funcional RNFM001</i> .....	131
Tabla 31 <i>Requerimiento no funcional RNFM002</i> .....	132
Tabla 32 <i>Requerimiento no funcional RNMF003</i> .....	132
Tabla 33 <i>Login</i> .....	134
Tabla 34 <i>Registro Usuario</i> .....	136
Tabla 35 <i>Registro Productos</i> .....	138
Tabla 36 <i>Registro Proveedores</i> .....	140
Tabla 37 <i>Registro Clientes</i> .....	142
Tabla 38 <i>Registro Categorías</i> .....	144
Tabla 39 <i>Registro Pedidos</i> .....	146
Tabla 40 <i>Registro Inventarios</i> .....	148
Tabla 41 <i>Registro Detalles Pedidos</i> .....	150
Tabla 42 <i>Registro Suministro</i> .....	152
Tabla 43 <i>Control de Bitácora Sesión</i> .....	154
Tabla 44 <i>Control de Datos Dashboard</i> .....	156
Tabla 45 <i>Tabla Categorías</i> .....	175

Tabla 46 <i>Tabla Proveedores</i> .....	175
Tabla 47 <i>Tabla Productos</i> .....	176
Tabla 48 <i>Tabla Inventarios</i> .....	176
Tabla 49 <i>Tabla Clientes</i> .....	177
Tabla 50 <i>Tabla Pedidos</i> .....	177
Tabla 51 <i>Tabla Detalles de Pedidos</i> .....	178
Tabla 52 <i>Tabla Suministros</i> .....	178
Tabla 53 <i>Tabla Usuarios</i> .....	179
Tabla 54 <i>Tabla Bitácora de Sesión</i> .....	179

### **Dedicatoria y agradecimiento**

Le dedico este trabajo a las personas más importantes de mi vida, a mis padres José Enrique Gamboa Barrantes y Roció Navarro Blanco, y a mis hermanas mayores, Estefany Gamboa y Graciela Gamboa. Ustedes siempre han estado presentes en todas las etapas de mi vida, brindándome su apoyo y amor incondicional.

A mis abuelos y a mi tía que, a pesar de que ya no están con nosotros, sé que desde el Cielo continúan cuidándonos.

Gracias por el apoyo constante que me han brindado durante toda mi vida, a las incontables noches que pasaron en el hospital cuidando de mí cuando era una niña y que, a pesar de todos mis problemas de salud, ustedes siempre han sido los motores que mueven mi vida, cuidándome y alentándome para que siga avanzando. A mi mamá, por haber sido mi primera maestra cuando ya no puede ir más a la escuela, me enseñaste a leer y escribir. A mi papá, que siempre ha estado presente y, a pesar de estar tan lejos, nunca nos has dejado de dar tu amor y apoyo, les agradezco de corazón a toda mi familia porque sin ustedes no habría llegado hasta aquí.

## Resumen

Este trabajo de investigación se centra en el desarrollo de un sistema de control y gestión de inventarios para la microempresa Productos de Concreto Daniel, ubicada en San Marcos de Tarrazú, enfocándose en la transformación digital de sus procesos administrativos. La finalidad principal radica en establecer metodologías estructuradas que permitan superar las actuales deficiencias en el manejo de información sobre productos y suministros.

Durante la fase investigativa se identificaron diversas oportunidades de mejora en los procedimientos actuales: entregas extraviadas, imprecisiones en el procesamiento de pedidos y deficiencias en la documentación sistemática del inventario. Para abordar esta problemática se propone implementar una arquitectura informática fundamentada en tecnologías de código abierto como Java, MySQL y Visual Studio Code, incorporando módulos específicos para categorías, proveedores, inventario, productos, clientes y gestión de pedidos.

La metodología contempla un riguroso protocolo de validación mediante pruebas exhaustivas del prototipo funcional. El sistema facilitará operaciones esenciales como registro, edición y eliminación de datos relacionados con el inventario. Adicionalmente, incorpora un sistema de autenticación basado en correo electrónico y una estructura de roles de usuario para garantizar la seguridad de la información.

Los resultados preliminares sugieren que la implementación del sistema propuesto podría optimizar significativamente los procesos operativos de Productos de Concreto Daniel, permitiendo un control más preciso sobre el manejo de inventarios y facilitando la toma de decisiones, basada en información actualizada y confiable.

## **Abstract**

This research work focuses on the development and implementation of an inventory control and management system for the microenterprise Productos de Concreto Daniel, located in San Marcos de Tarrazú, focusing on the digital transformation of its administrative processes. The main purpose lies in establishing structured methodologies that allow overcoming the current deficiencies in the management of information on products and supplies.

During the investigative phase, several opportunities for improvement in current procedures were identified: lost deliveries, inaccuracies in order processing, and deficiencies in systematic inventory documentation. To address this problem, it is proposed to implement a computer architecture based on open-source technologies such as Java, MySQL and Visual Studio Code, incorporating specific modules for categories, suppliers, inventory, products, customers and order management.

The methodology includes a rigorous validation protocol through exhaustive testing of the functional prototype. The system will facilitate essential operations such as recording, editing and deleting inventory-related data. Additionally, it incorporates an email-based authentication system and a user role structure to guarantee information security.

Preliminary results suggest that the implementation of the proposed system could significantly optimize the operational processes of Productos de Concreto Daniel, allowing more precise control over inventory management and facilitating decision making based on updated and reliable information.

**Capítulo I. Problema**

## **Planteamiento del Problema**

La microempresa Productos de Concreto Daniel en la actualidad está ubicada en la provincia de San José, en el cantón de Tarrazú, distrito de San Marcos. Enfrenta un serio problema con la gestión de datos de productos y suministro. La falta de un sistema efectivo para organizar el inventario de productos y materiales impide que las operaciones diarias se realicen correctamente, lo que genera ineficiencia.

Esta investigación presenta una solución práctica y comprobada que aborda problemas específicos como entregas perdidas, pedidos incorrectos, productos no entregados y la falta de información actualizada sobre los productos y el inventario. A través de un sistema de información de inventario, la microempresa podrá superar estos desafíos y crecer en un entorno altamente competitivo.

En conclusión, esta investigación ofrecerá una solución viable para mejorar la gestión de datos de productos y materiales, incrementando así la eficiencia operativa y la competitividad de Productos de Concreto Daniel a largo plazo.

## **Objetivos**

### ***Objetivo General***

Implementar un sistema para la gestión y control de inventario en la microempresa Productos de Concreto Daniel, enfocado en el manejo de los datos de productos y materiales.

### ***Objetivos Específicos***

1. Recopilar los requisitos funcionales y no funcionales mediante consultas para el análisis del sistema.

2. Realizar trabajos de diseño desarrollando casos de uso, utilizando notación UML para satisfacer las necesidades de una microempresa.
3. Programar el sistema para la gestión y control de inventario, utilizando herramientas de desarrollo de software que sean de código abierto como java, MySQL y Visual Studio Code.
4. Ejecutar las pruebas necesarias para confirmar un prototipo funcional.

### **Justificación**

La investigación se centra en un problema fundamental de las microempresas que está vinculado con la mala administración de la información de los productos y suministros. La falta de un sistema de información apropiado ha tenido un efecto negativo en la eficiencia, provocando inexactitudes en el tratamiento de la información. Los datos actuales muestran la importancia de tener un sistema de gestión de información vigente.

El análisis propone una alternativa para abordar estos inconvenientes, con el objetivo de mejorar la gestión y control de la información en la microempresa. Esto involucra proporcionar información a tiempo sobre cómo reabastecerse de suministros y gestionarlos. Al explorar la administración de productos y suministros, el objetivo es abordar los errores encontrados en el manejo de recursos, potenciar las estrategias operativas de la microempresa y, en definitiva, incrementar la ganancia económica.

De la misma manera, la observación resalta la relevancia significativa en la comunidad al implementar un sistema tecnológico que fomente la exactitud y la constante actualización de la información, exhibiendo la cultura de la empresa. Este método no solo aumenta la eficacia interna de la microempresa, sino también que eleva la calidad de los servicios ofrecidos, generando una experiencia confiable para el cliente. La meta de la investigación es incrementar la eficiencia en la gestión de datos y tareas operativas para que la microempresa pueda adaptarse de manera favorable

a los resultados. Incrementar la precisión en el manejo de inventario y la gestión de pedidos podría reducir los costos operativos, eliminar desperdicios y mejorar la competitividad.

Por último, estas mejoras facilitarán al cliente el acceso a productos y servicios disponibles en todo momento. Esto no solo favorecerá la experiencia del cliente, también garantizará la permanencia de la microempresa en el mercado local y reforzará su prestigio en la comunidad.

## **Antecedentes**

### ***Antecedentes Internacionales***

Un estudio destacado en esta área fue realizado por Corella-Parra (2023), abordando el desarrollo de un sistema de gestión de inventarios para un sistema de riego, titulado “Desarrollo de un sistema de control de inventario para una empresa dedicada a la comercialización de sistemas de riego”. Este artículo sugiere una solución integral que involucra el registro de productos, el monitoreo de entradas y salidas, y la generación de informes y alertas de inventario. También se incluyen tácticas de enseñanza para respaldar el progreso del sistema, como elaborar manuales para los usuarios, formar al personal y evaluar regularmente el sistema propuesto. Este método no solo pretende mejorar la administración interna de la compañía, sino también, brindar un servicio superior a sus clientes y está estrechamente vinculado con la investigación en proceso.

En esta empresa de venta de productos para riego se llevará a cabo este proyecto que implica al Departamento de Almacén y Ventas. La compañía se encuentra con múltiples desafíos al manejar su inventario, como la atención de control en los movimientos de productos y la complicación de monitorear eficientemente los artículos en existencia. La meta principal consiste en crear un procedimiento para gestionar el inventario, con el fin de optimizar la circulación de materiales en el depósito, lograr registros exactos y seguros, y supervisar los niveles de existencias a través de un sistema de organización, auditorías internas y un software de control.

Uno de los desafíos más grandes para las empresas es mantener satisfechos a sus clientes y ofrecer un valor añadido que represente una ventaja competitiva. La dinámica en el crecimiento de las operaciones y los alcances regularmente opaca los problemas que impiden o limitan la mejora de un servicio u operar de manera eficiente algún proceso. La adaptación de la empresa a estos cambios puede determinar su supervivencia en el mercado. El inventario, siendo frecuentemente el mayor activo en las firmas, requiere de una gestión y control eficientes para garantizar el desempeño y la estabilidad. El exceso de producto puede incrementar los costos de almacenamiento, mientras que su agotamiento puede reducir las ventas, afectando negativamente al negocio (Sooksai, 2019).

El segundo trabajo internacional es de Rueda Vera (2022) y corresponde a “Sistemas de información y control de inventarios en microempresa y medianas empresas - MiPyMEs de la ciudad de Cúcuta, Colombia”, en ella se investiga cómo los sistemas de información pueden afectar la gestión de inventarios de las microempresas (MiPyMEs) en Cúcuta, Colombia.

Este estudio fue llevado a cabo por medio de una encuesta realizada a los gerentes y dueños. Se encontró que el uso de sistemas de información adecuados puede mejorar la eficiencia y rentabilidad de las MiPyMEs. Sin embargo, el informe también identificó varias dificultades en la implementación de los sistemas, tales como la falta de habilidades técnicas y la oposición a los cambios por parte de los usuarios.

La investigación se centró en un enfoque cuantitativo. Se utilizó un cuestionario validado por expertos. El análisis detallado de los resultados demostró una relación positiva entre la adopción de estos sistemas y la mejora en la gestión de inventarios. El estudio proporciona valiosos

conocimientos y recomendaciones para las MiPyMEs que pueden servir como antecedente para futuras investigaciones en esta área.

Este tercer proyecto trata acerca de un sistema computacional de Milagro E. Izquierdo Albarracín Miguel Rolando, Sr Shigla Morocho y Alex Paul S. (2021), llamado “Creación de un Sistema de Control y Administración en Línea de Reactivos e Insumos para el Laboratorio de Ingeniería de la Universidad Estatal de Milagro”. En el campo de la informática adaptada a la gestión de recursos de laboratorios, es cada vez más imprescindible implementar tecnologías que simplifiquen la gestión y control de los materiales y productos químicos en instituciones educativas.

En la Facultad de Ingeniería de la Universidad Estatal de Milagro se creó una herramienta en línea para controlar y administrar productos y materiales, empleando tecnologías web innovadoras y metodologías de desarrollo de software flexibles. Producción de informes de uso y petición automática de reposiciones.

Este estudio sienta un precedente significativo en el campo de la gestión de recursos de laboratorio, demostrando la efectividad y eficiencia de las soluciones tecnológicas para mejorar los procesos académicos y administrativos en las instituciones de educación superior. Igualmente, los métodos y utensilios utilizados durante la instalación del sistema podrían guiar futuros avances en esta industria.

### ***Antecedentes Nacionales***

Este trabajo corresponde a En M., De Proyectos G., Aguilar A., Alajuela P., Del J. (2021), que realizó una tesis llamada “Plan de gestión del proyecto Inventario Forestal Nacional de Costa Rica”, que corresponde al sector forestal de Costa Rica que ha experimentado un crecimiento significativo.

Ante este problema, esta tesis propone el diseño e implementación de un sistema de control de inventarios para el Instituto Costarricense de Electricidad (ICE), basándose en la metodología del IFNC.

El propósito principal de este sistema es mejorar la eficiencia y exactitud en la administración de los recursos forestales del ICE, al mismo tiempo que se disminuyen los riesgos de pérdida y se optimiza la toma de decisiones.

El proyecto propuesto se fundamenta en los principios y enfoque empleados en el IFNC, también se propone la utilización de tecnologías de seguimiento y localización avanzadas, como el empleo de GPS y sensores remotos, con el fin de mejorar la exactitud y eficacia en la recolección de información.

El segundo trabajo corresponde a Rodríguez (2023), quien elaboró una “Propuesta de sistema de gestión y control de inventarios en bodegas de almacenamiento a partir de un estudio de rotación y trazabilidad de los productos comercializados por Distribuidora La Ruta Costa Rica durante el periodo 2021-2022”. En un esfuerzo para superar este problema se realizó una clasificación de productos.

Se utiliza el método del proceso de análisis jerárquico para determinar la importancia relativa de los distintos productos almacenados en la bodega y se asignan ubicaciones con base en criterios establecidos. Se asignaron identificaciones a los estantes y se establecen políticas de almacenamiento para mantener actualizados los niveles de inventario y el espacio libre disponible.

Este trabajo de investigación se basa en la implementación de un sistema de gestión de almacenes, el cual incluye el uso de un sistema WMS (Warehouse Management System) para facilitar las tareas de gestión de almacenes, asignar ubicaciones de almacenamiento y reducir el

tiempo de las tareas. Además, se desarrolló un sistema de inventario apoyado en un sistema WMS para controlar el stock del almacén y reducir los desabastecimientos.

Este tercer trabajo nacional corresponde a Comunicación (2023) bajo el título de “Propuesta para mejorar la gestión de inventarios de roles en la bodega central de suministros, utilizando herramientas de calidad que ayuden a disminuir los costos operativos del proceso, en el departamento y área seleccionados en Ingenio Taboga, en el año 2022”.

Debido a que la gestión de inventario es un aspecto muy importante en cualquier organización, en especial en las industrias que requieren de unas grandes cantidades de materias primas y suministros, la industria azucarera no es una excepción. Esta investigación muestra que los sistemas de gestión de inventario que existen en el almacén central de suministros del Ingenio Taboga pueden tener algunas ineficiencias, lo que resultará en mayores costos operativos y posibles retrasos en la cadena de suministro.

Durante la gestión de inventario, las herramientas de calidad son útiles para mejorar los niveles de inventario, acortar los tiempos de entrega y disminuir los gastos operativos. En este estudio resalta la importancia de administrar la cadena de suministro. Al utilizar herramientas de calidad para la gestión de inventarios de los almacenes principales se puede incrementar la eficacia y eficiencia de los procedimientos y, a la vez, disminuir los gastos operativos.

## **Proyecciones**

### ***Alcances***

1. Mejora la gestión y control de inventario en la microempresa Productos de Concreto Daniel para evitar la pérdida de información sobre los suministros y productos.

2. Desarrollar un sistema para la gestión y control de inventario en la microempresa Productos de Concreto Daniel que cumpla con los requerimientos de seguridad, que se pueda eliminar, editar y agregar información sobre los productos y suministros.
3. Desarrollar un módulo de registro por medio del correo electrónico para tener acceso al sistema.
4. Se desarrollarán módulos para categorías, proveedores, inventario, productos, clientes, pedidos. Además, se incluirá un módulo de seguridad para la asignación de roles de usuario.
5. No se implementará un manual de usuario.
6. Tampoco se capacitará al personal.
7. Solo se implementarán los permisos que sean requeridos por el usuario.
8. No se realizarán estudios de factibilidad comercial, de tiempo, política o legal.

### ***Limitaciones***

1. Una de las más grandes limitaciones con la que se cuenta es que la tecnología cambia constantemente y no se puede medir en qué momento un programa o permiso va a tener alguna variación, la cual nos impida seguir utilizándola.
2. Debido a la falta de un sistema de inventario eficiente, hay limitaciones en la disponibilidad de datos e información.
3. El proyecto no contempla los gastos asociados a licencias de software ni al equipo de hardware requerido para su implementación.

## **Capítulo II. Marco Teórico**

## Historia de la Programación

La historia de los lenguajes de programación comienza con computadores mecánicos y avanza hacia herramientas modernas de desarrollo de software. Inicialmente eran especializados, con notación matemática y sintaxis compleja. La investigación en teoría de compiladores creó lenguajes de alto nivel, más accesibles. Ada Lovelace tradujo el primer programa de computadora de Charles Babbage en la década de 1840. Alonzo Church formalizó el cálculo lambda y la Máquina de Turing simplificó la operación de las máquinas. Los primeros lenguajes describían acciones en máquinas como los telares Jacquard y la máquina de diferencia de Babbage (Historia de los lenguajes de programación, 2023).

### Primeros lenguajes de programación

Se crearon para 1940, los primeros ordenadores alimentados por medio de electricidad y también se desarrollaron los primeros lenguajes de programación, los cuales fueron Plankalkül y Short Code. Pero estos lenguajes fueron altamente especializados y se basaron en las notaciones matemáticas y en la sintaxis oscura. Esto también llevó al desarrollo de la programación de alto nivel, debido a lo complejo que es la programación del lenguaje de ensamblador, dado que requiere de un esfuerzo intelectual muy grande. En 1950, se crearon los primeros lenguajes funcionales de programación, entre ellos están:

**Autocode:** automatizó la traducción del código en los primeros lenguajes de alto nivel.

**FORTRAN:** estableció la programación científica y de alto rendimiento.

**LISP:** introdujo la programación simbólica y el procesamiento de listas.

**COBOL:** Facilitó la programación para aplicaciones comerciales y de gestión.

**ALGOL:** sentó las bases de la programación estructural y la sintaxis de lenguajes modernos.

Estos lenguajes fueron los que introdujeron los conceptos de estructura de bloques anidados, ámbito léxico y notación matemáticamente exacta. En particular, FORTRAN fue el primer lenguaje que tuvo una implementación funcional de alto nivel con propósito general y fue ampliamente utilizado.

**Figura 1**

*Historia de la programación*



Fuente: *(Dos siglos de historia del lenguaje de programación, s. f.)*

Para 1960 y 1970 se crearon los paradigmas fundamentales, entre ellos:

**Simula:** introdujo la programación orientada a objetos.

**NORTH:** se centró en la programación concatenativa.

**C:** estableció la programación de sistemas.

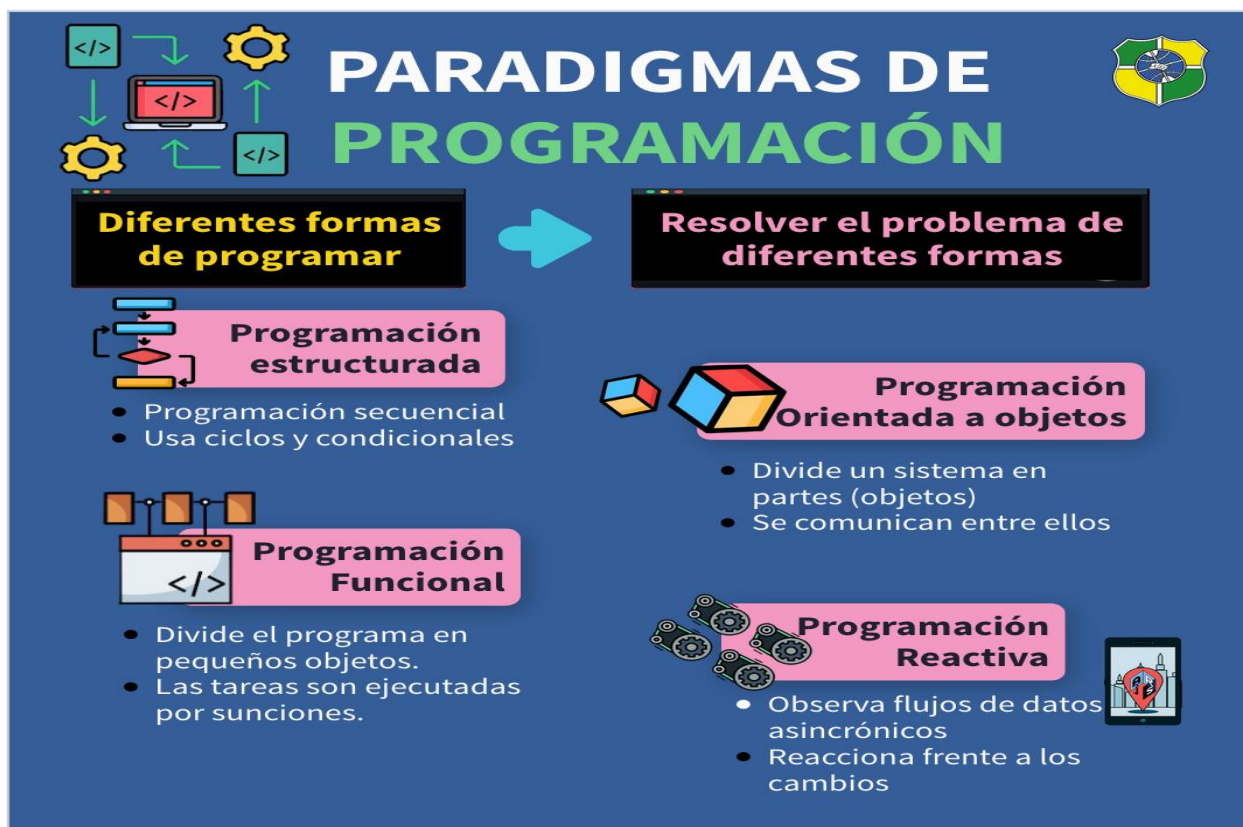
**Prolog:** introdujo la programación lógica.

**ML:** representó la programación funcional.

La mayoría de los lenguajes modernos son descendientes de estos paradigmas.

Figura 2

Paradigmas de la programación



Fuente: (¿Qué son los paradigmas de programación? - Platzi, s. f.)

En 1980 los lenguajes imperativos experimentaron una consolidación relativa con el desarrollo de lenguajes como:

**ADA:** introdujo la programación para sistemas críticos y embebidos con alta facilidad.

**ML:** permitió el aprendizaje automático para datos.

**C++:** extendió C con programación orientada a objetos y de alto rendimiento.

Durante este período se enfatizó la programación para sistemas a gran escala, utilizando módulos. Esta tendencia continuó hasta la década de 1990.

Para 1990, la llegada de la Web condujo al desarrollo de lenguajes como:

**Perl:** popularizó la manipulación de texto y el procesamiento de archivos.

**Python:** destacó por su sencillez y legibilidad en programación general.

**Java:** introdujo el concepto de escribir una vez y ejecutar en cualquier lugar con una máquina virtual.

**Ruby:** promovió el desarrollo ágil y la programación orientada a objetos con una sintaxis sencilla.

Los lenguajes de programación de secuencias también se hicieron populares, como fue el caso de:

**JavaScript:** se encarga de cargar en el navegador web y que se ejecute en el sistema del cliente.

Los lenguajes de programación han evolucionado para extender, mejorar y desarrollar lenguajes más seguros desde 2000. Algunos de estos lenguajes serían:

**Rust:** prioriza la seguridad de memoria sin sacrificar el rendimiento.

**GO:** simplifica el desarrollo concurrente y estable.

**Swift:** modernizó el desarrollo de iOS y macOS con una sintaxis segura y eficiente.

Y lenguajes específicos para la ciencia de datos y el aprendizaje automático. La evolución de la informática garantiza un futuro con variedad de lenguajes y aplicaciones únicas. Algunos de los lenguajes más demandados incluyen:

**JavaScript:** revolucionó el desarrollo web interactuando en el navegador.

**Python:** destacó por ser simple y versátil en múltiples dominios.

**Java:** estableció la probabilidad con escribir una vez y ejecutar en cualquier lugar.

**C#:** potencia el desarrollo en las plataformas de Microsoft con programación orientada a objetos.

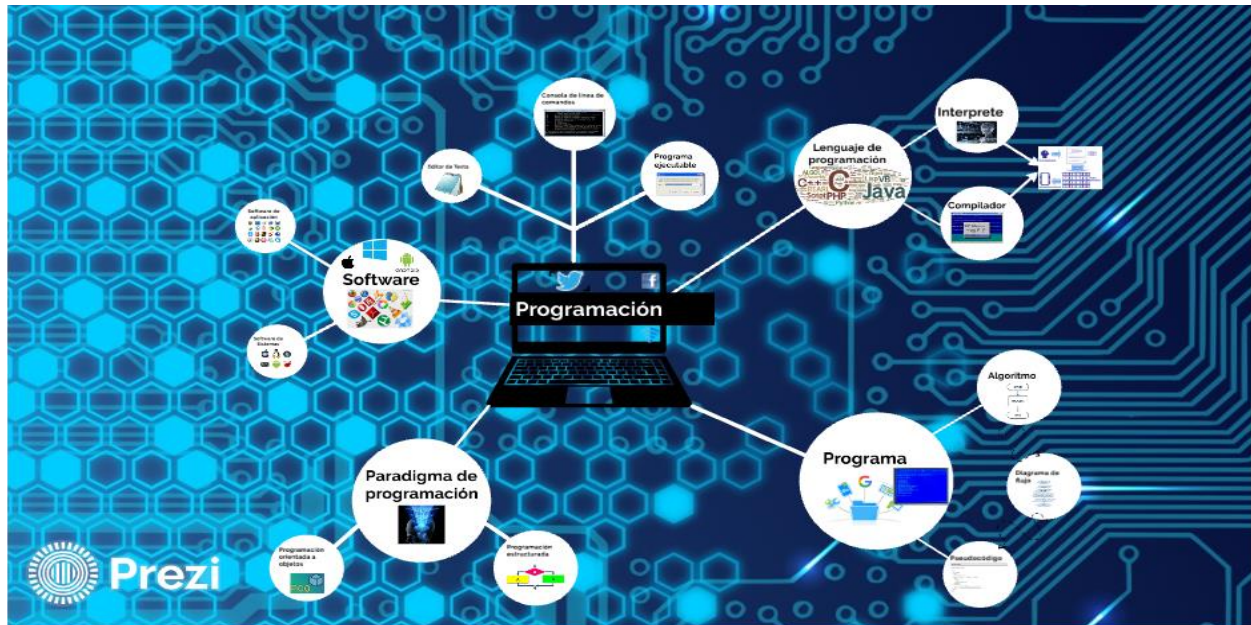
**SQL:** normalizó el manejo de las bases de datos relacionales por medio de consultas.

Estas se utilizan en una amplia gama de aplicaciones, desde el desarrollo web y móvil hasta la inteligencia artificial y el análisis de datos.

## Fundamentos de la Programación

**Figura 3**

*Mapa conceptual fundamentos de programación*



Fuente: (Mapa conceptual fundamentos de programación por Christopher Diego Jiménez Almaraz en Prezi, s. f.)

### *¿Qué es un lenguaje de programación?*

Es una herramienta fundamental que habilita al usuario para dar órdenes mediante un grupo de normas y signos de programación. Los programadores pueden comunicarse con las computadoras usando estos lenguajes para solicitarles la realización de tareas concretas.

### *Tipos de Programación*

Existen varios tipos de programación para el desarrollo de software, según el artículo escrito por ¿Qué Es La Programación, Sus Lenguajes y Tipos de Enfoque? (Tiffin University, n.

d.), por lo que cada uno cuenta con sus características y enfoques específicos. Los principales tipos de programación serían los siguientes:

**Imperativa:** este tipo de programación está diseñado para hacer un seguimiento de un conjunto de instrucciones paso a paso. Están organizados en forma de una serie de instrucciones que alteran el estado del programa mientras son ejecutados. Las instrucciones son indicaciones para la computadora que le indican los pasos necesarios para lograr el resultado deseado (Tiffin University, n. d.).

**Declarativa:** en lugar de especificar cada paso que la computadora debe completar, describe lo que desea lograr. Se proporcionan especificaciones para los resultados deseados y el sistema o lenguaje es responsable de encontrar la mejor manera de lograrlo (Tiffin University, n. d.).

**Funcional:** este tipo de programación trata a la computación como la evaluación de funciones matemáticas, debido a que en lugar de cambiar los estados o modificar datos directamente, los programas se construyen por medio de la composición de funciones, promoviendo un enfoque más declarativo y menos centrado en la manipulación directa de los datos (Tiffin University, n. d.).

**Lógica:** en este modelo, los programas se desarrollan detallando los hechos y reglas, y un motor de inferencia deduce conclusiones a partir de estos elementos. Es especialmente útil en situaciones en las que el razonamiento y la manipulación simbólica son claves (Tiffin University, n. d.).

**Orientada a eventos:** el enfoque principal es en el control de eventos, los cuales son actividades o situaciones que suceden en el sistema, como, por ejemplo, las interacciones de un

usuario. Este estilo de codificación es típico en las apps que reaccionan a interacciones en directo, tales como juegos, interfaces visuales o sistemas de interacción (Tiffin University, n. d.).

**Orientada a objetos:** la POO organiza el código en torno a “objetos”, como unidades independientes que combinan los datos y funciones relacionadas. Este imita conceptos del mundo real, lo que facilita la representación de problemas complejos y promueve la reutilización del código (Tiffin University, n. d.).

#### Figura 4

*Lenguaje de programación*



Fuente: *(Tipos y niveles de lenguajes de programación ejemplos 2024 | AAU, s. f.)*

#### **Tipos de Lenguajes de Programación**

No hay un lenguaje de programación universal según el artículo proporcionado por, Latam (2023). La elección depende del propósito del proyecto. Los lenguajes varían según características y necesidades específicas.

**Bajo nivel:** los lenguajes de bajo nivel, como ensamblador y máquina controlan directamente el hardware para ejecuciones eficientes. Adecuados para aplicaciones exigentes en

rendimiento, requieren conocimiento profundo, siendo menos accesibles y más complicados que los de alto nivel (Maldonado, 2024).

**Alto nivel:** los lenguajes de programación de alto nivel, como Python, Java y C++, son más fáciles de entender y usar que los de bajo nivel. Facilitan la escritura de código legible y eficaz, separando los detalles técnicos del hardware. Estas herramientas simplifican la programación al ofrecer sintaxis clara, bibliotecas y frameworks, siendo ampliamente utilizadas en la industria del software (¿Qué es un lenguaje de programación de alto nivel? Y por qué deberías saberlo, 2024).

**Figura 5**

*Lenguajes de alto nivel vs. de bajo nivel*

**LENGUAJES DE PROGRAMACIÓN ALTO NIVEL VS BAJO NIVEL**

Es un lenguaje que entienden **los humanos**.

Son instrucciones **para el procesador**.

```

1 #include <iostream>
2
3 using namespace std;
4
5 int main() {
6
7     cout << "Hola EDteam" << endl;
8
9     return 0;
10 }

```

```

1 COPY    START    2010H
2         LDH      ZERO
3 MOVECH  LDCH     STR1, X
4         STCH    STR2, X
5         TIX     FOUR
6         JLT     MOVECH
7 STR1    BYTE    C'HOLA'
8 STR2    RESB   4
9 ZERO   WORD    0
10 FOUR  WORD    4
11 END

```

Está orientado al **software**.

Está orientado al **hardware**.

Utilizan **menos instrucciones** para realizar una acción.

Nos ayuda a entender **cómo funcionan las instrucciones** en la computadora.

Te permite programar **aplicaciones y videojuegos**.

**ASSEMBLY** Puedes construir **sistemas operativos y núcleos**.

EDteam Aprende a programar en: [ed.team/programacion](https://ed.team/programacion)

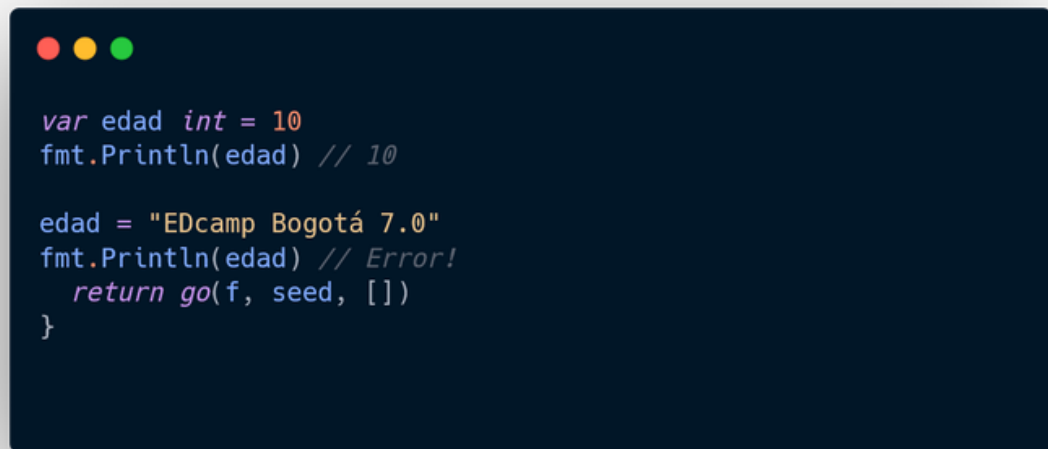
Fuente: (*Lenguajes de Alto nivel vs. de Bajo nivel* / EDteam, s. f.)

**Tipado estático:** en ciertos lenguajes como Go, Java, C#, C, C++, TypeScript y Kotlin, las variables deben declararse con un tipo específico que no puede cambiarse más adelante. Por

ejemplo, una variable diseñada para almacenar números no puede usarse para texto. Estos lenguajes verifican los tipos durante la compilación para evitar errores en tiempo de ejecución (*¿Qué son los lenguajes tipados y no tipados? (Explicación sencilla) | EDteam, s. f.*).

## Figura 6

*Lenguajes tipados estático*

A screenshot of a code editor with a dark background and three colored window control buttons (red, yellow, green) at the top left. The code is written in Go and demonstrates a type error. The first line declares a variable 'edad' of type 'int' and assigns the value 10. The second line prints '10'. The third line reassigns 'edad' to a string 'EDcamp Bogotá 7.0'. The fourth line attempts to print 'edad', which causes a compile-time error because the variable's type has changed from int to string. The code ends with a 'return go(f, seed, [])' statement and a closing brace.

```
var edad int = 10
fmt.Println(edad) // 10

edad = "EDcamp Bogotá 7.0"
fmt.Println(edad) // Error!
return go(f, seed, [])
}
```

Fuente:(*¿Qué son los lenguajes tipados y no tipados? (Explicación sencilla) | EDteam, s. f.*)

**Tipado dinámico:** Python es un ejemplo de lenguaje de programación dinámico que permite que las variables cambien de tipo durante la ejecución del programa. La verificación de tipos en estos idiomas ocurre durante la ejecución, lo que puede llevar a identificar errores relacionados con los tipos después de que el programa esté en marcha. Otros ejemplos de lenguajes dinámicos incluyen JavaScript, PHP y Ruby (*¿Qué son los lenguajes tipados y no tipados? (Explicación sencilla) | EDteam, s. f.*).

**Figura 7***Lenguajes tipados dinámicos*

```
edad = 7
print(edad) # 7

edad = "EDcamp Bogotá 2023"
print(edad) # "EDcamp Bogotá 2023" No da error
```

Fuente:(¿*Qué son los lenguajes tipados y no tipados? (Explicación sencilla)* | EDteam, s.

f.)

**Compilado:** de acuerdo con la información proporcionada, la compilación convierte el código fuente en código máquina para una ejecución eficiente y rápida. Ejemplos de lenguajes compilados incluyen C#, C++ y Go, que se convierten a código máquina antes de ejecutarse (Tipos de Lenguajes de Programación | EDteam, n. d.).

**Interpretado:** según la investigación los lenguajes como JavaScript, PHP, Python y Ruby se traducen en tiempo real mediante intérpretes, no compiladores. La diferencia principal es que el intérprete traduce durante la ejecución, lo que puede ralentizar el proceso (Tipos de Lenguajes de Programación | EDteam, n. d.).

**Intermedio:** el artículo redactado por Tipos de Lenguajes de Programación | EDteam (n. d.), menciona una etapa adicional en el proceso de desarrollo. Tras compilar el código fuente, se convierte en “Bytecode” para la JVM, permitiendo ejecutar el código en diferentes sistemas

operativos. Ejemplos de estos lenguajes son Java, Kotlin y Scala, ofreciendo flexibilidad y ejecución multiplataforma.

## Figura 8

### *Lenguajes de programación*



Fuente: *(Tipos de lenguajes de programación | EDteam, s. f.)*

### ***Paradigmas de Programación***

Robert C. Martín indica que los paradigmas son principios que guían la estructura y redacción del código. Los principales paradigmas de programación son: el imperativo que incluye la programación orientada a objetos y el declarativo que abarca la programación funcional y lógica. Estos influyen en el pensamiento y diseño de los programadores.

Figura 9

¿Qué son los paradigmas de programación?



Fuente: (¿Qué son los paradigmas de programación? | EDteam, s. f.)

**Orientada a objetos:** una técnica que facilita notablemente la simplicidad del código es la creación de estructuras conocidas como clases. Estas permiten agrupar funcionalidades bien definidas y delimitadas, que no solo mejora la organización, sino también, acelera el desarrollo de los programas al favorecer la reutilización del código. Lo más relevante es que esta organización más estructurada permite abordar el desarrollo de programas más complejos de manera eficiente.

Figura 10

*Programación orientada a objetos*



Fuente: (¿Qué Es La Programación Orientada a Objetos? | EDteam, n. d.)

**Funcional:** las funciones se consideran valores, permitiendo pasarse como argumentos.

Tres principios importantes: invariabilidad, funciones puras y funciones de grado superior. La invariabilidad implica que los datos no deben modificarse una vez creados. Las funciones puras siempre devuelven el mismo resultado al recibir los mismos argumentos sin efectos secundarios.

Las funciones de alto nivel reciben funciones como argumentos.

**Figura 11**

*Programación funcional*

```
public class Main {  
    public static void main(String[] args) {  
  
        IHello hello = () -> { System.out.println("Hello world"); }  
        hello.SayHello();  
  
    }  
}
```

Fuente: (*Introducción a la programación funcional con Java | by Johan Del Valle | Medium, s. f.*)

**Imperativa:** la programación imperativa enfoca detallar procesos para cambiar el estado del sistema con instrucciones como bucles y condicionales. Se basa en “estado” y “cambio de estado” para controlar el flujo del programa. Aunque brinda control detallado, puede generar código menos modular y complejo de mantener en comparación con enfoques más simples para el manejo del estado.

**Figura 12**

*Programación imperativa (PHP)*

```
$listaParticipantes = [1 => 'Peter', 2 => 'Hans', 3 => 'Sarah'];  
$nombres = [];  
foreach ($listaParticipantes as $id => $name) {  
    $nombres[] = $name;  
}
```

Fuente: (*Programación imperativa: ventajas y desventajas del paradigma - IONOS, s. f.*)

**Declarativa:** la programación declarativa prioriza el resultado sobre los pasos, utilizando lógica y reglas para definir el comportamiento del sistema. Facilita la creación de aplicaciones complejas al extraer detalles de implementación.

### Figura 13

*Programación declarativa*

```
$nombres = array_values($listaparticipantes);
```

Fuente: (*Programación imperativa: ventajas y desventajas del paradigma - IONOS*, s. f.)

### *Algoritmos de Programación*

Los algoritmos son esenciales en software para optimizar rendimiento, ejecutar tareas y resolver problemas matemáticos y decisiones en sistemas complejos.

Figura 14.

*Los algoritmos en la programación*



Fuente: (*¿Cómo aprender a programar desde cero?* | EDteam, s. f.)

## Sistema de Inventarios

Los inventarios aseguran control y organización para empresas de mercancías, siendo fundamentales para el éxito del negocio. Con un sistema adecuado se determina la cantidad precisa de productos, garantizando distribución puntual y adaptación a la demanda. Es importante conocer los diferentes tipos de inventarios y métodos de gestión para optimizar su funcionamiento. (*¿Cómo funciona un sistema de inventario?* | Defontana Chile, s. f.)

## Figura 15

### *Sistemas de inventarios*



Fuente: *(El concepto del sistema de control de inventario, el gerente profesional y el trabajador están verificando el suministro de bienes y existencias, la gestión de inventario con la demanda de bienes 12494550 Vector en Vecteezy, s. f.)*

### ***¿Qué son los Sistemas de Inventarios?***

Los inventarios son esenciales para el control y organización de empresas que manejan mercancías. Un sistema adecuado permite determinar cantidades, cumplir plazos y adaptarse a cambios en la demanda. Es crucial para el éxito del negocio. Conociendo los diferentes tipos de inventarios y métodos de gestión se logra un manejo más efectivo de los recursos (sistema de inventario: Toma el control de tu almacén (2023) | Colombia, s. f.).

### ***¿Qué son los Métodos Para el Control de Inventarios?***

Los métodos para el control de inventarios son enfoques utilizados para gestionar eficientemente los niveles de inventarios que existen, ayudando a las empresas a nivelar la oferta y la demanda, optimizar recursos y minimizar costos (*Sistema de inventarios: qué es y cuáles son sus tipos, s. f.*). Estos son algunos métodos:

**Método ABC:** los artículos de clase A representan el 20% del inventario, pero el 80% del valor. Los de clase B aportan el 15% del valor. Tienen ventas moderadas y requieren de atención especial para una gestión financiera rigurosa.

**Método PEPS:** PEPS se usa en alimentos y textiles para mantener precios competitivos en épocas inflacionarias. Vende mercancía comprada a precios anteriores, evitando la obsolescencia al sacar primero los productos más antiguos del inventario.

**Método EOQ:** busca minimizar los costos totales del inventario, determinando la cantidad óptima de pedido para equilibrar los costos de almacenamiento, pedidos y demanda esperada. Este método se basa en tres principales.

- Que la demanda sea constante y predecible.
- Que la frecuencia de uso del inventario sea estable.
- Que los pedidos se reciban exactamente cuando el inventario se agota.

**Método UEPS:** el método UEPS (último en entrar, primero en salir) es una estrategia de gestión de inventarios en la cual se retiran primero los productos más recientes, también conocido como LIFO en inglés. El conteo cíclico implica contar regularmente una parte del inventario total para garantizar que cada segmento sea contado al menos una vez en un período específico (*Método UEPS (last in, first out): usos y ejemplos, s. f.*).

**Método de conteo cíclico:** el conteo cíclico implica el recuento frecuente de una parte del inventario total, asegurando que cada segmento del inventario sea contado al menos una vez en un período determinado.

Este método se complementa con el método ABC, asignando diferentes frecuencias de conteo a cada categoría. Entre sus utilidades se encuentra la mejora en la precisión y fiabilidad del

control de inventarios, permitiendo identificar y corregir discrepancias sin necesidad de un conteo total.

El conteo cíclico, en combinación con el método EOQ, es útil cuando la demanda es conocida. Utilizar un software especializado para el seguimiento regular del inventario puede facilitar este proceso, permitiendo programar ciclos de conteo para otras categorías de productos o ubicaciones, reduciendo la interrupción de las operaciones diarias.

**Figura 16**

*Métodos para el Control de Inventarios*



Fuente: *(El concepto del sistema de control de inventario, el gerente profesional y el trabajador están verificando el suministro de bienes y existencias, la gestión de inventario con la demanda de bienes 12494550 Vector en Vecteezy, s. f.)*

***Tipos de Inventarios***

Concordando con el artículo Tipos y clasificación de inventarios existentes (s. f.): “Los diversos tipos de inventarios permiten a las empresas recolectar y clasificar sus activos y productos utilizando distintos criterios”. El monitoreo del inventario puede detectar errores logísticos como diferencias en referencias, defectos en etiquetas o selección incorrecta. Existen diversos tipos de inventarios que varían según el criterio utilizado de definición.

**Según el momento:** estos tipos de inventario son claves para manejar picos de demanda y garantizar la disponibilidad de productos para satisfacer las necesidades del mercado.

***Inventario inicial:*** el inventario inicial al inicio del ejercicio contable muestra la cantidad de mercancías en cada departamento. Debe detallarse para evitar la escasez en un área y el exceso en otra.

***Inventario final:*** entre los tipos de inventarios de productos, el inventario final es la cantidad y el valor total de los bienes al final del año, determinado al cierre del ejercicio contable para productos específicos.

**Según la periodicidad:** el tipo de inventario según la frecuencia con la que se realiza. Se utiliza para verificar y actualizar el registro de stock de manera programada, ya sea a través de sistemas de gestión o de forma manual por parte de los empleados. Estos tipos de inventario incluyen:

***Inventario intermitente:*** se realiza para determinar la cantidad de mercancías, materiales o materia prima disponibles. En este tipo de inventario, la fecha para la próxima revisión es a largo plazo y las verificaciones pueden hacerse según sea necesario sin incorporarlas al inventario permanente.

***Inventario perpetuo:*** el inventario perpetuo, el sistema de inventarios más utilizado, registra las existencias de manera continua y detallada, permitiendo conocer en todo momento el monto vendido y los inventarios finales sin conteos manuales. Esto ahorra tiempo y acelera la elaboración de los estados financieros mensuales.

***Inventario periódico:*** es un tipo de inventario cíclico programado regularmente, como cada mes, semestre o año, para controlar y planificar el stock de productos, aprovisionamiento y ventas. Permite conocer cantidades exactas y optimizar la gestión de inventario.

**Según la fase de producción:** la siguiente clasificación agrupa el inventario según la fase de producción en la que se encuentre:

***Inventario materia prima:*** este tipo de inventario de materias primas es el stock de insumos esenciales para la fabricación de productos.

***Inventario de productos en etapa de fabricación:*** se monitorea la mercancía según su etapa de fabricación para establecer fechas de entrega. Se revisa el inventario para garantizar una producción eficiente.

***Inventario de productos terminados:*** cuando las mercancías están terminadas se incorporan al inventario de productos terminados para su distribución y comercialización.

**Según la función por determinar:** la siguiente clasificación se basa en la función del inventario para lograr una gestión óptima.

***Inventario en tránsito:*** son bienes pagados por un proveedor en algún punto de la cadena de suministro sin estar en almacenes. Probablemente en camino para entrega.

***Inventario de ciclo:*** se utiliza para asegurar que la producción sea equivalente a la cantidad de productos vendidos o demandados por los clientes. Su función es evitar la sobreproducción y el exceso de stock.

***Inventario de seguridad:*** se utiliza para prevenir emergencias y establecer un plan de contingencia que permita actuar rápidamente ante posibles incidentes.

***Inventario de desacoplamiento:*** el inventario de desacoplamiento coordina procesos de inventariado para evitar solapamientos, basándose en función y ubicación en la cadena de suministro.

**Según logística:** según el plan logístico se lleva a cabo para optimizar la cadena de suministro y se clasifica en diversas categorías.

***Inventario de existencias para especulación:*** es un sistema de control que permite conocer la cantidad de productos y materias primas disponibles para determinar si es viable aumentar la producción. Este inventario se lleva a cabo para anticiparse a momentos de alta demanda.

***Inventario de existencias obsoletas:*** permite controlar las mercancías, materias primas y productos terminados, con el objetivo de reducir las pérdidas en productos obsoletos, dañados o caducados.

***Inventario de previsión o anticipación:*** se realiza para cubrir necesidades futuras, reduciendo peligros y asegurando el stock antes de periodos de alta demanda, promociones o cierres de plantas.

***Inventario en consignación:*** la mercancía en el almacén está disponible para la venta, pero sigue siendo propiedad del proveedor, los registros controlan las fechas de ingreso y ventas.

***Inventario de lote:*** este tipo de inventario se basa en pedir mercancías en tamaño de lote o producirlas en grandes cantidades, dado que resulta más económico anticipar el stock y la demanda.

Existen otras maneras de clasificar diferentes tipos de inventarios, tales como el inventario físico, de mínimo y máximo, de agregados, la línea y el de cuarentena.

## **Sistemas de Información**

Es una estructura organizada de personas, procedimientos y herramientas destinadas a la gestión eficiente de datos e información. Su objetivo es asegurar que la recuperación y el procesamiento de la información se realicen de manera ágil y sencilla.

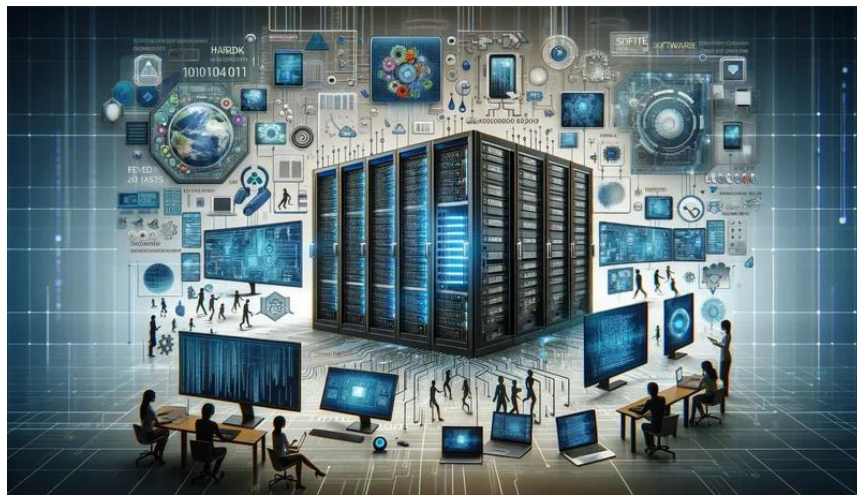
De acuerdo con Pursell (2023): “Un sistema de información es el conjunto de técnicas, herramientas y agentes involucrados en la administración y uso de datos para la obtención de objetivos empresariales”. Estos sistemas ayudan con la gestión de información que produce y

utiliza una organización para el mejoramiento de procesos y operaciones. Los componentes de un sistema de información facilitan varios procesos claves, a saber:

- La entrada de datos.
- La administración.
- El procesamiento.
- El almacenamiento.
- La distribución de la información a quienes necesitan acceder a ella.

**Figura 17**

*Sistemas de Información*



Fuente: *(Los sistemas de información de una empresa, s. f.)*

De acuerdo con *Sistemas de información en empresas: definición, tipos y ejemplos (s. f.)* se pueden clasificar en siete tipos, los cuales se detallan a continuación:

- Sistemas de procesamiento de transacciones
- Sistemas de información gerencial
- Sistemas de control de procesos de negocio
- Sistemas de información de marketing

- Sistemas de colaboración empresarial
- Sistema de soporte a la toma de decisiones
- Sistemas de información ejecutiva

### ***Sistemas de Procesamiento de Transacciones***

Son sistemas empresariales utilizados para el nivel operativo de una empresa. Constituyen un sistema computarizado que realiza y registra transacciones de rutina diarias para el buen funcionamiento de una compañía. En otras palabras, da soporte a las operaciones cotidianas de un negocio. Sus componentes son:

- Hardware
- Software
- Bases de datos
- Telecomunicaciones
- Personal
- Procedimientos con los que se procesan las transacciones:
  - La recopilación
  - La edición
  - La corrección
  - La manipulación
  - El almacenamiento de datos
  - La producción de documentos

### ***Sistemas de Información Gerencial***

Conocidos también como sistemas de información de gestión, apoyan la toma de decisiones estructuradas o semiestructuradas de los mandos intermedios. Su principal labor es la de sintetizar

la información de rutina de una empresa, para asegurar el buen funcionamiento de los procesos, por ejemplo, los informes bimestrales, mensuales o semanales. Existe un sistema de información gerencial para diferentes departamentos de una compañía. Entre los más utilizados se encuentran:

- Marketing: producto, precio, competencia
- Producción: rendimiento, inventarios, proveedores
- Finanzas: inversiones, mercados, cotizaciones
- Recursos humanos: incentivos, reclutamiento, desempeño

Componentes del sistema:

- Hardware
- Software
- Datos
- Personas por gerenciar
- Métodos: las prácticas operativas, modelos e instrucciones que están en los manuales de la empresa.

### ***Sistemas de Control de Procesos de Negocio***

Este tipo de sistemas es utilizado para monitorear y controlar procesos industriales o físicos. Se utilizan sobre todo en la industria petrolera, siderúrgica o de generación de energía. Incluyen varios equipos, programas especializados y procedimientos de operación.

Con un BPM, las empresas adoptan una serie de pasos o acciones para modificar su forma de trabajar con el fin de mejorar sus procesos y facilitar la colaboración con un enfoque centrado en el cliente. Los componentes del sistema son:

- Entorno gráfico de diseño: son los que modelan la secuencia de actividades del proceso o proyecto.

- Middleware: son las que facilitan la comunicación del motor con las aplicaciones que se usan en el BPM.
- Metadatos: rol, usuario, sistema, servicio
- Interfaz de usuario
- Monitorización: son para darle seguimiento puntual a procesos y actividades.
- Reglas de negocio
- Simulación: las que permiten la ejecución de los procesos en un ambiente controlado.
- Automatización
- Ejecución en la nube
- Estándares: se utilizan para que todos los involucrados conozcan y manejen los mismos códigos, gráficos, actividades.

### ***Sistemas de Información de Marketing***

Las SIM son un conjunto de relaciones estructuradas entre personal, máquinas y procedimientos para generar un mejor flujo de información. Estos datos provienen tanto de fuentes internas como externas y sirven para tomar decisiones más inteligentes relevantes al área de marketing.

Con su uso pueden determinar principalmente el grado de fiabilidad que tiene una información, el ritmo actualizado de información y un grado de detalle más específico. Por medio de este proceso es posible hacer las modificaciones y adaptaciones pertinentes en diferentes situaciones de la empresa. Los componentes del sistema son:

- Datos internos: estas son métricas de actividades y rendimiento interno.

- Inteligencia de marketing: es información del mercado e industria de fuentes confiables.
- Investigación de mercados: son los datos cualitativos y cuantitativos que se recogen por medio de herramientas y actividades realizadas directamente por la empresa.
- Soporte a las decisiones de marketing: en este se incluyen también las plataformas de análisis de datos y reportes.

### ***Sistemas de Colaboración Empresarial***

Los ERP son los más utilizados en el mercado. Tienen la función de ayudar a los directivos de una empresa para así controlar el flujo de información y proporcionar una vista integral de cada una de las áreas que conforman la compañía para que de esta forma se pueda mejorar la gestión y toma de decisiones.

Su tecnología está encargada, se registra e integra los procesos de negocio y ayuda a las empresas a optimizar sus procesos para tener una mayor comprensión de esta a través de la automatización. Con este tipo de programas, los líderes de los negocios pueden actuar con más rapidez con la seguridad de que estarán basando sus decisiones en información confiable y en tiempo real. Los componentes del sistema son:

- Datos
- Hardware
- Software
- Procedimientos

### ***Sistema de Soporte de Toma de Decisiones***

Conocido por sus siglas en inglés como DSS, es un sistema basado en ordenadores que es utilizado regularmente por los gerentes para tomar una decisión con el fin de resolver un problema

en la empresa. Permite la formulación de cálculos, comparar opciones y predecir escenarios para saber cuál es la mejor opción por tomar. Está diseñado para ayudar en situaciones complejas.

El software que se utiliza tiene la función de recolectar información procedente de documentos, personal interno, modelos comerciales, experiencias o análisis externos. Con estos datos se identifican los problemas o desafíos. De igual forma, favorecen el incremento de la efectividad de manera oportuna. Los componentes del sistema son:

- Gestor de datos: es la base de conocimiento.
- Gestor de modelos
- Motor de conocimiento
- Interfaz de usuario
- Usuarios: cada uno tiene sus roles; recolectores de datos, asesores, tomadores de decisiones.
- Contexto de decisión: establecido y bien definido

### ***Sistemas de Información Ejecutiva***

Los EIS son los encargados de brindar acceso rápido a la información interna y externa relevante de una empresa. Suelen visualizarse en formatos gráficos para la presentación de datos básicos, pero detallados, con la finalidad de que los gerentes o ejecutivos puedan leer la información con facilidad.

Uno de los caracteres más sobresalientes de este tipo de sistemas es que les dan a los que toman las decisiones los indicadores de negocio que afectan al instante. También, permiten un análisis detallado de los que no están cumpliendo con las expectativas para determinar el plan de acción más conveniente. Los componentes del sistema son:

- Interfaz gráfica

- Indicadores de negocio: estos son definidos por la empresa.
- Integración de bases de datos: pueden ser con datamart o data warehouse.
- Usuarios
- Acceso en la nube

### **El Ciclo de Vida de un Sistema de Información**

El ciclo de vida de un sistema de información garantiza la satisfacción del cliente, controla el presupuesto y los plazos. Incluye fases que van desde el nacimiento hasta la expiración del sistema y consiste en las siguientes fases:

#### ***Planificación***

El analista identifica en esta etapa inicial las necesidades del usuario y realiza la planificación necesaria para visualizar el proceso y definir cómo lograrlo. No hay sistema desarrollado todavía. El analista determina los datos necesarios para alcanzar el objetivo, así como los recursos disponibles y necesarios para el proyecto.

#### ***Análisis de Requerimientos***

La etapa de desarrollo de un sistema de información inicia luego de conocer todos los requerimientos necesarios. El objetivo es crear un diseño base rentable y eficiente desde el principio. Durante esta fase, el analista examina los requerimientos funcionales del sistema, a través de entrevistas o encuestas a los usuarios actuales. Es crucial determinar la viabilidad del plan de acción, considerando aspectos financieros, tecnológicos y organizativos. En caso necesario, se pueden realizar modificaciones para mejorar el proyecto.

***Diseño***

El analista ha definido los requisitos del nuevo sistema de información y comienza a trabajar en planos y diseño, integrando información clave para la empresa. Se centra en el diseño de software e infraestructura, incluyendo interfaces de usuario, bases de datos y seguridad.

***Desarrollo***

Se activa un periodo de prueba tras finalizar el sistema para ajustar detalles y satisfacer expectativas. Este proceso implica una colaboración completa entre desarrolladores y analistas para integrar tecnología e información adecuada.

***Prueba***

Se verifica en esta fase, el funcionamiento real del nuevo sistema de información, eliminando detalles no deseados y realizando los ajustes necesarios. Requiere de inspección minuciosa para detectar y corregir problemas, garantizando el alineamiento correcto.

***Integración y Ejecución***

Se lanza el sistema por primera vez, requiriendo seguimiento para corregir posibles imperfecciones menores. Tras solucionar los últimos detalles, estará listo para los usuarios finales.

***Operación y Mantenimiento***

Como en todo proceso, es crucial hacer un seguimiento y mantenimiento regulares para garantizar el correcto funcionamiento y óptimas condiciones del sistema.

**Figura 18***Ciclo de vida de un sistema*

Fuente: (El ciclo SDLC en 7 fases - Viewnext, s. f.).

## **Requerimientos de Sistema**

Es fundamental comprender, al iniciar un proyecto de software, las necesidades del cliente mediante la creación de un documento detallado. Este documento facilita la planificación, desarrollo y pruebas del sistema, aumentando las posibilidades de éxito al garantizar que el proyecto cumpla con las expectativas iniciales del cliente.

### ***Requerimientos Funcionales***

Los requisitos de un sistema especifican sus funcionalidades y comportamientos. Indican las tareas generales, procesos específicos y la interacción usuario-sistema. Permiten al software responder a diferentes entradas para satisfacer las necesidades del usuario.

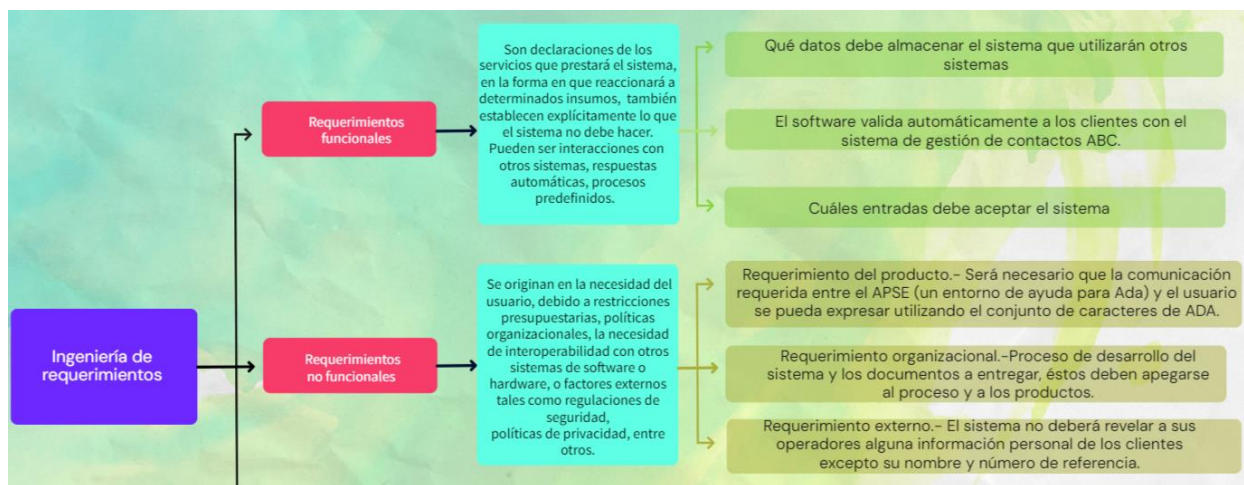
### ***Requerimientos no Funcionales***

Abordan aspectos que no están directamente relacionados con las funcionalidades específicas del sistema, pero igualmente importantes para su operatividad y calidad. Estos incluyen criterios como el rendimiento, la seguridad, la usabilidad y la compatibilidad del sistema. Los

requerimientos no funcionales definen cómo debe comportarse el sistema en diversas condiciones y establecen los estándares necesarios para evaluar su eficiencia y eficacia.

## Figura 19

### Requerimientos del sistema



Fuente: (3.1 *Requerimientos funcionales y no funcionales*, s. f.)

## Diseño Integral de Sistemas

El objetivo del diseño integral es crear soluciones que sean coherentes y efectivas para todos los aspectos, garantizando así que el producto o sistema cumpla con las expectativas y necesidades de los usuarios (Diseño Integral - FasterCapital, n. d.).

## Arquitectura

Es la estructura base de un sistema de software. Es la que define cómo se organizan e interactúan sus componentes. Este concepto es muy importante para garantizar que el sistema sea escalable, flexible y adaptable a las necesidades de su proyecto. Los tipos de arquitectura más conocidos son:

**Monolítica:** todos los componentes del sistema están integrados en un solo bloque de código. Este código es simple de implementar, pero puede volverse complejo y difícil de mantener a medida que el sistema crece.

**Capas:** organización del sistema en distintos niveles de importancia, cada capa cumple roles específicos y solo interactúa con los estratos cercanos. Este enfoque se utiliza combinado para dividir responsabilidades y lograr escalabilidad.

**Microservicios:** esta arquitectura tiene una tendencia a dividir el sistema en múltiples pequeños y completamente independientes. Cada uno ejecuta una tarea determinada, específica. Un diseño como este aporta una gran flexibilidad por la escalabilidad, pero tiene una dificultad de administración.

**Eventos:** este método se fundamenta en la interacción entre componentes mediante eventos, lo que posibilita un gran nivel de separación entre los diversos módulos del sistema.

**Cliente-Servidor:** el sistema se separa en dos partes, el cliente pide servicios y el servidor los brinda. Este tipo de sistema es muy frecuente en aplicaciones web y sistemas distribuidos.

**Componentes:** así, esta armoniosa estructura, que favorece la utilización de herramientas modulares, posibilita la creación de sistemas complejos mediante el uso de bloques ya existentes.

**Orientada a Servicios (SOA:** el sistema ofrece servicios reutilizables para crear aplicaciones grandes. Ideal para entornos corporativos y sistemas complejos. La arquitectura adecuada depende de las especificaciones y requisitos técnicos del proyecto (Arquitectura de Software: ¿Qué Es y Qué Tipos Hay?, Gluo, 2024, n. d.).

### ***Estructura de Datos***

Existen varios tipos de arquitecturas de datos que se utilizan para gestionar y organizar la información dentro de una organización, cada uno con características y aplicaciones específicas:

**Arquitectura de datos centralizada:** en este modelo, todos los datos se almacenan en un único repositorio central, lo que facilita su administración y control. Sin embargo, puede convertirse en un punto único de falla si no se gestiona adecuadamente.

**Arquitectura de datos descentralizada:** los datos se distribuyen entre varios sistemas o ubicaciones, lo que mejora la disponibilidad y reduce los riesgos de pérdida de datos, pero puede generar desafíos en la integración y consistencia de la información.

**Arquitectura de datos federada:** este enfoque combina aspectos de las arquitecturas centralizadas y descentralizadas, permitiendo que los datos se almacenen en múltiples ubicaciones, pero manteniendo una vista unificada y coherente de los mismos.

**Arquitectura de datos orientada a servicios (SOA):** en este modelo, los datos se gestionan como servicios independientes que pueden ser reutilizados y combinados para formar aplicaciones más grandes y complejas. Es especialmente útil en entornos empresariales donde se requieren altos niveles de flexibilidad y escalabilidad.

**Arquitectura de datos en la nube:** la arquitectura en la nube gestiona eficientemente grandes volúmenes de datos con ventajas en escalabilidad y costos, pero enfrenta desafíos en seguridad y cumplimiento normativo. También es crucial para la integración y gobernanza de datos, alineando la información con los objetivos del negocio y los requisitos de seguridad (Arquitectura de Datos: ¿Qué Es? Importancia, Tipos y Usos, Astera, 2024, n. d.).

### ***Interface***

Las interfaces gráficas de usuario (GUI) facilitan en el desarrollo de software la interacción a través de elementos visuales como botones y ventanas en lugar de comandos de texto. El diseño de una GUI es crucial para una experiencia satisfactoria, considerando aspectos como la disposición visual, coherencia de elementos y claridad en la presentación de información para

garantizar eficiencia y fluidez en la interacción. (¿Qué es una interfaz? | KeepCoding Bootcamps, s. f.; ¿Qué Es Interfaz Gráfica de Usuario y Para Que Sirve? NeoAttack, 2024, n. d.).

### **Componentes**

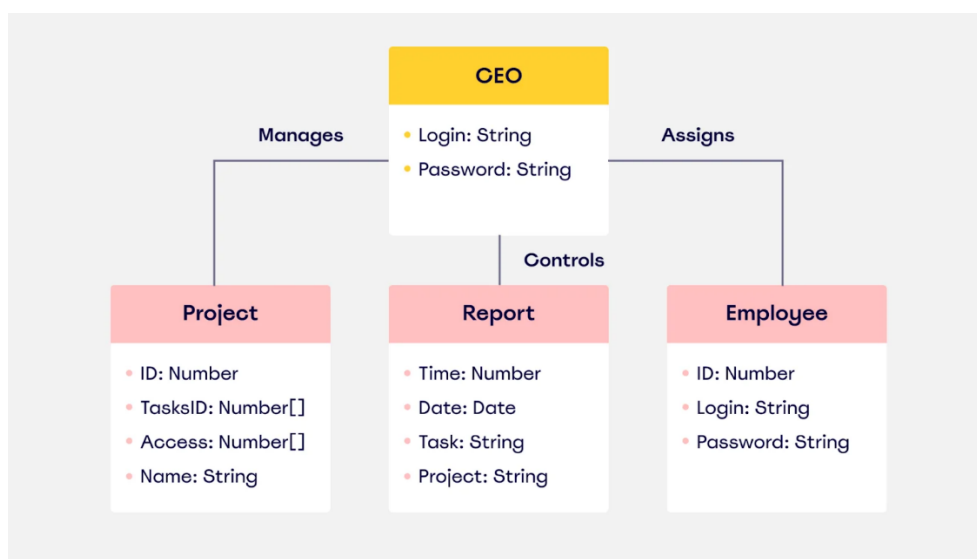
Los componentes pueden actualizarse o reemplazados de forma independiente, mejorando la flexibilidad y adaptabilidad del sistema. La arquitectura de componentes es útil en sistemas complejos con alta cohesión y bajo acoplamiento entre partes (Arquitectura de Componentes - DEV Community, Vanessa Marely. (2024), n. d.).

### **Diagrama UML**

Un diagrama visual muestra la estructura y organización de los componentes en un sistema de software, destacando cómo se conectan y colaboran módulos y bibliotecas. Facilita la planificación y comprensión de la arquitectura del software al resaltar interfaces y dependencias, identificando mejoras y asegurando una integración efectiva (Diagrama de Componentes UML: Qué Es y Cómo Hacerlo | Miro, 2024, n. d.).

### **Figura 20**

#### *Diagrama UML*



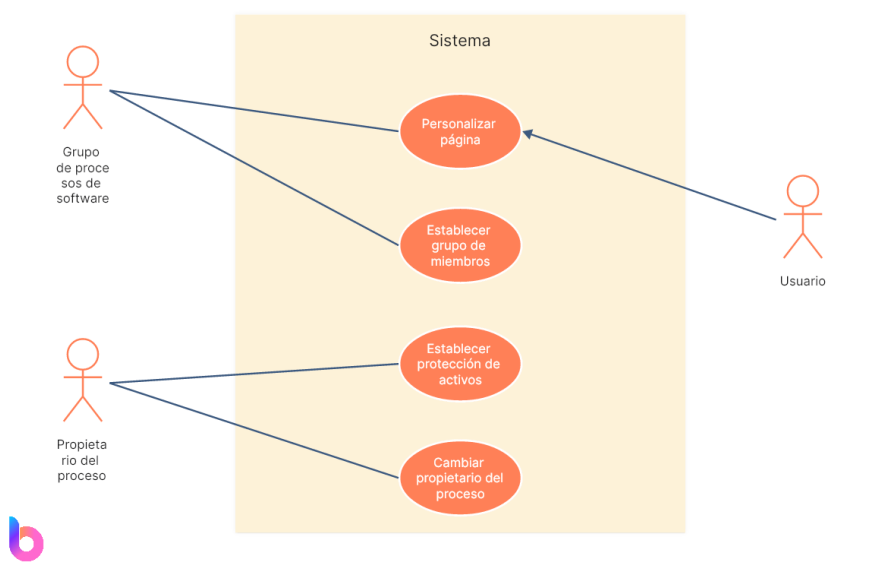
Fuente: (Diagrama de Componentes UML: Qué Es y Cómo Hacerlo | Miro, 2024, n. d.)

## Casos de Uso

Son una técnica de modelado que describe cómo los usuarios interactúan con un sistema para alcanzar un objetivo específico. Cada caso de uso define un conjunto de acciones que el sistema debe realizar para satisfacer las necesidades de los usuarios o actores en un escenario particular (*Definición de casos de uso - Documentación de IBM, s. f.-a*).

### Figura 21

#### Casos de uso



Fuente:(8 ejemplos de diagramas de casos de uso para inspirarse / Boardmix, s. f.)

## Base de Datos

### Historia de las Bases de Datos

Se remonta a los años 60 cuando surgieron los primeros sistemas de almacenamiento de datos. Inicialmente, los sistemas jerárquicos y en red fueron predominantes, pero su rigidez y complejidad llevaron al desarrollo de modelos más flexibles. En los años 70, el modelo relacional, fue propuesto por Edgar F. (Historia de Bases de Datos: Evolución y Orígenes - jhonmosquera.com, s. f.; Un poco de historia sobre las bases de datos - ADR Formación, s. f.).

## Figura 22

### *Base de datos*



Fuente: (Bases de datos y su importancia en las empresas-Gestión de base de datos, s. f.)

### *Qué son las Bases de datos de Código Abierto (Open Source)*

Las bases de datos de código abierto proporcionan acceso gratuito a su código fuente, lo que permite a los usuarios ver, modificar y redistribuir el software. Además, la comunidad de desarrolladores global asegura la actualización y el cumplimiento de altos estándares de seguridad en estas bases de datos.

**MySQL:** es un sistema de gestión de bases de datos relacional (RDBMS) de código abierto que emplea el lenguaje SQL para administrar y recuperar datos de manera eficiente. Este sistema es ampliamente adoptado en el desarrollo de aplicaciones web debido a su capacidad para manejar grandes volúmenes de información y su compatibilidad con diversas plataformas. (¿Qué es MySQL? Explicación y características | Blog de Arsys, s. f.).

Figura 23

*Qué es MySQL*

**QUÉ ES MySQL**

Es el **sistema de gestión de base de datos relaciones (RDBMS)** más usado en el mundo.

**Multiplataforma** (Windows, Apple, Linux)

**Doble licencia:** Open Source y comercial con soporte (desde que fue adquirida por Oracle).

Tiene una **sencilla curva de aprendizaje** y responde muy rápido las consultas.

Es la **más usada en el desarrollo web** gracias a su integración con PHP.

**Soporta** procedimientos almacenados, triggers y funciones.

**HERRAMIENTAS COMPLEMENTARIAS:**

- phpMyAdmin:** Administrador de bases de datos MySQL escrito en PHP.
- MySQLWorkbench:** Sirve para diseñar, modelar y administrar visualmente las BD.
- MySQLDump:** realiza copias de seguridad lógicas.

Aprende a administrar bases de datos MySQL en:  
[ed.team/cursos/mysql](https://ed.team/cursos/mysql)

EDteam

Fuente: (Curso: MySQL (Comienza Gratis) | EDteam, n. d.)

**PostgreSQL:** También conocido como “Postgres”, es un sistema de gestión de bases de datos objeto-relacional que cumple con los estándares de SQL. Por otro lado, MySQL se destaca por su facilidad de uso, siendo ideal para proyectos más pequeños. Sin embargo, para startups con previsiones de crecimiento en la carga de datos, la migración de MySQL a PostgreSQL puede ser la mejor opción. (Las mejores bases de datos open source - Codemotion, s. f.).

**Redis:** destaca por su capacidad para realizar operaciones de lectura y escritura a altas velocidades, lo cual es ideal para aplicaciones que requieren de un rápido acceso a los datos. Redis es compatible con más de 50 lenguajes de programación y ofrece una API de módulos que permite a los desarrolladores crear extensiones personalizadas. En estos casos, es aconsejable combinar Redis con otras bases de datos para manejar cargas de trabajo adicionales y proporcionar una solución más robusta. (Las mejores bases de datos open source - Codemotion, s. f.).

**MongoDB:** es una base de datos No-SQL que se caracteriza por su diseño basado en documentos, lo que contrasta con el enfoque tradicional de almacenamiento en tablas utilizado en bases de datos relacionales. Esta base de datos emplea el MongoDB Query Language (MQL) para operar con los datos, lo que permite una mayor flexibilidad en el almacenamiento y la consulta de información. (Las mejores bases de datos open source - Codemotion, s. f.).

**Neo4j:** es una base de datos de grafos NoSQL diseñada para administrar, consultar y almacenar datos en forma de grafos en tiempo real. Su estructura basada en nodos y relaciones permite una gestión eficiente de componentes de datos, facilitando el análisis y la navegación de estos datos. (Las mejores bases de datos open source - Codemotion, s. f.).

## Figura 24

*Bases de datos open source*



Fuente: <https://guidocutipa.blog.bo/8-mejores-sistemas-administracion-base-datos-codigo-abierto/>

## Metodologías de Desarrollo

Estas metodologías se dividen en varios tipos, entre los cuales destacan las tradicionales, como el modelo en cascada y las ágiles. El modelo en cascada es secuencial, con fases bien definidas, lo que facilita la gestión de proyectos con requisitos estables.

### Metodologías Tradicionales

#### *Cascada*

Es un enfoque tradicional y lineal en el desarrollo de software, en el cual cada fase del proyecto se completa secuencialmente antes de pasar a la siguiente. (Metodología en cascada: Ventajas e inconvenientes | SafetyCulture, s. f.).

#### *Incremental*

Es una metodología en la que el sistema se construye en partes o incrementos sucesivos, cada uno de los cuales proporciona una porción funcional del producto final. La entrega continua

de incrementos funcionales también permite una validación temprana del sistema y asegura que los componentes se integren de manera efectiva a lo largo del desarrollo. (¿Qué es el desarrollo incremental? | PhoenixNAP Glosario de TI, s. f.).

### ***Prototipo***

Este enfoque permite a los desarrolladores y a las partes interesadas explorar conceptos, identificar problemas y hacer ajustes en etapas tempranas del proyecto. Al utilizar prototipos, los equipos pueden obtener retroalimentación continua, refinando y ajustando el diseño según las necesidades del usuario. (Prototipo – el papel del prototipado en el proceso de desarrollo de productos - Copymate, s. f.).

### **Metodologías Ágiles**

#### ***SCRUM***

Pueden abordar problemas complejos a través de esta metodología las personas, a la vez que se entregan productos de forma eficiente y creativa con el máximo valor. Además, Scrum es una metodología que ayuda a los equipos a colaborar y realizar un trabajo de alto impacto. Por otra parte, debe saberse que en Scrum trabaja con equipos pequeños multidisciplinares en ciclos iterativos centrados en el cliente y se crea un producto de forma incremental.(Scrum: conceptos clave y cómo se aplica en la gestión de proyectos [2024] • Asana, s. f.).

#### ***Kanban***

Es una metodología ágil que utiliza un sistema de gestión de tareas. Su propósito principal es supervisar y optimizar el flujo de trabajo desde el inicio hasta el final de las tareas, asegurando un proceso continuo y eficiente. Kanban, una palabra japonesa que combina, refleja la esencia del método: el uso de tarjetas visuales para representar tareas y su estado dentro de un proyecto. (Metodología Kanban: en qué consiste y cómo utilizarla | APD, s. f.).

### ***Programación Extrema (Extreme programming, XP)***

Es una metodología ágil de gestión de proyectos que se concentra en la velocidad y la simplicidad con ciclos de desarrollo cortos y con menos documentación. Como las otras metodologías ágiles, la programación extrema es un método de desarrollo de software dividido en sprints de trabajo. Del mismo modo que otros métodos ágiles, este diseño permite a los desarrolladores responder a las solicitudes de los clientes, adaptarse y realizar cambios en tiempo real. (¿Qué es la programación extrema (XP)? [2024] • Asana, s. f.).

### **Figura 25**

*Metodologías ágiles y tradicional.*



Fuente: ((25) Explorando Las Similitudes Entre Metodologías Ágiles y Desarrollo Tradicional. | LinkedIn, n. d.).

### **¿Qué es el Desarrollo Web?**

El desarrollo web es un proceso que abarca la creación, el diseño y el mantenimiento de un sitio web. Dado que el desarrollo web abarca muchos componentes, se refiere principalmente a la programación de sitios web. Por supuesto, la alineación social de las páginas no solo implica esto, sino también que abarca la programación, la seguridad cibernética y otras cuestiones de

mantenimiento. Un desarrollador web, por lo tanto, combina habilidades técnicas y artísticas para construir las bases en las que se basan las actividades informáticas modernas. Algunos ejemplos comunes de estos sitios son tiendas en línea y servidores de redes sociales que dependen de miles de líneas de código.

### **¿Qué Hace un Desarrollador Web?**

El desarrollador web, también conocido como ingeniero o programador web, es un profesional multifuncional especializado en sitio y aplicación web, también es responsable del aspecto de este último. En particular, este especialista se ocupa del diseño y desarrollo de soluciones de software basadas en web y planea la arquitectura del sitio, desarrollo del sitio, mantenimiento y optimización posterior del sitio. Las responsabilidades más usuales serían:

- Reunirse con los clientes para discutir el diseño del proyecto, las necesidades técnicas.
- Desarrollar y probar aplicaciones web y, en general, escribir el código web.
- Trabajar con un grupo mayor integrando elementos adicionales, como gráficos, audio y video y desarrollar una gama más amplia de funcionalidades, dependiendo del cliente
- Supervisar la velocidad y capacidad del sitio.

**Figura 26**

*Que Hace un Desarrollador web*



Fuente: (Perfil y Salario de un Perfil de Desarrollador Web en México en 2024 - Hireline México, s. f.).

## **Tipos de Desarrollo web**

### ***Front-end (interfaz)***

El desarrollo front-end se centra en el aspecto y la experiencia que un usuario puede experimentar en un sitio web. De esta manera, los desarrolladores front-end deben mantenerse actualizados sobre las tendencias en diseño web y desarrollo, para asegurarse que los sitios web estén optimizados para los consumidores y la búsqueda, así como se apegan a las mejores prácticas de seguridad informática.

## **Lenguajes de Desarrollo**

**HTML:** es un lenguaje de gran importancia para controlar la estructura de cualquier sitio web. HTML tiene elementos básicos como encabezados, párrafos e imágenes.

**CSS:** esta es la presentación visual, permite a los desarrolladores dar estilo a los elementos con diversos colores, fuentes y diseños. Garantiza que el sitio sea atractivo y funcione bien en todos los dispositivos.

**JavaScript:** este es el lenguaje se utiliza para hacer interactivas las páginas web, permitiendo contenidos dinámicos como formularios y animaciones.

### **Funciones:**

- Crear una estructura visual
- Presentar datos e información
- Usar el diseño responsive
- Optimizar el rendimiento

### **Backend (servidor)**

El desarrollo backend o del lado del servidor, funciona con los servidores, bases de datos y lenguajes de programación tales como Python, PHP o Java. Un desarrollador backend sabe crear y mantener aplicaciones, procesar solicitudes desde el frontend, actualizar la base de datos e interactuar con otros sistemas.

## **Lenguajes de Desarrollo**

**Python:** es conocido por su claridad y eficacia, Python es el lenguaje más utilizado para el desarrollo web back-end, con marcos como Django y Flask que permiten un desarrollo rápido.

**Java:** es un lenguaje orientado a objetos utilizado para aplicaciones a gran escala, Java es valorado por su alto rendimiento y fiabilidad en diferentes plataformas.

**PHP:** es manejado para contenidos web dinámicos y para integrar bases de datos, PHP es compatible con muchos sistemas de gestión de contenidos.

**Ruby:** es mayor mente usado por su elegante sintaxis, Ruby y el framework Rails ofrecen un método sencillo para crear aplicaciones respaldadas por bases de datos.

**Node.js:** es un tiempo de ejecución de JavaScript que permite a los desarrolladores utilizar JavaScript en el lado del servidor, ayudando al desarrollo de aplicaciones de red rápidas y escalables.

#### **Funciones:**

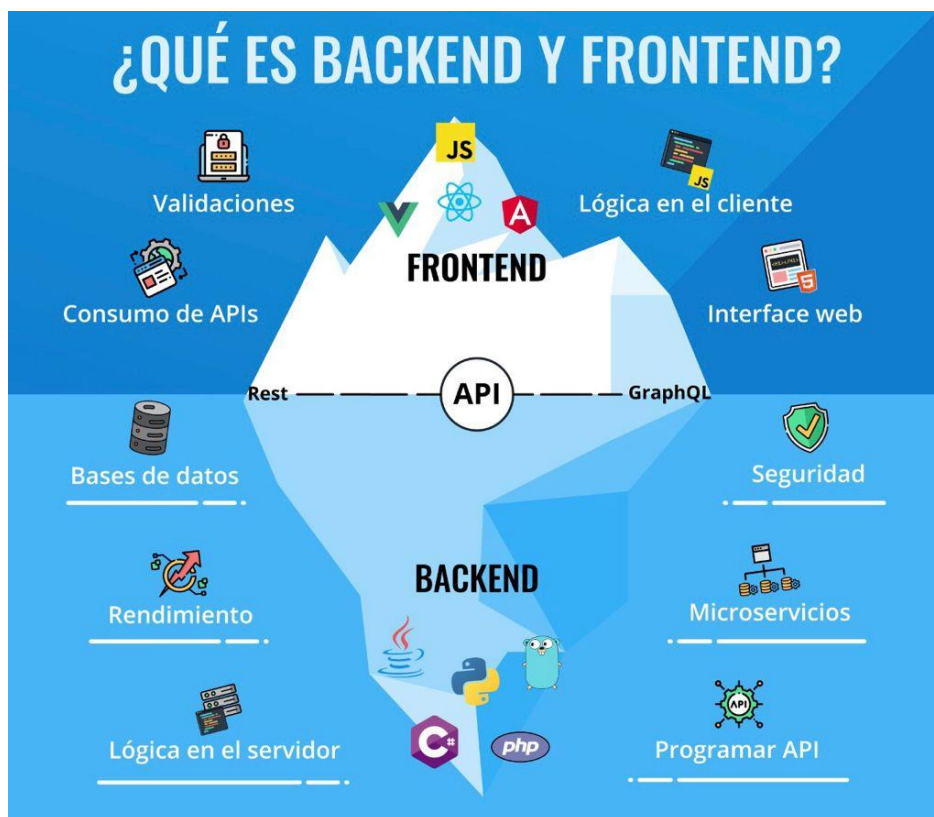
- Gestionar la lógica de negocio
- Administrar el almacenaje y recuperación de datos
- Implementar mecanismos de seguridad
- Rastrear y gestionar las sesiones de los usuarios
- Optimizar el rendimiento

#### **Full Stack**

Los desarrolladores full stack tienen el conocimiento y las habilidades para ir en ambas direcciones, ya sea hacia el frontend o el backend. Además, el desarrollo full stack se ocupa tanto de la parte del cliente como del servidor del sitio. Un buen desarrollador conoce los lenguajes de programación, las tecnologías y los frameworks más importantes, y participa en todo el ciclo de desarrollo web.

**Figura 27**

*¿Qué es frontend y backend?*



Fuente: (*¿Qué es frontend y backend?* - Ecdisis Estudio, s. f.)

## Ventajas y Desventajas del Desarrollo Web

### Ventajas

- Se tiene acceso a múltiples herramientas y marcos que les permiten diseñar soluciones únicas y flexibles.
- Gracias a los recursos modernos como los microservicios, ahora los sitios web pueden gestionar con mayor eficacia el aumento de datos y de usuarios.
- Las técnicas de desarrollo eficaces mejoran el acceso a los sitios web y aumentan el alcance de la audiencia.

- Las empresas que buscan progresar con sus ventas online pueden beneficiarse de un desarrollo web sólido.

### ***Desventajas***

- Debido a que la creación de sitios web de alta calidad requiere de conocimientos específicos, puede resultar costoso.
- Los sitios web que tiene un mal diseño pueden no funcionar bien o cargarse con lentitud.
- Pueden surgir proyectos complejos por la necesidad de gestionar una amplia gama de herramientas y tecnología.
- Es necesario actualizar la seguridad de los sitios web, dado que pueden ser blanco de ataques.

## **Herramientas Para Desarrollo del Proyecto**

### ***IDE***

Un Entorno de Desarrollo Integrado (IDE) es una herramienta esencial para programadores y desarrolladores de software que facilita la escritura, depuración y administración del código. Un IDE combina varias funciones en una sola interfaz, incluyendo un editor de código, herramientas de depuración, y opciones de compilación, lo que permite a los usuarios escribir código de manera más eficiente y efectiva. (IDE: ¿qué es un entorno de desarrollo integrado?, s. f.; IDEs: Qué son, para qué sirven y los más populares - Blog de SW Hosting, s. f.).

### ***Editores de Código***

Los editores de código son herramientas esenciales para programadores que facilitan la escritura y modificación del código. Una de sus ventajas es la codificación con colores según el tipo de datos y la sugerencia automática de completar líneas. Además, permiten una fácil

navegación entre líneas y archivos, lo que agiliza la corrección de errores. Estas características hacen que los programadores puedan trabajar de manera más eficiente y precisa (Los 17 Mejores Editores de Código, n. d.; Explorar las características del editor de código - Visual Studio (Windows) | Microsoft Learn, s. f.).

## Figura 28

### *Editor vs IDE*



Fuente: (🗉 Editor vs IDE | EDteam, n. d.).

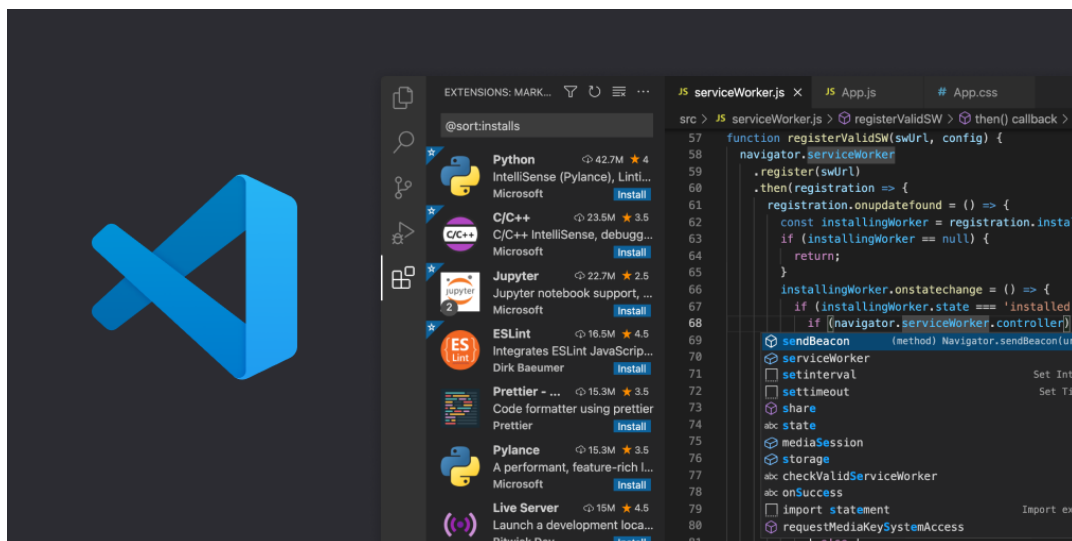
### *Visual Studio Code*

Visual Studio Code es una herramienta altamente personalizable, debido a su variedad de extensiones. Esto permite a los desarrolladores adaptarla a sus necesidades, convirtiéndola en una

opción confiable, flexible y eficiente para cualquier proyecto, independientemente del sistema operativo utilizado. (¿Qué Es Visual Studio Code y Cuáles Son Sus Ventajas? | Blog de Arsys, n. d.).

## Figura 29

*Visual Studio Code*



Fuente: (Visual Studio Code - Code Editing. Redefined, s. f.)

## Java

Estas características hacen de Java una opción popular para el desarrollo de aplicaciones empresariales, móviles y web. La versatilidad de Java y su capacidad para manejar entornos complejos lo convierten en una herramienta esencial en el desarrollo de software contemporáneo. (Java: ¿Qué es y para qué sirve?, s. f.).

**Figura 30***Java*

Fuente:(Visual Studio Code - Code Editing. Redefined, n. d.).

***Spring Boot***

Es un marco de desarrollo basado en Java que facilita la creación de aplicaciones empresariales robustas y eficientes al simplificar la configuración y despliegue de aplicaciones Spring. (¿Qué es Spring Boot y cómo se desarrollan aplicaciones?, s. f.).

**Figura 31***Spring boot*

Fuente: (Spring-Boot-Logo | Yes, n. d.).

## *Thymeleaf*

Es un motor de plantillas para Java que permite el uso de condicionales en las vistas HTML, facilitando la personalización dinámica del contenido. Las directivas `th:if` y `th:unless` permiten controlar la visibilidad de los elementos en función de condiciones específicas, mientras que `th:switch` y `th:case` se utilizan para seleccionar entre múltiples opciones. (Conditionals in Thymeleaf | Baeldung, n. d.).

### **Figura 32**

*Thymeleaf*



Fuente: (Qué Es Thymeleaf | OpenWebinars, n. d.).

## ***HTML***

El HTML (HyperText Markup Language) es el lenguaje fundamental para la creación de páginas web, proporcionando la estructura básica de los documentos a través de elementos y etiquetas. Cada página web se compone de una serie de elementos HTML que definen la estructura y el contenido, como encabezados, párrafos, listas y enlaces. (Conceptos Básicos de HTML - Aprende Desarrollo Web | MDN, n. d.).

## ***CSS***

Es un lenguaje utilizado para definir y gestionar la presentación visual de los documentos HTML en la web. Este lenguaje permite separar el contenido de la presentación, facilitando así la

modificación de la apariencia de un sitio web sin alterar su estructura subyacente. (¿Qué Es CSS y Para Qué Sirve? - Webempresa, n. d.).

### ***JavaScript***

JavaScript es esencial para el desarrollo web, permitiendo agregar funcionalidades dinámicas a los sitios. A diferencia de HTML y CSS, que se enfocan en la estructura y diseño, JavaScript se encarga de la lógica y manipulación de elementos según las acciones del usuario. (Fundamentos de JavaScript - Aprende desarrollo web | MDN, s. f.) .

### **Figura 33**

*HTML, CSS, JS*



Fuente: (Crear Un Sitio Web Html Usando Html, Css y Javascript, n. d.).

### ***Bootstrap***

Un framework de código abierto que ofrece herramientas predefinidas como cuadrículas y diseños de botones para crear sitios web de alta calidad de manera rápida y uniforme. Su popularidad se debe a su compatibilidad y facilidad de integración, convirtiéndolo en una opción común para proyectos web eficientes y modernos. (Bootstrap: qué es, para qué sirve y cómo usarlo | Blog de Arsys, s. f.).

**Figura 34***Bootstrap*

Fuente:(Bootstrap Blog · Official blog for the Bootstrap framework., s. f.).

### **Capítulo III. Marco Metodológico**

## ¿Qué es un Marco Metodológico?

Ortega (2024) menciona en su artículo que, un marco metodológico es una estructura o conjunto de reglas y principios que guían el proceso de investigación. Es un mapa que ayuda a los investigadores a planificar y llevar a cabo su estudio de manera organizada y sistemática. Estos son algunos puntos claves para el marco metodológico:

- **Diseño de investigación:** el marco metodológico contiene el diseño general de la investigación, es decir, la manera como se va a recopilar y analizar la información, por ejemplo, si es un estudio experimental, observacional, de caso, etc.
- **Métodos de recopilación de datos:** describe las herramientas y técnicas que se utilizarán para recopilar la información. Esto puede incluir encuestas, entrevistas, experimentos o análisis de documentos, entre otros.
- **Población y muestra:** indica quiénes son los participantes o elementos de estudio y cómo se seleccionará un subconjunto relevante para realizar la investigación.
- **Procedimiento:** define los pasos específicos que se seguirán para llevar a cabo la investigación para proporcionar ayuda, asegurar la consistencia y la transparencia del estudio.
- **Instrumentos de medición:** si se utilizan encuestas, test u otros instrumentos, el marco metodológico describe cómo se desarrollarán y validarán para garantizar su fiabilidad.
- **Análisis de datos:** explica cómo se procesarán y analizarán los datos recopilados. Esto puede incluir el uso de software estadístico, métodos cualitativos o una combinación de ambos, según el tipo de estudio.

- **Ética:** considera los aspectos éticos, como la privacidad y el consentimiento de los participantes, para garantizar que la investigación se realice de manera ética y respetuosa.

### Figura 35

#### *Marco Metodológico*



Fuente:(Metodología de la Investigación: Marco Metodológico, s. f.)

### Importancia de un Marco Metodológico

Como se mencionó anteriormente un marco metodológico es de vital importancia en una investigación por varias razones, no solo guía el proceso de investigación, también mejora la calidad y la confiabilidad de los resultados, permitiendo a otros investigadores comprender, replicar y construir sobre el trabajo realizado. Estas son algunas de las importancias de tener esta estructura:

- **Organización y estructura:** el marco metodológico proporciona una estructura clara y organizada para la investigación. Define los pasos a seguir, desde la selección de métodos hasta la interpretación de los resultados, lo que ayuda a mantener la coherencia y la lógica en todo el proceso.
- **Rigurosidad científica:** un marco metodológico sólido ayuda a la rigurosidad científica de la investigación. Establece reglas y normas que aseguran que los

métodos utilizados sean apropiados para abordar las preguntas de investigación y que los resultados sean fiables y válidos.

- **Reproducibilidad y generalización:** al seguir un marco metodológico, se permitirá la reproducción del estudio por otros investigadores. Esto es algo esencial para validar los resultados y generalizar las conclusiones a otras poblaciones o contextos.
- **Claridad en la planificación:** esto es para ayudar a los investigadores a planificar sus estudios de manera eficiente. Al definir los métodos, las muestras y los procedimientos por adelantado, se minimiza la posibilidad de errores o malentendidos durante la implementación de la investigación.
- **Énfasis en la pertinencia:** es para tener un buen marco metodológico y que se pueda adaptar a las características específicas de la investigación. Permite a los investigadores seleccionar métodos y técnicas que sean las más adecuadas para abordar sus preguntas de investigación, aumentando así la calidad y relevancia de los resultados.
- **Control de variables:** facilita el control de las variables relevantes. Al establecer los criterios específicos para la selección de las muestras, la recopilación de datos y el análisis, el marco metodológico ayuda a minimizar la influencia de variables no deseadas en los resultados.
- **Ética y responsabilidad:** esta parte incluye consideraciones éticas que protegen los derechos y el bienestar de los participantes. Esto es algo esencial para realizar las investigaciones de manera responsable y respetuosa.

## **Desarrollo de un Marco Metodológico**

### ***Definición de la Investigación***

**Objetivos de investigación:** se especifica claramente lo que se pretende lograr con la investigación. Define los objetivos generales y específicos.

### ***Revisión de la Literatura***

**Marco teórico:** se realiza una revisión exhaustiva sobre la literatura que existente sobre el tema. Aquí se identifican las teorías, conceptos y los modelos relevantes que sirvan como base para la investigación.

### ***Diseño de Investigación***

**Tipo de Investigación:** en esta parte se decide si la investigación será descriptiva, exploratoria, experimental y correlacional, entre otros.

**Enfoque:** se elige entre un enfoque cuantitativo, cualitativo o mixto, esto dependiendo de la naturaleza de las preguntas de investigación.

### ***Población y Muestra***

**Población de estudio:** se identifica el grupo o población que será objeto de la investigación.

**Criterios de inclusión/exclusión:** se definen los criterios para seleccionar participantes.

### ***Instrumentos de Recopilación de Datos***

**Selección de instrumentos:** en esta parte se decide qué herramientas se usarán para la recopilación de datos, por ejemplo: encuestas, entrevistas, observaciones, pruebas, etc.

**Validación y fiabilidad:** si es necesario, se deben de válida y verifica la confiabilidad de los instrumentos.

### *Procedimiento*

**Pasos detallados:** esto describe de manera detallada los pasos que se deben de seguir para llevar a cabo la investigación, desde la selección de la muestra hasta la recopilación y análisis de los datos.

### *Análisis de Datos*

**Métodos de análisis:** se especifican las técnicas estadísticas o cualitativas que se utilizarán para analizar los datos.

**Interpretación de resultados:** se muestra cómo se interpretarán los resultados obtenidos.

### *Aspectos Éticos*

**Consideraciones éticas:** para asegurarse de abordar aspectos éticos, como el consentimiento informado, la confidencialidad y el respeto a los participantes.

### *Plan de Trabajo*

**Cronograma:** se elabora un cronograma detallado que indique las etapas clave de la investigación y los plazos para cada una.

### *Presupuesto*

**Recursos necesarios:** se estiman los recursos necesarios, como personal, equipo, material. Si es posible, también se incluye un presupuesto.

### *Validación del Marco Metodológico*

**Revisión por expertos:** se considera al obtener retroalimentación de colegas o expertos en el campo para validar la capacidad de tu marco metodológico.

### *Redacción del Marco Metodológico*

**Estructura clara:** redacta el marco metodológico de forma clara y organizada, siguiendo la estructura tradicional de introducción, objetivos, diseño de investigación, población y muestra.

### ***Revisión Continua***

**Adaptaciones:** a medida que se avanza con la investigación, se necesita ser más flexible para realizar los ajustes en el marco metodológico si fueran necesarios.

### **Enfoque de la Investigación**

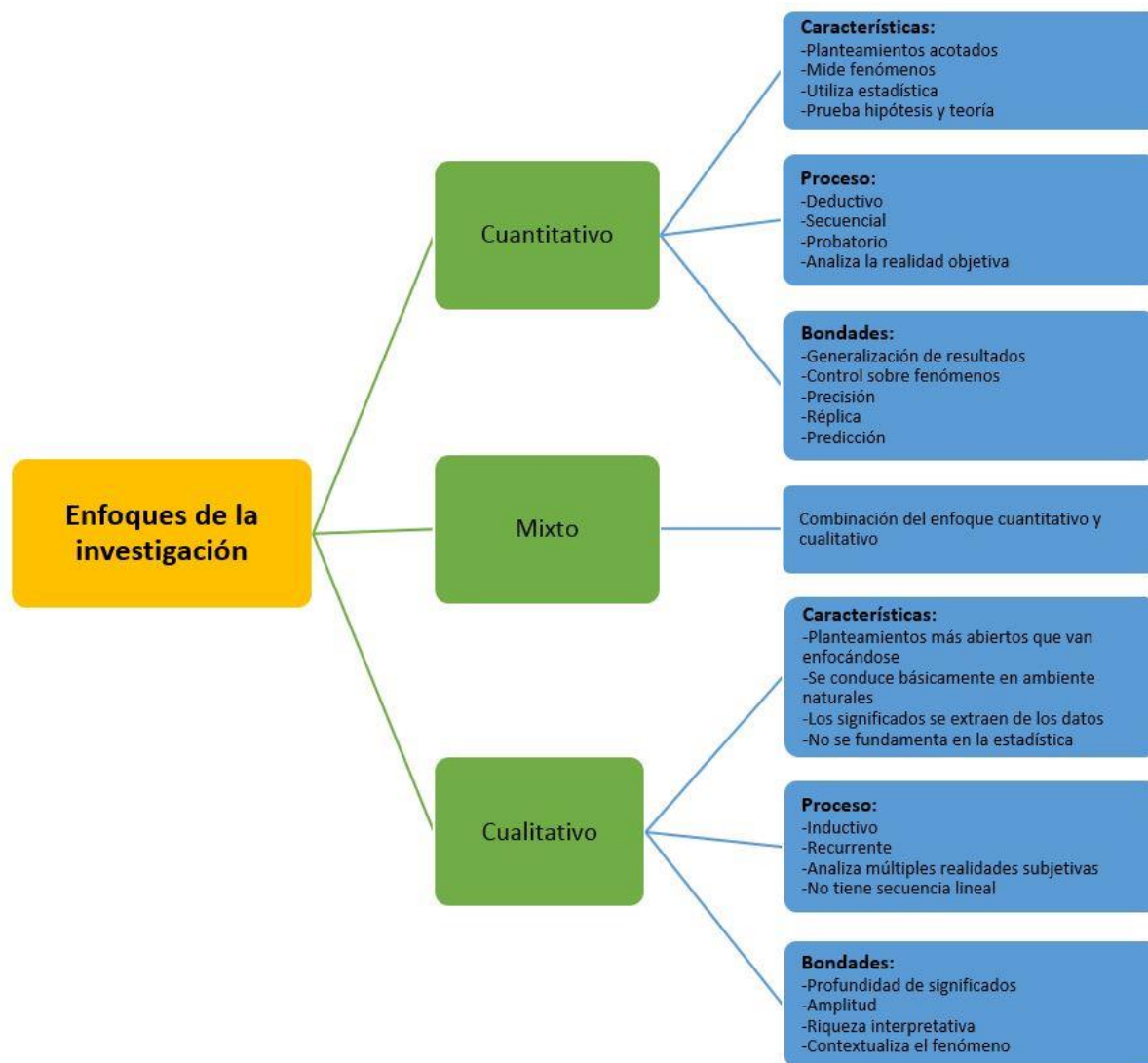
El artículo escrito por Santos (2023) nos presenta que el enfoque de una investigación está definido por los intereses del investigador, el enfoque disciplinario, la base teórica y por aquellos factores que recaen dentro del problema de la investigación y que se desarrollaron en el argumento del problema.

Es por eso por lo que el enfoque de investigación no se desarrolla hasta que se tenga por lo menos el argumento del problema y una breve base teórica relacionada con este.

Se debe de recordar que las cuestiones teóricas y metodológicas están estrechamente relacionadas y que todas las decisiones metodológicas están amparadas por decisiones teóricas.

La teoría sobre la metodología es aquella que define qué cuestiones teóricas, están relacionadas con los aspectos procedimentales de la investigación, por ejemplo, la base teórica define ¿Qué es una entrevista?, ¿Qué es un cuestionario, ¿Qué es un estudio de caso? O ¿Qué es una muestra?

Figura 36

*Enfoques de la Investigación*

Fuente: (El portal de la tesis, s. f.)

**Tipos de Enfoques*****Enfoque Cualitativo***

El enfoque cualitativo básicamente se utiliza cuando se está interesado en las cualidades del objeto investigado. Cuando el interés recae en el porqué de las cosas, las fuerzas en relación, la subjetividad de los sujetos involucrados, las relaciones entre los sujetos involucrados, las

características de estas relaciones. Y, básicamente, cuando no se está interesado en cuantificar el proceso.

### Figura 37

#### *Métodos de Investigación Cualitativa*



Fuente: (Investigación cualitativa | QuestionPro, s. f.)

#### **Características:**

- Entornos naturalistas
- Enfoque inductivo
- Perspectiva holística
- Subjetividad e interpretación
- Flexibilidad

#### ***Tipos de Investigación Cualitativa***

**Entrevistas en grupo:** las entrevistas pueden no ser individuales, también pueden ser grupales y los propósitos pueden ser muy diversos.

**Entrevistas estructuradas y no estructuradas:** se pueden contar como una guía de preguntas abiertas o no, se pueden aplicar preguntas con base en lo que dice el entrevistador.

**Métodos de observación cualitativa:** se utiliza para recopilar información sobre los comportamientos no verbales de los sujetos. Tiene la ventaja de que permite estudiar un comportamiento de forma más efectiva al realizar esta acción de manera presencial.

**Investigación etnográfica:** entre las características que diferencian este tipo de investigación están:

- La estancia prolongada dentro de la comunidad estudiada para recoger información a través de la observación.
- La elaboración de informes que dan prioridad a la contextualización y la relación interna de los fenómenos que se observan.
- La presentación de interpretaciones que combinan voluntariamente la narrativa.
- La descripción y la conceptualización teórica.

**Análisis de redes sociales:** el objetivo del análisis de redes es comprender a la comunidad mediante el mapeo de las relaciones que los conectan.

### ***Enfoque Cuantitativo***

Respecto al enfoque cuantitativo, sí se encuentra enfocado en los números, estadísticas o en procesos que son básicamente relacionados con lo cuantificable. Es un enfoque que está relacionado con el procesamiento estadístico a definiciones básicamente cuantificables.

**Figura 38***Proceso de Investigación Cuantitativa*

## Proceso de investigación cuantitativa



Fuente: (Los enfoques cuantitativo y cualitativo en la investigación | PPT, s. f.)

### Características:

- Sirve para formular leyes y generalizaciones
- Se resguarda de procedimientos específicos para obtener datos
- Arroja información sobre una población
- Se trata de un método objetivo
- Se vale de la lógica deductiva

### ***Tipos de Investigación Cuantitativa***

**Investigación descriptiva:** explica las propiedades, características y rasgos importantes del fenómeno estudiado mediante dinámicas objetivas de observación, análisis y demostración.

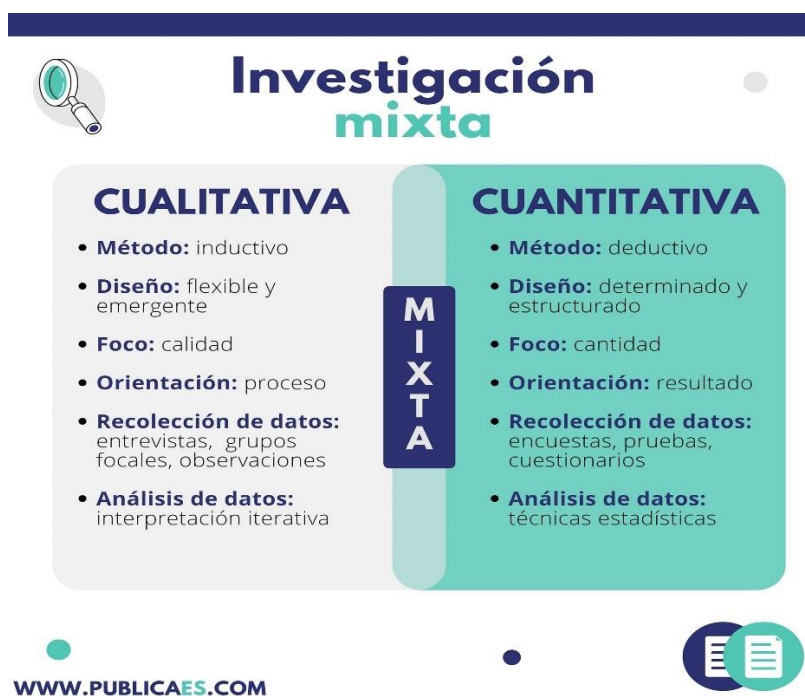
**Investigación analítica:** esto consiste en establecer una relación entre las variables del fenómeno estudiado En el cual se registran resultados para poder comprobar o refutar alguna hipótesis previamente establecida.

**Investigación experimental:** se trata de obtener información mediante un experimento controlada, de forma que se puedan comprender y eventualmente manipular las variables que determinan un fenómeno, para poder establecer cuáles son sus causas y sus efectos.

### ***Enfoque Mixto***

Este tercer enfoque básicamente se trata de investigaciones en las cuales el interés recae en hacer una interpretación como, por ejemplo, de la subjetividad de los sujetos involucrados en el problema, pero también se interesa cuantificar, analizar algunas estadísticas, aplicar algunos métodos o técnicas provenientes del enfoque cuantitativo. Puede ser mixta con enfoque en lo cuantitativo.

Figura 39

*Investigación Mixta*

Fuente: (Publicaes, s. f.)

**Características:**

- En el momento que se quiere validar o corroborar los resultados obtenidos con otros métodos.
- Cada vez que se necesita utilizar un método para informar a otro.
- Siempre que se quiera analizar continuamente una pregunta de investigación desde diferentes ángulos.
- Cuando se quiere elaborar, clarificar o construir sobre los hallazgos de otros métodos.
- Tan pronto como se quiera desarrollar una teoría sobre un fenómeno de interés y luego ponerla a prueba.
- Si se quiere generalizar los resultados de la investigación cualitativa.

### ***Tipos de Investigación Mixta***

**Diseño explicativo secuencial:** este diseño implica la recopilación y análisis de datos cuantitativos, seguido de datos cualitativos, priorizando los datos cuantitativos e integrando conclusiones durante la fase de interpretación del estudio.

**Diseño exploratorio secuencial:** este diseño prioriza la investigación cualitativa, asegurando la recolección y análisis de datos a través de la recolección y análisis de datos cuantitativos, integrando los hallazgos durante la fase de interpretación del estudio.

**Estrategia concurrente de triangulación:** este diseño utiliza una única fase de recolección y análisis de datos, separando la recolección y análisis de datos cuantitativos y cualitativos, integrando los resultados durante la fase de interpretación del estudio.

**Anidado concurrente:** este diseño de investigación mixto utiliza una fase de recopilación de datos donde un método predominante se combina con un método menos prioritario, lo que da como resultado la fusión de datos durante la fase de análisis del proyecto.

### **Método de Investigación**

Como lo indica Compilatio (2024), los métodos de investigación son procesos organizados y sistemáticos para obtener información y conocimiento sobre un tema en específico. Se basan en un conjunto de pasos y técnicas que permiten recolectar, analizar e interpretar datos de manera objetiva y confiable. Los métodos de investigación son utilizados en diversas disciplinas, tanto académicas como científicas, para abordar preguntas de investigación, resolver problemas y llegar a conclusiones basadas en evidencias.

### **Fuentes de Información**

“Las fuentes de información son documentos que se consultan para obtener datos de un tema sobre el que se quiere saber más” (Fuentes de información - Qué son, tipos y ejemplos, s. f.).

En el mundo moderno, la información es amplia y accesible, pero es difícil distinguir entre fuentes de valor y erróneas. Identificar fuentes confiables y pertinentes es difícil, pero lo más importante es empleado para trabajos académicos, escolares, informes, conferencias, noticias, guías de clase, y decisiones en empresas y organizaciones.

### ***Fuentes Primarias***

Estas provienen de la experiencia directa o la observación de los hechos. Son documentos originales o primeras impresiones de un evento o investigación. Estos serían algunas de ellas:

- Documentos históricos
- Entrevistas
- Encuestas
- Diarios de campo
- Registros médicos

Estas fuentes son importantes en una investigación porque proporcionan una comprensión directa de los fenómenos y sucesos estudiados.

### ***Fuentes Secundarias***

Las fuentes secundarias son aquellas que recogen, sintetizan y analizan información de las fuentes primarias. Son una compilación de información que otros han recogido y publicado previamente. Estos son ejemplos de fuentes secundarias.

- Los libros de texto
- Los artículos de revistas
- Los informes de investigación

Estas fuentes son útiles para contextualizar los eventos o para obtener una visión más general sobre un tema.

### ***Fuentes Terciarias***

Las fuentes terciarias, sintetizan y presentan información de las fuentes secundarias.

Estas incluyen:

- Diccionarios
- Enciclopedias
- Directorios
- Catálogos
- Otros recursos de referencia

Son de gran utilidad para obtener información general sobre un tema, identificar fuentes secundarias relevantes y comprender términos y conceptos complejos.

### **Figura 40**

#### *Fuentes de Información*



Fuente: (¡Aprende a distinguir entre fuentes primarias, secundarias y terciarias!, s. f.)

### **Otros Tipos de Fuente de Información**

Existen otras maneras de obtener información mediante diversas fuentes de información; estas se clasifican, según alcance, formato y especialización.

### ***Local, Nacional o Internacional***

Se podría encontrar información específica de una región o país, o de ámbito global, dependiendo de la ubicación geográfica de la fuente. Es importante considerar la relevancia de la información según sea el enfoque del trabajo.

### ***Textual o Digital***

Las fuentes de información pueden estar en formato impreso o digital. En la actualidad, las fuentes digitales son mayormente utilizadas debido a su facilidad de acceso y rapidez para obtener información. Sin embargo, es importante verificar la fiabilidad de la fuente y asegurarse de que no se trata de información falsa o manipulada.

### ***General o Especializada***

Las fuentes de información pueden ser generales o especializadas. Las fuentes generales suelen ofrecer información amplia y básica sobre un tema, por otro lado, las fuentes especializadas brindan información más detallada y específica. Dependiendo del tema y de los objetivos de la investigación, es de vital importante seleccionar fuentes que ofrezcan información relevante y útil.

### ***Formales o Informales***

Las fuentes de información pueden ser formales o informales. Las formales son aquellas que están publicadas por instituciones o expertos reconocidos y verificadas por revisores y editores profesionales. Algunas de las principales fuentes serían las siguientes:

- Artículos técnicos-científicos
- Informes tecnológicos
- Informes de vigilancia tecnológica
- Legislación
- Marcas

- Medios de comunicación
- Ofertas y demandas tecnológicas
- Redes sociales
- Repertorio de dirigentes
- Tendencias
- Tesis

Las informales, en cambio, incluyen blogs, redes sociales, foros y otros medios no profesionales en los cuales, cualquier persona puede publicar información. Algunas de las principales fuentes serían las siguientes:

- Clientes
- Comités y asociaciones empresariales
- Competidores
- Congresos, seminarios, jornadas
- Empresas subcontratadas
- Ferias, exposiciones, exhibiciones
- Fuentes internas de la empresa
- Inversores
- Misiones empresariales
- Proveedores
- Socios

Es importante tener en cuenta que estas fuentes de informales pueden contener información errónea.

### **¿Qué es una Variable?**

Llamamos variables a todo aquello que sea susceptible de asumir diferentes valores, bajo una metodología de investigación. Se puede decir que una variable es aquello que sometemos a comparación, con la finalidad de responder la pregunta de investigación.

Como se menciona en el artículo podemos definir al término de variable como “Una característica o cualidad, magnitud o cantidad, que puede sufrir cambios, y que es objeto de análisis, medición, manipulación o control en una investigación” (Bianconi, 2024).

En el método hipotético-deductivo es fundamental, debido a que a través de ellas se validan los aportes al conocimiento científico. En este sentido, las variables son el eje transversal del proceso de investigación.

**Figura 41***Cuadro de Variables*

OBJETIVO	VARIABLE	DIMENSION	INDICADORES	FINALIDAD
Describir el proceso logístico de las empresas del sector cervecero del Estado Zulia.	Mapa de proceso logístico	Proceso logístico	1. Servicio al cliente 2. Inventario 3. Suministro de materiales 4. Almacenamiento 5. Transporte y distribución	Describir cuales son las actividades que conforman el proceso logístico desde la perspectiva del personal logístico de las empresas del sector cervecero del estado Zulia.
Determinar las actividades estratégicas de la gestión logística en las empresas del sector cervecero del Estado Zulia.	Mapa de proceso logístico	Actividades estratégicas de la gestión logística	1. Gestión de compras y de aprovisionamientos 2. Gestión de pedidos 3. Política de inventarios 4. Política de transportes 5. Política de servicio al cliente 6. Red logística 7. Sistema de planificación	Determinar cuales son las actividades estratégicas que conforman la gestión logística desde la perspectiva del personal logístico de las empresas del sector cervecero del estado Zulia.
Determinar las actividades claves de la gestión logística en las empresas del sector cervecero del Estado Zulia.	Mapa de proceso logístico	Actividades claves de la gestión logística	1. Nivel de servicio al cliente 2. Transporte 3. Mantenimiento de inventarios 4. Procesamiento de pedidos	Conocer de acuerdo a la percepción de los clientes de las empresas del sector cervecero del estado Zulia cuáles consideran son las actividades claves que conforman la gestión logística, en contraparte a la óptica del personal logístico y los proveedores.
Determinar las actividades de apoyo de la gestión logística en las empresas del sector cervecero del Estado Zulia.	Mapa de proceso logístico	Actividades de apoyo de la gestión logística	1. Almacenamiento 2. Manejo de materiales 3. Embalaje de protección 4. Compras 5. Programación del producto 6. Mantenimiento de información	Conocer de acuerdo a la percepción de los clientes de las empresas del sector cervecero del estado Zulia cuáles consideran son las actividades de apoyo que conforman la gestión logística, en contraparte a la óptica del personal logístico y los proveedores.
Estructurar el mapa de procesos logístico de las empresas del sector cervecero del Estado Zulia.	Mapa de proceso logístico	Mapa de proceso logístico	Este objetivo se logró con los resultados de los objetivos anteriores	Establecer el mapa de procesos logístico de las empresas del sector cervecero del estado Zulia
Establecer los lineamientos estratégicos para la administración del mapa de procesos logístico en las empresas del sector cervecero del Estado Zulia.	Mapa de proceso logístico	Lineamientos estratégicos	En este objetivo se establecieron los lineamientos para la administración del mapa de procesos logístico propuesto por el investigador	Administrar el mapa de procesos logístico de las empresas del sector cervecero del estado Zulia, como estrategia para satisfacer a los clientes y fortalecer la gestión de las empresas objeto de estudio.

Fuente:(Cuadro de Operacionalización de La Variable | PDF | Logística | Inventario, s. f.)

**Requisitos Funcionales**

Los requisitos funcionales en ingeniería de software también se llaman especificación funcional. Estas son las descripciones del servicio que un software debe ofrecer, a su vez describe un sistema de software y sus componentes.

Una función no es más que una entrada al sistema del software, su comportamiento y salida. Este puede ser un cálculo, manipulación de datos, interacción del usuario, proceso de negocio o cualquier otra funcionalidad específica que defina qué función es probable que realice un sistema.

En la ingeniería de sistemas o de software, un requisito funcional puede variar desde una declaración abstracta de alto nivel de la necesidad del remitente hasta especificaciones matemáticas detalladas.

### ***Tipos de Requisitos Funcionales***

- Manejo de transacciones
- Reglas de negocios
- Requisitos de certificación
- Los requisitos de información
- Funciones administrativas
- Niveles de autorización
- Seguimiento de auditoría
- Interfaces externas
- Gestión de datos históricos
- Requisitos legales y reglamentarios

### **Requerimientos no Funcionales**

Son también conocidos como “requisitos de calidad” o “requisitos de atributos del sistema”. Los requisitos no funcionales son criterios y características que describen cómo debe ser la seguridad, el rendimiento, la usabilidad y otros aspectos de un sistema o software más allá de su funcionalidad básica.

Utilizar requisitos no funcionales es una parte muy importante al momento de desarrollar un sistema, dado que hará que el software o aplicación sea eficaz, eficiente y cumpla con las expectativas y necesidades del usuario. No obstante, se debe comprender que estos trabajan de la mano con los requisitos funcionales porque los requisitos funcionales se enfocan en el qué,

mientras que los requisitos no funcionales se enfocan en el cómo y en qué medida cumplen con ciertas cualidades.

### ***Tipos de Requisitos No Funcionales***

- Rendimiento
- Seguridad
- Usabilidad
- Disponibilidad
- Fiabilidad
- Mantenibilidad
- Portabilidad
- Escalabilidad

### ***Ventajas Requisitos No Funcionales***

- Mejora de la calidad del sistema
- Experiencia del usuario mejorada
- Optimización del rendimiento
- Cumplimiento normativo
- Facilitación de mantenibilidad
- Mayor seguridad
- Cumplimiento normativo
- Planificación y gestión efectiva
- Competitividad y reputación

### ***Desventajas de los Requisitos No Funcionales***

- Difíciles de entender e implementar

- Llevan mucho tiempo
- Son costosas de probar
- Afectan la funcionalidad del sistema si no se implementan correctamente.
- Dificultades de mantenimiento.
- Pueden llegar a generar conflictos entre requisitos.

### ***Ejemplos de Requisitos No Funcionales***

**Redes sociales:** deben de ser capaces de aumentar los usuarios por medio de cuentas y los datos del usuario tiene que estar protegidos. Además, tiene que ser versátil para todos los usuarios sin importar su idioma o cultura.

**Aplicación de banca en línea:** tanto la información del cliente como del servidor debe estar encriptada mediante el protocolo HTTPS y el código fuente debe estar bien documentado y seguir las mejores prácticas. También, la interfaz debe ser accesible y utilizable para personas con discapacidades visuales.

**Sistema de gestión de inventario:** el sistema debe tener la capacidad de funcionar durante al menos seis meses continuos sin requerir reinicios a causa de fallos y en casos aun de caída del servidor, el sistema debe tener la capaz de recuperar y restaurar los datos del inventario desde la última copia de seguridad. Además, las consultas de búsqueda y filtrado de inventario deben completarse en menos de un segundo.

**Sistema de comercio electrónico:** el sistema debe tener la capacidad de manejar un aumento del tráfico durante la temporada de ventas sin degradar el rendimiento, por lo que debe estar disponible las 24 horas del día, los siete días de la semana, con un tiempo de inactividad máximo de una hora al mes para mantenimiento del programado. Por otro lado, la página debe de tener un tiempo de cargar por lo menos de dos segundos.

## Plan de Pruebas

“Un plan de pruebas es un documento que describe los pasos que se deben de llevar a cabo y el enfoque de las pruebas en un proyecto de desarrollo de software” (Plan de pruebas de software: 8 pasos para realizarlas - Icaria Technology, s. f.). El objetivo del plan es garantizar el rendimiento y la calidad del software antes del lanzamiento. El plan se ocupa de:

- Establecer los objetivos y metas específicas.
- Proporciona la estructura para las actividades de testing.
- Determina los criterios de aceptación de cada prueba.
- Facilita la evaluación objetiva en vistas a determinar si el software es apto para su lanzamiento.

### *¿Cómo se Realiza un Plan de Pruebas?*

- Comprendiendo los objetivos del plan de pruebas.
- Determinando qué nuevas funcionalidades se desea testear.
- Definiendo la estrategia de pruebas.
- Estableciendo los entornos requeridos.
- Definiendo los criterios.
- Eligiendo la metodología.
- Estimando los requisitos de personal.
- Planificando.

### *Tipos de Pruebas de Software*

- **Prueba de aceptación:** verifica que todo funcione según lo previsto.
- **Pruebas de integración:** se asegura de que las funciones del software o componentes operen juntas.

- **Pruebas de unidad:** esto valida que cada dispositivo de software funcione según lo esperado.
- **Pruebas funcionales:** verifica las funciones por medio de emulaciones de escenarios de negocio.
- **Pruebas de rendimiento:** prueba cómo funciona el software bajo diferentes cargas de trabajo.
- **Pruebas de regresión:** por medio de esto se puede verificar si las nuevas características rompen o degradan la funcionalidad.
- **Pruebas de estrés:** prueba cuánta tensión puede soportar el sistema antes de que falle.
- **Pruebas de usabilidad:** mediante esto se valida qué tan bien un cliente puede usar un sistema o una aplicación web para completar una tarea.

### **Casos de uso**

“Los casos de uso se crean para refinar un conjunto de requisitos basados en un rol o tarea” (Definición de casos de uso - Documentación de IBM, s. f.-b). En vez de la lista tradicional de requisitos que pueden no abordar directamente el uso de la solución, los casos de uso agrupan requisitos comunes basados en el tipo de rol u objetivo. Definen lo que los usuarios o roles están haciendo en la solución, un proceso de negocio establece cómo realizan estas funciones.

Figura 42

*Ejemplo Casos de Uso Registrar Cliente*

<b>1.- Proceso de Negocio</b>	Registrar Cliente	
<b>2.- Objetivo</b>	Brindar al cliente un servicio adicional de acumulación de puntos u otras promociones, así también tener información que nos ayude a tomar decisiones en el mercado de la empresa.	
<b>3.- Actores</b>	Empleado(Principal), Sistema Farmacia (Secundario)	
<b>4.- Pre-condiciones</b>		
1. Datos del Cliente (cliente debe ser mayor de edad).		
<b>5.- Interacción</b>		
	<b>Empleado</b>	<b>Sistema Farmacia</b>
	1. Ingresar DNI del cliente	2. Verifica DNI (validez).
	3. Ingresar datos del cliente	4. Registra datos del cliente 5. Reporte de datos
	6. El cliente verifica la información 7. Finaliza registro cliente	
<b>6.-Extensiones</b>		
	<b>Empleado</b>	<b>Sistema Farmacia</b>
	5. Reporte de registro 5.a.- Consultar datos del cliente	. Muestra datos del cliente para ser confirmados
<b>7.- Post-condiciones</b>		
1. El cliente está registrado en la Base de Datos. 2. Sus puntos bonus se inicializan en cero.		
<b>8.- Trigger</b>		
Que el cliente requiera registrarse (así comienza el caso de uso).		
<b>9.- Excepciones</b>		
2. No se verificó validez del DNI <b>Mensaje:</b> "Ingrese DNI nuevamente, este no existe"		
<b>10.- Versión</b>	1.0	
<b>11.- Tiempo de Ejecución</b>	1 minuto	
<b>12. Diagrama de Actividades del Proceso de Negocio</b>		

Fuente: (Casos Uso Farmacia | PDF | Farmacia | Caso de uso, s. f.)

*Aplicaciones de Casos de Uso*

- Diseñar interacciones claras
- Probar y evaluar prototipos
- Comunicación efectiva
- Documentación del sistema
- Evaluación de riesgos
- Optimización continua

### *Componentes del Diagrama de Caso de uso*

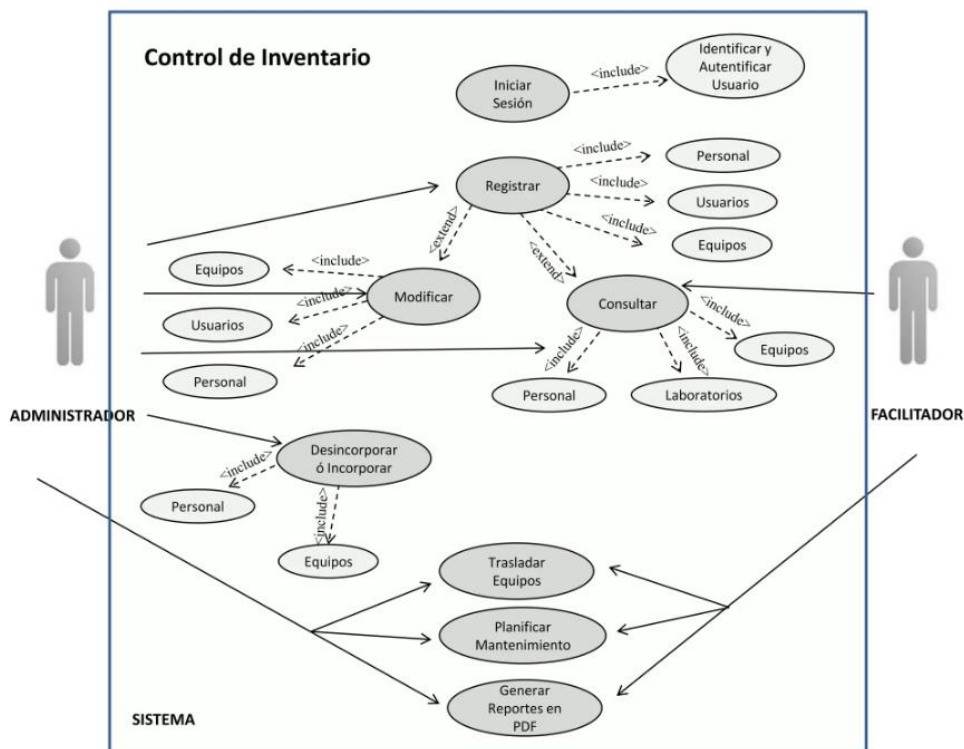
- **Actores:** son los usuarios.
- **Caso de uso:** representan las acciones.
- **Relaciones:** muestran las relaciones entre caso de uso y actores.
- **Sistema:** agrupan diferentes elementos.

### *¿Cómo hacer un diagrama de caso de uso?*

- Recopilar fuentes de información
- Identificar actores potenciales
- Identificar posibles casos de uso
- Conectar los casos de uso
- Describir actores
- Buscar más casos de uso
- Documentar casos de uso
- Relacionar modelos entre casos de uso empresarial
- Verificar la vista

Figura 43

Diagrama de Casos de Usos



Fuente: (Diagrama de casos de usos | PPT, s. f.)

### Instrumentos

Se utilizará en el actual caso, una mezcla entre entrevistas a los usuarios implicados, así como observaciones y cuestionarios. Todo con la finalidad de obtener la información lo más detalladamente posible.

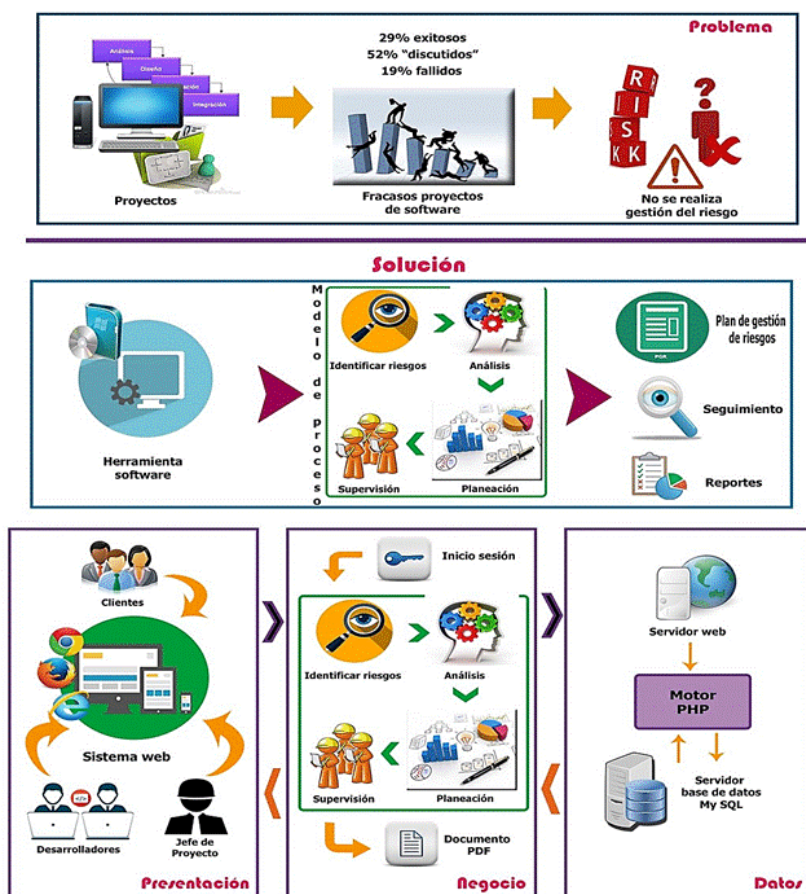
## **Capítulo IV. Análisis de Resultados**

## Plan de Gestión de Riesgos

Un plan de gestión de riesgos es un documento que describe cómo se identificarán, gestionarán, evaluarán y supervisarán estos sucesos. Este documento puede referirse a un proyecto específico, a un período específico o incluso, una actividad o sector específicos. Además, un plan de riesgos es esencial para garantizar que los posibles problemas se aborden sistemática y proactivamente, de esta forma, se pueden minimizar los impactos negativos y también, encontrar y aprovechar oportunidades de mejora continua.

**Figura 44**

*Gestión de Riesgos*



Fuentes: (Desarrollo de un software web para la generación de planes de gestión de riesgos de software, s. f.)

### ***Componentes Principales***

**Identificación de riesgos:** tener un método para enumerar e identificar todos los riesgos potenciales que podrían afectar el proyecto. Este puede ser un documento, ya sea en una hoja de cálculo o un software. Lo importante es tener un criterio bien definido y un control constante de todo esto.

**Análisis del riesgo:** para evaluar las probabilidades de ocurrencia y el impacto de cada riesgo que se identificó en el paso anterior. Ese análisis puede ser cualitativo o cuantitativo.

**Plan de Respuesta al Riesgo:** para incluir estrategias y acciones específicas que tengan como objetivo abordar o mitigar los riesgos y así analizarlos con anticipación.

#### **Categorías:**

***Evitar:*** se debe de cambiar el plan de riesgos para eliminarlos.

***Reducir:*** se implementan acciones para reducir la probabilidad de que ocurran.

***Transferencia:*** tiene como objetivo transferir el riesgo a otra parte.

***Aceptar:*** este plan de respuesta reconoce los posibles riesgos y prepara a la empresa para enfrentarlos.

**Seguimiento y revisión:** sigue los riesgos y evalúa la eficiencia de las estrategias para responder. Aquí también se puede modificar el plan de riesgos de ser necesario.

**Documentación y Comunicación:** por último, se tiene que registrar toda la información relacionada con los riesgos y estrategias. Además, se debe de contar con medios de comunicación ágiles y fáciles para todos los que necesiten utilizarlos.

## *Tareas de la gestión de riesgos*

### **Identificar:**

- Considerar las amenazas potenciales, incluida la gestión avanzada del riesgo de los datos en la ingeniería de software y la gestión global del riesgo.
- Definir un riesgo en relación con la gestión de proyectos de software y evaluar su importancia.

### **Clasificar y priorizar:**

- El clasificar los riesgos, aplicando la gestión de riesgos en la gestión de proyectos de software.
- Priorizar los riesgos identificados del proyecto de software para centrarse en los de mayor impacto.

### **Desarrollar un plan de acción:**

- Crear una plantilla de acción para la mitigación los riesgos y aprovechando el software de mitigación.
- Al abordar las aplicaciones específicas de la gestión de riesgos en la gestión de proyectos de software, se podrán adaptar las estrategias consecuentemente.

### **Supervisar continuamente:**

- Plantar la supervisión continua en la gestión de riesgos del ciclo de vida del software
- Promover la comunicación constante dentro del equipo para identificar y abordar nuevas amenazas.

### **Aplicar planes de acción:**

- Actuar conforme al plan de acción si se materializa alguna amenaza identificada.

- Tramitar los riesgos de desarrollo de productos proactivamente mediante una evaluación eficaz de los riesgos del software.

### ***Matriz de riesgos***

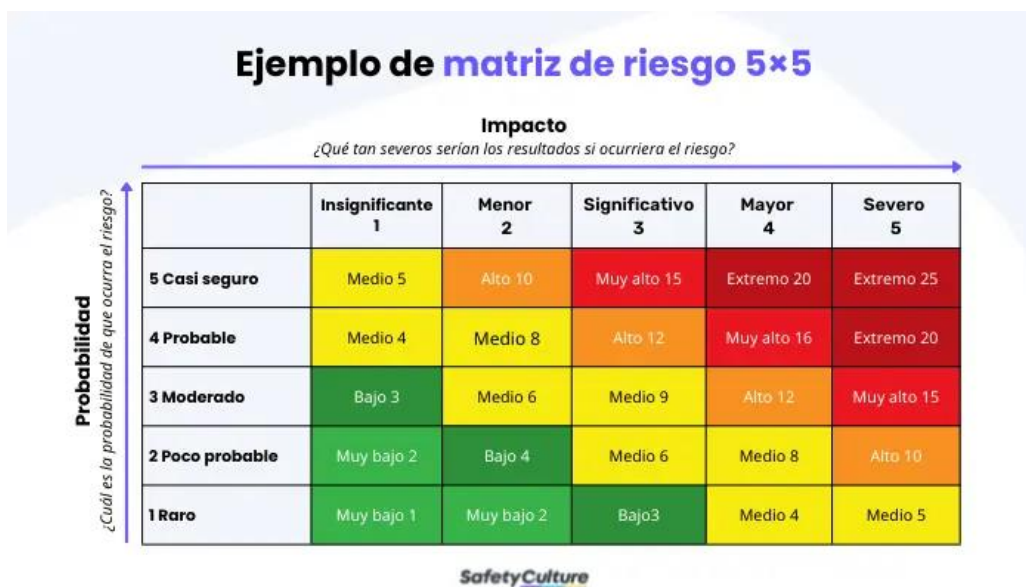
Como describe Matriz de Riesgos: cómo evaluar los riesgos para lograr el éxito del proyecto [2024] (Asana, s. f.): “Una matriz de riesgos es una herramienta que análisis los riesgos y se utiliza para evaluar la probabilidad y la gravedad del riesgo durante el proceso de planificación del proyecto”. Una vez que se haya evaluado la probabilidad y la gravedad de cada riesgo, se pueden ubicar en la matriz para calcular la calificación del impacto de cada riesgo. Estas calificaciones ayudarán al equipo a establecer qué prioridad asignar a los riesgos del proyecto y a gestionarlos efectivamente.

### ***Tipos de riesgos:***

- **Riesgo estratégico:** implica los errores de rendimiento o de decisión.
- **Riesgo operativo:** es el error de proceso o procedimiento.
- **Riesgo financiero:** involucra varios eventos que causen una pérdida de ganancias para la empresa.
- **Riesgo técnico:** incluye cualquier aspecto relacionado con la tecnología de la empresa.
- **Riesgo externo:** es el que está fuera del control humano.

Figura 45

## Matriz de Riesgos



Fuente: (Matriz de riesgo: Importancia y ejemplos | SafetyCulture, s. f.)

### Estudio de Factibilidad

Como lo indica el artículo de Estudio de factibilidad - Qué es y qué tipos hay (s. f.): “Un estudio de factibilidad es el que hace una empresa para determinar la posibilidad de poder desarrollar un proyecto o un negocio”. Sus puntos claves son:

- Ayuda a las empresas a tomar mejores decisiones.
- Revela posibles desafíos que el proyecto podría tener, permitiendo a la empresa prepararse adecuadamente.
- Facilitan una base para desarrollar estrategias efectivas que aumenten la probabilidad de éxito del proyecto.

## Figura 46

### *Estudio de Factibilidad*



Fuente: (Estudio de factibilidad - Qué es y qué tipos hay, s. f.)

### *Análisis de factibilidad*

Se evaluarán a través de este procedimiento, las metas del proyecto actual para asegurar que su ejecución sea factible para la microempresa, demostrando que se cuenta con el personal adecuado y la infraestructura necesaria para su implementación.

Hay que destacar que la viabilidad es la medida de posibilidad de completar una acción, al evaluar todos los elementos del proyecto. Tal como se ha mencionado anteriormente en este estudio, las ventajas que se podrían lograr al realizar la iniciativa son:

- Organización y gestión más eficiente de los datos
- Control del inventario
- La optimización de procesos reducirá el tiempo operativo y aumenta la eficiencia.
- Control de acceso por roles (solo se utilizará el administrador).
- Seguridad en el manejo de información

### ***Recursos tecnológicos***

Este aspecto evalúa si la infraestructura técnica que posee la empresa puede responder eficiente y favorablemente para desarrollar un proyecto o negocio que se tiene panificado.

Además, se debe verificar si las personas poseen los conocimientos técnicos necesarios para poder utilizar el equipo y el software necesario. Con respecto al hardware necesario para este proyecto, el servidor o terminal por utilizar debe cumplir con los requisitos mínimos son los siguientes:

**Tabla 1**

#### *Recursos del Equipo*

<b>Características</b>	<b>Especificaciones</b>
<b>Sistema Operativo</b>	Windows 10/ 11
	Windows server
<b>Procesador</b>	Intel Core i3 en adelante
	Ryzen 3 en adelante
<b>Almacenamiento</b>	219 Gb en adelante
<b>Memoria RAM</b>	8 Gb en adelante
<b>Marca</b>	Cualquiera

Al valorar el hardware y tener en cuenta la configuración mínima requerida, la microempresa no tiene que invertir dinero, debido a que cuenta con los equipos necesarios para el correcto funcionamiento de la página que se va a desarrollar.

Con respecto al software, se dispone de todas las aplicaciones necesarias para el desarrollo del proyecto, como MySQL y Visual Studio Code. El sistema operativo de los dispositivos es Windows 10, lo cual indica que no se requiere gastar más dinero en la compra de otros programas.

### ***Optimización de recursos***

Se debe realizar un análisis de la relación costo-beneficio del negocio o proyecto y calcular ambos aspectos. Si en la valoración se observa que los costos superan a los beneficios sería mejor no desarrollarlo. En cambio, si el beneficio supera los costos, la decisión de la realización del proyecto se vuelve menos arriesgada, aunque no implica que no existan riesgos.

Debido a que el sistema está siendo desarrollado por un pariente de la dueña de la microempresa, no implicará un gasto adicional. Y dado que el sistema se está desarrollando como parte de una tesis académica, lo que significa que no generará gastos adicionales. No se cobrará por el servicio prestado, dado que se trata de un proyecto educativo. Por lo tanto, los costos asociados al desarrollo no se reflejarán en la contabilidad de la microempresa. De acuerdo con la evaluación realizada, se detallan los gastos financieros necesarios para llevar a cabo el proyecto de software.

**Tabla 2**

#### *Costo de Proyecto*

<b>Concepto</b>	<b>Descripción</b>	<b>Costo</b>
<b>Desarrollo</b>	Por un pariente de la dueña de la microempresa	Sin costo adicional
<b>Proyecto</b>	Parte de una tesis académica	No se cobrará por el servicio
<b>Contabilidad</b>	No generará gastos en la microempresa	No reflejados

**Tabla 3**

#### *Costos del Proyecto*

<b>Concepto</b>	<b>Descripción</b>	<b>Costo</b>
<b>Desarrollo</b>	Por un estudiante de ingeniería informática	\$1.600
<b>Proyecto</b>	Parte de una tesis académica	\$0
<b>Contabilidad</b>	No generará gastos en la microempresa	\$1.600

### ***Gestión del proyecto***

Está relacionada con el personal que tiene que realizar el proyecto. Sobre todo, depende de los recursos humanos que forman parte de la organización. En consecuencia, al momento de desarrollar un proyecto o negocio se tiene que implementar un sistema.

En resumen, se encarga de determinar si un proyecto de desarrollo de software es sostenible en términos operativos, teniendo en cuenta todos los factores que puedan afectar su implementación y éxito a largo plazo. Por lo tanto, se debe considerar lo siguiente:

- Que no sea muy complicado para el usuario que lo vaya a operar.
- Observar si el usuario se resiste a usarlo por miedo o por cualquier otra razón.
- Reconocer si el usuario se puede adaptar a los cambios, sobre todo si el cambio se produce muy rápidamente.
- Se debe valorar si no se puede volver obsoleto de forma muy rápida y buscar planes de contingencia.

### **Descripción del Proyecto**

- Desarrollo de un sistema de gestión y control de inventario, utilizando Visual Studio Code, Spring Boot, Bootstrap, HTML, CSS, JS y Thymeleaf.
- Funcionalidades: autenticación de usuarios, gestión de productos, categorías, proveedores, inventarios, clientes, pedidos, detalles pedidos, suministros y sesiones.
- Parte de una tesis académica sin costos adicionales para la microempresa.

### **Objetivos del Proyecto**

- Facilitar la gestión de inventarios para la microempresa
- Mejorar la trazabilidad de las operaciones

- Proporcionar un sistema seguro y fácil de usar

### **Recursos Necesarios**

#### *Hardware*

- Servidor web para alojar el sistema
- Computadoras para los usuarios

#### *Software*

- Java
- Spring Boot
- Thymeleaf
- Base de datos MySQL

### **Recursos Humanos**

- Un estudiante de Ingeniería informática con conocimiento en Java y Spring Boot.

### **Evaluación de Riesgos**

- Dependencia del desarrollador: riesgo de falta de disponibilidad del único desarrollador.
- Tiempo de implementación: ajustar el desarrollo e implementación al cronograma académico.
- Problemas de programación: problemas de versiones.

### **Evaluación de la Viabilidad Operativa**

- La microempresa no incurre en costos adicionales significativos.
- La experiencia del estudiante es suficiente para cumplir con los objetivos del proyecto.

Figura 47

Tipos de Factibilidad



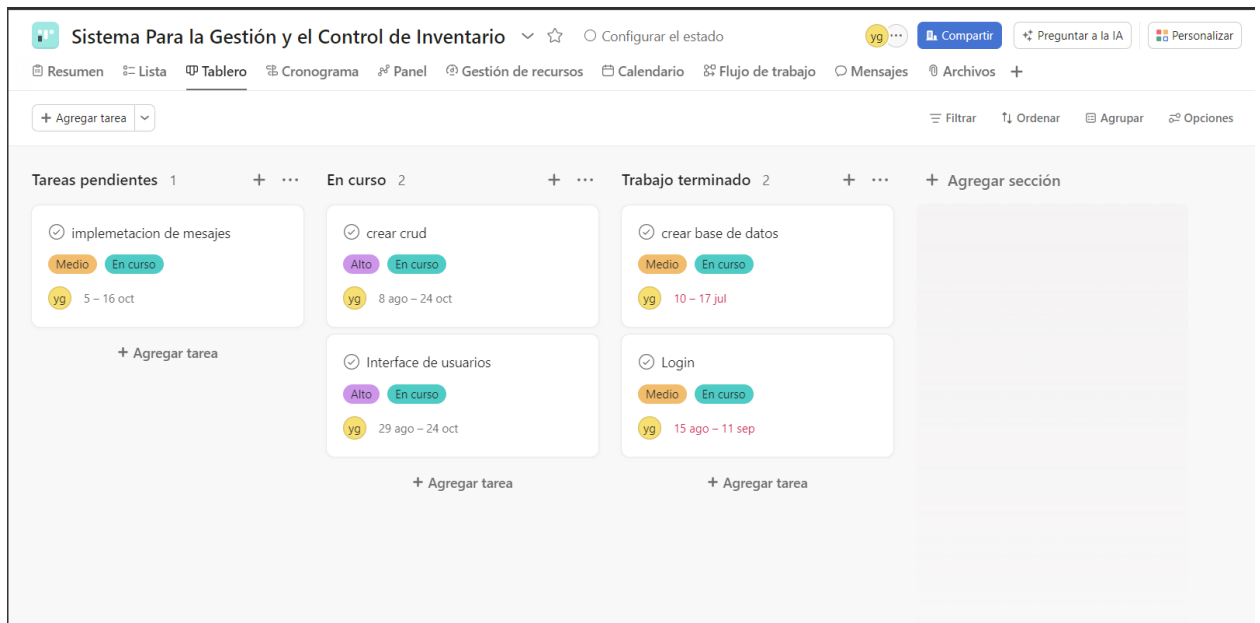
Fuente: (Estudio de factibilidad - Qué es y qué tipos hay, s. f.)

### Metodología Utilizada

La metodología que se utilizó para el desarrollo del sistema de control y gestión de inventarios para la microempresa Productos de Concreto Daniel fue Kanban, dado que esta metodología se enfoca en la visualización del trabajo, la gestión del flujo y la mejora continua. Al usar Kanban, se logró:

- **Observar el progreso:** se implementó un tablero de Kanban para poder visualizar el estado actual de las tareas en tiempo real, lo que simplifica su gestión y orden.
- **Adaptabilidad:** debido a que es posible modificar el ritmo de trabajo según las demandas del proyecto y las actividades en desarrollo.

- **Enfoque en eficacia:** se reduce la cantidad de trabajo en curso para prevenir la saturación y garantizar que cada tarea se finalice de forma eficiente.
- **Mejora continua:** mediante revisiones periódicas del tablero, se detectan bloqueos y se implementan decisiones para perfeccionar el proceso de desarrollo.

**Figura 48***Sistema Para la Gestión y el Control de Inventario Kanban*

**Figura 49***Tarea Terminadas Sistema Para la Gestión y el Control de Inventario Kanban*

The image shows a Kanban board interface for a project titled "Sistema Para la Gestión y el Control de Inventario". The board is organized into four columns: "Tareas pendientes" (0 items), "En curso" (0 items), "Trabajo terminado" (5 items), and "Agregar sección". The "Trabajo terminado" column contains five task cards, each with a status label, a priority label, a user icon, and a date range. The tasks are: "5 - 16 oct" (Medio, En curso), "crear base de datos" (Medio, En curso), "Login" (Medio, En curso), "10 - 17 jul" (Medio, En curso), and "15 ago - 11 sep" (Medio, En curso). The bottom-most card is "8 ago - 24 oct" (Alto, En curso). The interface includes a top navigation bar with various menu items and a bottom toolbar with filtering and sorting options.

**Sistema Para la Gestión y el Control de Inventario** Configurar el estado

Resumen Lista **Tablero** Cronograma Panel Gestión de recursos Calendario Flujo de trabajo Mensajes Archivos +

+ Agregar tarea Filtrar Ordenar Agrupar Opciones

Tareas pendientes 0 + ... En curso 0 + ... Trabajo terminado 5 + ... + Agregar sección

- Medio En curso yg 5 - 16 oct
- ✓ crear base de datos Medio En curso yg 10 - 17 jul
- ✓ Login Medio En curso yg 15 ago - 11 sep
- ✓ crear crud Alto En curso yg 8 ago - 24 oct

## **Capítulo V. Conclusiones y Recomendaciones**

## *Conclusiones*

- Se implementó un sistema funcional que satisface las necesidades de gestión y control del inventario de la microempresa. Este sistema optimiza el manejo de datos relacionados con productos y materiales, mejorando los procesos internos y reduciendo los errores manuales.
- Se llevó a cabo un análisis exhaustivo, logrando recopilar los requisitos funcionales y no funcionales necesarios para el diseño y desarrollo del sistema, alineándose con las necesidades de la microempresa.
- El diseño del sistema se desarrolló mediante casos de uso y diagramas UML, lo que permitió estructurar las funcionalidades de manera clara, facilitando la implementación y asegurando que todas las necesidades identificadas fueran cubiertas.
- El sistema fue programado utilizando herramientas de código abierto, lo que permitió reducir costos y aprovechar recursos tecnológicos disponibles. Este enfoque garantizó un desarrollo eficiente y adaptable a las necesidades de la microempresa.
- Las pruebas realizadas demostraron que el sistema cumple con los objetivos establecidos y que es funcional para las operaciones diarias de la microempresa, asegurando la confiabilidad y estabilidad.

### ***Recomendaciones***

- Realizar evaluaciones periódicas del sistema para identificar áreas de mejora, asegurando que siga cumpliendo con las necesidades de la microempresa a medida que esta crezca o diversifique sus operaciones.
- Actualizar periódicamente el análisis de requisitos para garantizar que el sistema se mantenga alineado con los procesos y necesidades cambiantes de la microempresa.
- Mantener los diagramas UML como parte de la documentación técnica del sistema y actualizarlos conforme se realicen cambios o mejoras en el sistema.
- Continuar utilizando herramientas de código abierto para futuros desarrollos o mejoras, aprovechando su flexibilidad, escalabilidad y bajo costo.
- Establecer un plan de pruebas continuo para garantizar que el sistema funcione correctamente tras futuras actualizaciones o modificaciones.

**Capítulo VI. Análisis de Requerimientos**

## **Análisis de Requerimientos**

Se muestran a continuación, los requerimientos que sirven para obtener un correcto funcionamiento del prototipo que se desarrolló en la microempresa Productos de Concreto Daniel

- Requerimientos de usuario
- Requerimientos funcionales
- Requerimientos no funcionales

### ***Requerimientos de usuario***

Los requisitos de usuario son las necesidades, expectativas y deseos que los usuarios tienen respecto a un producto. Estos requisitos pueden abarcan desde la funcionalidad, usabilidad y experiencia en general y son esenciales para diseñar productos que satisfagan las necesidades y expectativas de los usuarios de manera efectiva (Estrategias para gestionar y priorizar requisitos de usuario en proyectos UX, s. f.).

Los requisitos de usuario también pueden incluir preferencias específicas, limitaciones tecnológicas o contextuales, así como expectativas emocionales y psicológicas. Comprender estos requisitos es esencial para diseñar productos que sean intuitivos, eficientes y atractivos para su público (Estrategias para gestionar y priorizar requisitos de usuario en proyectos UX, s. f.). A continuación, se presentan las tablas con los requisitos del usuario de acuerdo con las solicitudes, los cuales son importantes para el correcto desarrollo del sistema.

**Tabla 4***Requerimiento de usuario RUM01*

<b>Responsable:</b> Sofia Gamboa Navarro			
<b>Id:</b>	RUM01	<b>Prioridad:</b>	Alta
<b>Descripción:</b>			
Diseñar un sistema para la gestión y el control de inventario, para una microempresa.			
<b>Fuente:</b>			
Dueña de la microempresa			
<b>Dependencias:</b>			
Ninguna			

Tabla 4. Requerimiento generado por la estudiante para la microempresa Productos de Concreto Daniel.

**Tabla 5***Requerimiento de usuario RUM02*

<b>Responsable:</b> Sofia Gamboa Navarro			
<b>Id:</b>	RUM02	<b>Prioridad:</b>	Alta
<b>Descripción:</b>			
Desarrollar un módulo para el login que reconozca roles de usuario.			
<b>Fuente:</b>			
Dueña de la microempresa			
<b>Dependencias:</b>			
Ninguna			

Tabla 5. Requerimiento generado por la estudiante para la microempresa Productos de Concreto Daniel para el ingreso al sistema.

**Tabla 6***Requerimiento de usuario RUM03*

<b>Responsable:</b> Sofia Gamboa Navarro			
<b>Id:</b>	RUM03	<b>Prioridad:</b>	Alta
<b>Descripción:</b>			
Desarrollar un módulo para registro de usuario.			
<b>Fuente:</b>			
Dueña de la microempresa			
<b>Dependencias:</b>			
Ninguna			

Tabla 6. Requerimiento generado por la estudiante para la microempresa Productos de Concreto Daniel para el registro de usuarios

**Tabla 7***Requerimiento de usuario RUM04*

<b>Responsable:</b> Sofia Gamboa Navarro			
<b>Id:</b>	RUM04	<b>Prioridad:</b>	Alta
<b>Descripción:</b>			
Desarrollar un módulo para registro de productos.			
<b>Fuente:</b>			
Dueña de la microempresa			
<b>Dependencias:</b>			
Ninguna			

Tabla 7. Requerimiento generado por la estudiante para la microempresa Productos de Concreto Daniel para el registro de productos.

**Tabla 8***Requerimiento de usuario RUM05*

<b>Responsable:</b> Sofia Gamboa Navarro			
<b>Id:</b>	RUM05	<b>Prioridad:</b>	Alta
<b>Descripción:</b>			
Desarrollar un módulo para registro de proveedores.			
<b>Fuente:</b>			
Dueña de la microempresa			
<b>Dependencias:</b>			
Ninguna			

Tabla 8. Requerimiento generado por la estudiante para la microempresa Productos de Concreto Daniel para el registro de proveedores.

**Tabla 9***Requerimiento de usuario RUM06*

<b>Responsable:</b> Sofia Gamboa Navarro			
<b>Id:</b>	RUM06	<b>Prioridad:</b>	Alta
<b>Descripción:</b>			
Desarrollar un módulo para registro de clientes.			
<b>Fuente:</b>			
Dueña de la microempresa			
<b>Dependencias:</b>			
Ninguna			

Tabla 9. Requerimiento generado por la estudiante para la microempresa Productos de Concreto Daniel para el registro de clientes.

**Tabla 10***Requerimiento de usuario RUM07*

<b>Responsable: Sofia Gamboa Navarro</b>			
<b>Id:</b>	RUM07	<b>Prioridad:</b>	Alta
<b>Descripción:</b>			
Desarrollar un módulo para registro de categorías.			
<b>Fuente:</b>			
Dueña de la microempresa			
<b>Dependencias:</b>			
Ninguna			

Tabla 10. requerimiento generado por la estudiante para la microempresa Productos de Concreto Daniel para el registro de categorías.

**Tabla 11***Requerimiento de usuario RUM08*

<b>Responsable: Sofia Gamboa Navarro</b>			
<b>Id:</b>	RUM08	<b>Prioridad:</b>	Alta
<b>Descripción:</b>			
Desarrollar un módulo para registro de pedidos.			
<b>Fuente:</b>			
Dueña de la microempresa			
<b>Dependencias:</b>			
<b>Ninguna</b>			

Tabla 11. Requerimiento generado por la estudiante para la microempresa Productos de Concreto Daniel para el registro de pedidos.

**Tabla 12***Requerimiento de usuario RUM09*

<b>Responsable:</b> Sofia Gamboa Navarro			
<b>Id:</b>	RUM09	<b>Prioridad:</b>	Alta
<b>Descripción:</b>			
Desarrollar un módulo para registro de inventarios.			
<b>Fuente:</b>			
Dueña de la microempresa			
<b>Dependencias:</b>			
Ninguna			

Tabla 12. Requerimiento generado por la estudiante para la microempresa Productos de Concreto Daniel para el registro de inventario.

**Tabla 13***Requerimiento de usuario RUM10*

<b>Responsable:</b> Sofia Gamboa Navarro			
<b>Id:</b>	RUM10	<b>Prioridad:</b>	Alta
<b>Descripción:</b>			
Desarrollar un módulo para registro de detalles pedido.			
<b>Fuente:</b>			
Dueña de la microempresa			
<b>Dependencias:</b>			
Ninguna			

Tabla 13. Requerimiento generado por la estudiante para la microempresa Productos de Concreto Daniel para el registro de detalles de pedidos.

**Tabla 14***Requerimiento de usuario RUM11*

<b>Responsable: Sofia Gamboa Navarro</b>			
<b>Id:</b>	RUM11	<b>Prioridad:</b>	Alta
<b>Descripción:</b>			
Desarrollar un módulo para registro de suministros.			
<b>Fuente:</b>			
Dueña de la microempresa			
<b>Dependencias:</b>			
Ninguna			

Tabla 14. Requerimiento generado por la estudiante para la microempresa Productos de Concreto Daniel para el registro de suministros.

**Tabla 15***Requerimiento de usuario RUM12*

<b>Responsable: Sofia Gamboa Navarro</b>			
<b>Id:</b>	RUM12	<b>Prioridad:</b>	Alta
<b>Descripción:</b>			
Desarrollar un módulo para control de bitácora de sesiones.			
<b>Fuente:</b>			
Dueña de la microempresa			
<b>Dependencias:</b>			
Ninguna			

Tabla 15. Requerimiento generado por la estudiante para la microempresa Productos de Concreto Daniel para el control de las sesiones del usuario.

**Tabla 16***Requerimiento de usuario RUM14*

<b>Responsable:</b> Sofia Gamboa Navarro			
<b>Id:</b>	RUM14	<b>Prioridad:</b>	Alta
<b>Descripción:</b>			
Desarrollar un módulo de Dashboard para mostrar los datos.			
<b>Fuente:</b>			
Dueña de la microempresa			
<b>Dependencias:</b>			
<b>Ninguna</b>			

Tabla 16. Requerimiento generado por la estudiante para la microempresa Productos de Concreto Daniel para controlar y ver la cantidad de datos dentro de cada módulo.

### *Requerimientos Funcionales*

Se especifica a continuación, la lista de requisitos funcionales que el sistema debe cumplir para asegurar un correcto funcionamiento.

**Tabla 17**

#### *Requerimiento funcional RFM01*

<b>Responsable:</b> Sofia Gamboa Navarro			
<b>Id:</b>	RFM01	<b>Prioridad:</b>	Alta
<b>Descripción:</b>			
El sistema permite agregar, actualizar, eliminar y modificar los registros de usuarios, siendo el administrador el responsable de estas acciones.			
<b>Fuente:</b>			
Dueña de la microempresa			
<b>Dependencias:</b>			
Ninguna			

Tabla 17. Requerimiento funcional para la gestión de agregar, actualizar, eliminar y modificar los registros de usuarios, siendo el administrador el responsable de estas acciones.

**Tabla 18***Requerimiento funcional RFM02*

<b>Responsable:</b> Sofia Gamboa Navarro			
<b>Id:</b>	RFM02	<b>Prioridad:</b>	Alta
<b>Descripción:</b>			
El sistema permite agregar, actualizar, eliminar y modificar los roles de usuario, siendo el administrador el responsable de estas acciones.			
<b>Fuente:</b>			
Dueña de la microempresa			
<b>Dependencias:</b>			
Ninguna			

Table 18. Requerimiento funcional para la gestión de agregar, actualizar, eliminar y modificar los roles de usuario, siendo el administrador el responsable de estas acciones.

**Tabla 19***Requerimiento funcional RFM03*

<b>Responsable:</b> Sofia Gamboa Navarro			
<b>Id:</b>	RFM03	<b>Prioridad:</b>	Alta
<b>Descripción:</b>			
El sistema permite agregar, actualizar, eliminar y modificar los registros de productos, siendo el administrador el responsable de estas acciones.			
<b>Fuente:</b>			
Dueña de la microempresa			
<b>Dependencias:</b>			
Ninguna			

Tabla 19. Requerimiento funcional para la gestión de agregar, actualizar, eliminar y modificar los registros de productos, siendo el administrador el responsable de estas acciones.

**Tabla 20***Requerimiento funcional RFM04*

<b>Responsable: Sofia Gamboa Navarro</b>			
<b>Id:</b>	RFM04	<b>Prioridad:</b>	Alta
<b>Descripción:</b>			
El sistema permite agregar, actualizar, eliminar y modificar los registros de proveedores, siendo el administrador el responsable de estas acciones.			
<b>Fuente:</b>			
Dueña de la microempresa			
<b>Dependencias:</b>			
Ninguna			

Tabla 20. Requerimiento funcional para la gestión de agregar, actualizar, eliminar y modificar los registros de proveedores, siendo el administrador el responsable de estas acciones.

**Tabla 21***Requerimiento funcional RFM05*

<b>Responsable:</b> Sofia Gamboa Navarro			
<b>Id:</b>	RFM05	<b>Prioridad:</b>	Alta
<b>Descripción:</b>			
El sistema permite agregar, actualizar, eliminar y modificar los registros de clientes. Esta gestión es responsabilidad del administrador, así como de los usuarios que se han habilitado para realizar estas acciones.			
<b>Fuente:</b>			
Dueña de la microempresa			
<b>Dependencias:</b>			
Ninguna			

Tabla 21. Requerimiento funcional para la gestión de agregar, actualizar, eliminar y modificar los registros de clientes. Esta gestión es responsabilidad del administrador, así como de los usuarios que se han habilitado para realizar estas acciones.

**Tabla 22***Requerimiento funcional RFM06*

<b>Responsable:</b> Sofia Gamboa Navarro			
<b>Id:</b>	RFM06	<b>Prioridad:</b>	Alta
<b>Descripción:</b>			
El sistema permite agregar, actualizar, eliminar y modificar los registros de categorías, siendo el administrador el responsable de estas acciones.			
<b>Fuente:</b>			
Dueña de la microempresa			
<b>Dependencias:</b>			
Ninguna			

Tabla 22. Requerimiento funcional para la gestión de agregar, actualizar, eliminar y modificar los registros de categorías, siendo el administrador el responsable de estas acciones.

**Tabla 23***Requerimiento funcional RFM07*

<b>Responsable:</b> Sofia Gamboa Navarro			
<b>Id:</b>	RFM07	<b>Prioridad:</b>	Alta
<b>Descripción:</b>			
El sistema permite agregar, actualizar, eliminar y modificar los registros de pedidos. Esta gestión es responsabilidad del administrador, así como de los usuarios que se han habilitado para realizar estas acciones.			
<b>Fuente:</b>			
Dueña de la microempresa			
<b>Dependencias:</b>			
Ninguna			

Tabla 23. Requerimiento funcional para la gestión de agregar, actualizar, eliminar y modificar los registros de pedidos. Esta gestión es responsabilidad del administrador, así como de los usuarios que se han habilitado para realizar estas acciones.

**Tabla 24***Requerimiento funcional RFM08*

<b>Responsable:</b> Sofia Gamboa Navarro			
<b>Id:</b>	RFM08	<b>Prioridad:</b>	Alta
<b>Descripción:</b>			
El sistema permite agregar, actualizar, eliminar y modificar los registros de inventarios, siendo el administrador el responsable de estas acciones.			
<b>Fuente:</b>			
Dueña de la microempresa			
<b>Dependencias:</b>			
Ninguna			

Tabla 24. Requerimiento funcional para la gestión de agregar, actualizar, eliminar y modificar los registros de inventarios, siendo el administrador el responsable de estas acciones.

**Tabla 25***Requerimiento funcional RFM09*

<b>Responsable:</b> Sofía Gamboa Navarro			
<b>Id:</b>	RFM09	<b>Prioridad:</b>	Alta
<b>Descripción:</b>			
El sistema permite agregar, actualizar, eliminar y modificar los registros de detalles de pedido. Esta gestión es responsabilidad del administrador, así como de los usuarios que se han habilitado para realizar estas acciones.			
<b>Fuente:</b>			
Dueña de la microempresa			
<b>Dependencias:</b>			
Ninguna			

Tabla 25. Requerimiento funcional para la gestión de agregar, actualizar, eliminar y modificar los registros de detalles de pedido. Esta gestión es responsabilidad del administrador, así como de los usuarios que se han habilitado para realizar estas acciones.

**Tabla 26***Requerimiento funcional RFM10*

<b>Responsable:</b> Sofia Gamboa Navarro			
<b>Id:</b>	RFM10	<b>Prioridad:</b>	Alta
<b>Descripción:</b>			
El sistema permite agregar, actualizar, eliminar y modificar los registros de suministros, siendo el administrador el responsable de estas acciones.			
<b>Fuente:</b>			
Dueña de la microempresa			
<b>Dependencias:</b>			
Ninguna			

Tabla 26. Requerimiento funcional para la gestión de agregar, actualizar, eliminar y modificar los registros de suministros, siendo el administrador el responsable de estas acciones.

**Tabla 27***Requerimiento funcional RFM11*

<b>Responsable:</b> Sofia Gamboa Navarro			
<b>Id:</b>	RFM11	<b>Prioridad:</b>	Alta
<b>Descripción:</b>			
El sistema permite visualizar quién se ha logueado, la hora de ingreso y la hora de salida. La bitácora de sesiones lleva un registro de esta información y el administrador es responsable de su gestión.			
<b>Fuente:</b>			
Dueña de la microempresa			
<b>Dependencias:</b>			
Ninguna			

Tabla 27. Requerimiento funcional para la gestión del sistema permite visualizar quién se ha logueado, la hora de ingreso y la hora de salida. La bitácora de sesiones lleva un registro de esta información, y el administrador es responsable de su gestión.

**Tabla 28***Requerimiento funcional RFM13*

<b>Responsable:</b> Sofia Gamboa Navarro			
<b>Id:</b>	RFM13	<b>Prioridad:</b>	Alta
<b>Descripción:</b>			
El sistema permite visualizar la cantidad de datos almacenados. El dashboard muestra el contenido del sistema, siendo el administrador el responsable de su gestión.			
<b>Fuente:</b>			
Dueña de la microempresa			
<b>Dependencias:</b>			
Ninguna			

Tabla 28. Requerimiento funcional para la gestión del sistema permite visualizar la cantidad de datos almacenados. El dashboard muestra el contenido del sistema, siendo el administrador el responsable de su gestión.

## Requerimientos No Funcionales

**Tabla 29**

### Requerimiento no Funcionales del Sistema

Id	Descripción
RNFM001	Seguridad
RNFM002	Funcionalidad
RNFM003	Accesibilidad

Tabla 29. Muestra los requerimientos no funcionales del sistema.

**Tabla 30**

### Requerimiento no funcional RNFM001

<b>Responsable:</b> Sofia Gamboa Navarro			
<b>Id:</b>	RNFM001	<b>Prioridad:</b>	Alta
<b>Descripción:</b>			
El sistema valida el acceso de los usuarios que han sido registrados previamente por el administrador, quien les ha asignado un rol, un correo y una contraseña correspondiente.			
<b>Fuente:</b>			
Dueña de la microempresa			

Tabla 30. Requerimiento no funcional el sistema valida el acceso de los usuarios que han sido registrados previamente por el administrador, quien les ha asignado un rol, un correo y una contraseña correspondiente.

**Tabla 31***Requerimiento no funcional RNFM002*

<b>Responsable:</b> Sofía Gamboa Navarro			
<b>Id:</b>	RNFM002	<b>Prioridad:</b>	Alta
<b>Descripción:</b>			
El sistema se ha diseñado para que los usuarios puedan utilizarlo y acceder a él de manera sencilla y sin dificultad.			
<b>Fuente:</b>			
Dueña de la microempresa			

Tabla 31. Requerimiento no funcional el sistema se ha diseñado para que los usuarios puedan utilizarlo y acceder a él de manera sencilla y sin dificultad.

**Tabla 32***Requerimiento no funcional RNMF003*

<b>Responsable:</b> Sofía Gamboa Navarro			
<b>Id:</b>	RNMF003	<b>Prioridad:</b>	Alta
<b>Descripción:</b>			
El sistema no tiene un límite de registros, lo que permite acceder a la información necesaria y realizar nuevos registros de forma indefinida.			
<b>Fuente:</b>			
Dueña de la microempresa			

Tabla 32. Requerimiento no funcional el sistema no tiene un límite de registros, lo que permite acceder a la información necesaria y realizar nuevos registros de forma indefinida.

## Modelos UML

### Casos de Uso

Se detallan A continuación, los actores involucrados en el sistema.

**Administrador:** es la personal que se encarga de mantener un registro exacto de los usuarios que van a poder utilizar sistema.

**Usuario:** es la persona que puede agregar, eliminar o modificar información, aunque no tenga los permisos para eliminar usuarios o asignar roles a otros usuarios.

### Caso de Uso 1. Login

#### Figura 50

#### Diagrama de caso de uso - Login

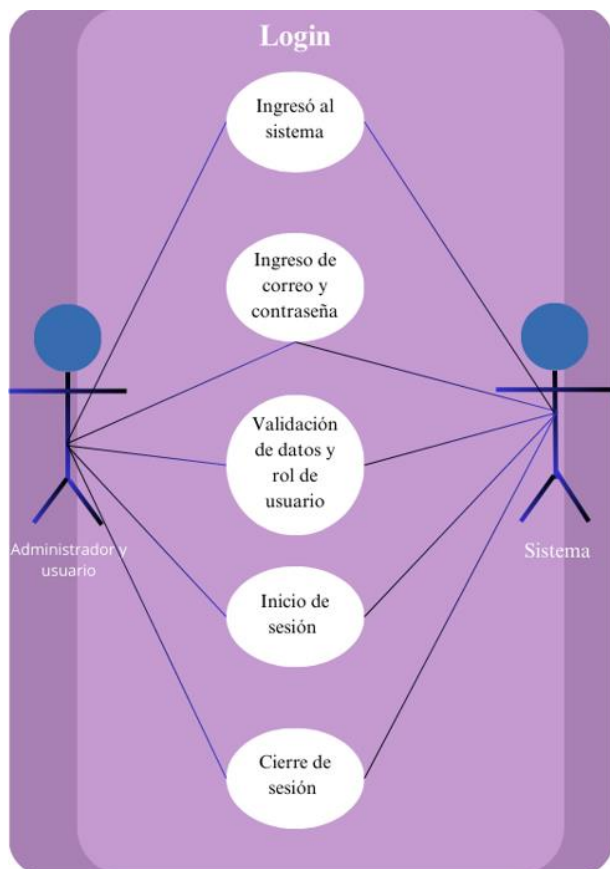


Figura 50. El diagrama de caso de uso pertenece al login del sistema para la gestión y el control de inventario para lo microempresa de Productos de Concreto Daniel.

Tabla 33

*Login*

<b>Casos de uso: Login.</b>	
<b>Autora:</b>	Sofia Gamboa Navarro.
<b>Actores:</b>	<ol style="list-style-type: none"> <li>1. Administrador.</li> <li>2. Usuario.</li> <li>3. Sistema.</li> </ol>
<b>Observaciones:</b>	<ol style="list-style-type: none"> <li>1. Ingresar los datos para iniciar de sesión.</li> <li>2. Cierre de sesión.</li> </ol>
<b>Descripción:</b>	<ol style="list-style-type: none"> <li>1. Acceder al sistema.</li> <li>2. Ingresar correo y contraseña en el módulo de login.</li> <li>3. Darle clic al botón de “Inicio Sesión”.</li> <li>4. Al validar los datos ingresará al sistema y se visualizará el menú principal del sistema según el rol que tenga.</li> <li>5. Al terminar las tareas, para salir de la página debe darle clic al botón de cerrar sesión.</li> </ol>
<b>Observaciones:</b>	Los usuarios que no estén autorizados no tendrán acceso al sistema. En caso de que olvide la contraseña, se deberá de solicitar al administrador que la restablezca.
<b>Mensajes de errores:</b>	<ol style="list-style-type: none"> <li>1. Se mostrará un mensaje de error al ingresar la contraseña incorrecta.</li> <li>2. Se mostrará un mensaje de error al ingresar un usuario erróneo.</li> <li>3. Se mostrará un mensaje de error al ingresar un usuario no registrado.</li> </ol>

## Caso de Uso 2. Registro de Usuario

Figura 51

Diagrama de caso de uso - Registro Usuario

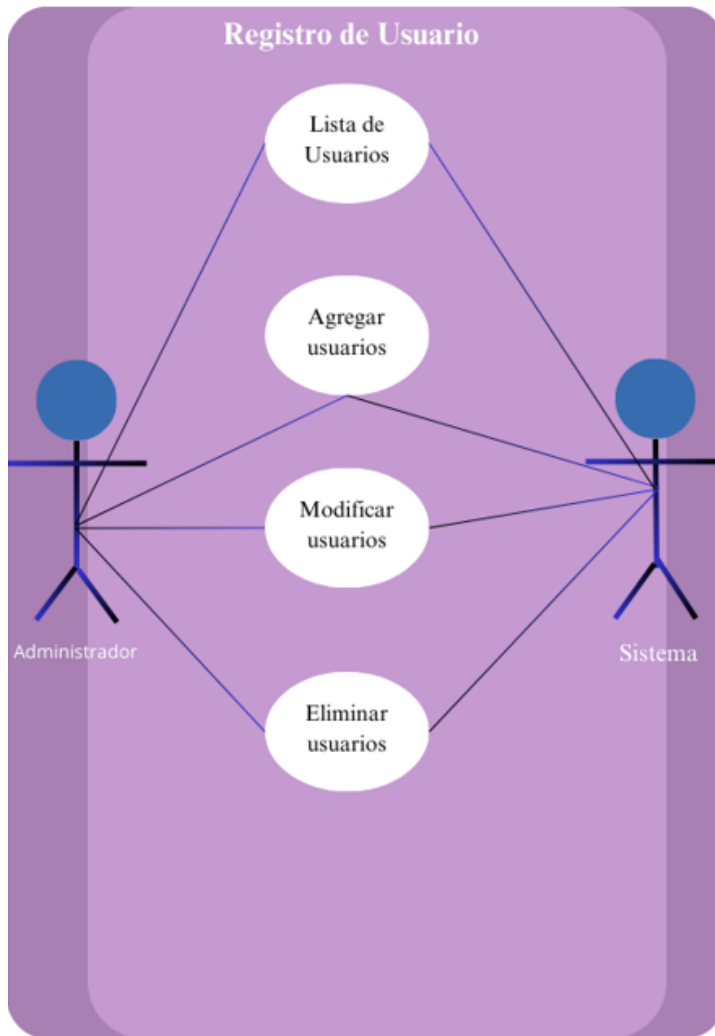


Figura 51. El diagrama de caso de uso pertenece al registro de usuarios del sistema para la gestión y el control de inventario para lo microempresa de Productos de Concreto Daniel.

Tabla 34

*Registro Usuario*

<b>Casos de uso: Registro usuario.</b>	
<b>Autora:</b>	Sofia Gamboa Navarro.
<b>Actores:</b>	<ol style="list-style-type: none"> <li>1. Administrador.</li> <li>2. Sistema.</li> </ol>
<b>Observaciones:</b>	<ol style="list-style-type: none"> <li>1. Lista los datos del usuario.</li> <li>2. Agrega nuevos usuarios con un rol asignado.</li> <li>3. Modifica los datos del usuario.</li> <li>4. Elimina los datos del usuario.</li> </ol>
<b>Descripción:</b>	<ol style="list-style-type: none"> <li>1. Acceder a la página de registro.</li> <li>2. Ingresar los datos requeridos del nuevo usuario.</li> <li>3. Darle clic al botón de guardar.</li> <li>4. El sistema válido que los campos estén completos y no exista un usuario registrado con el mismo correo electrónico.</li> <li>5. El sistema cifra la contraseña y guarda al nuevo usuario en la base de datos.</li> </ol>
<b>Observaciones:</b>	Solo los administradores pueden agregar, modificar o eliminar a un usuario.
<b>Mensajes de errores:</b>	<ol style="list-style-type: none"> <li>1. Si los datos ingresados son incompletos o incorrectos, el sistema muestra un mensaje de error indicando los campos que faltan o están mal.</li> <li>2. El administrador corrige los datos y vuelve a intentar el registro.</li> <li>3. Si el correo electrónico ya está registrado, el sistema muestra un mensaje de error.</li> <li>4. El administrador debe ingresar un nuevo correo electrónico.</li> </ol>

### *Caso de Uso 3. Registro de Productos*

#### **Figura 52**

*Diagrama de caso de uso - Registro Producto*

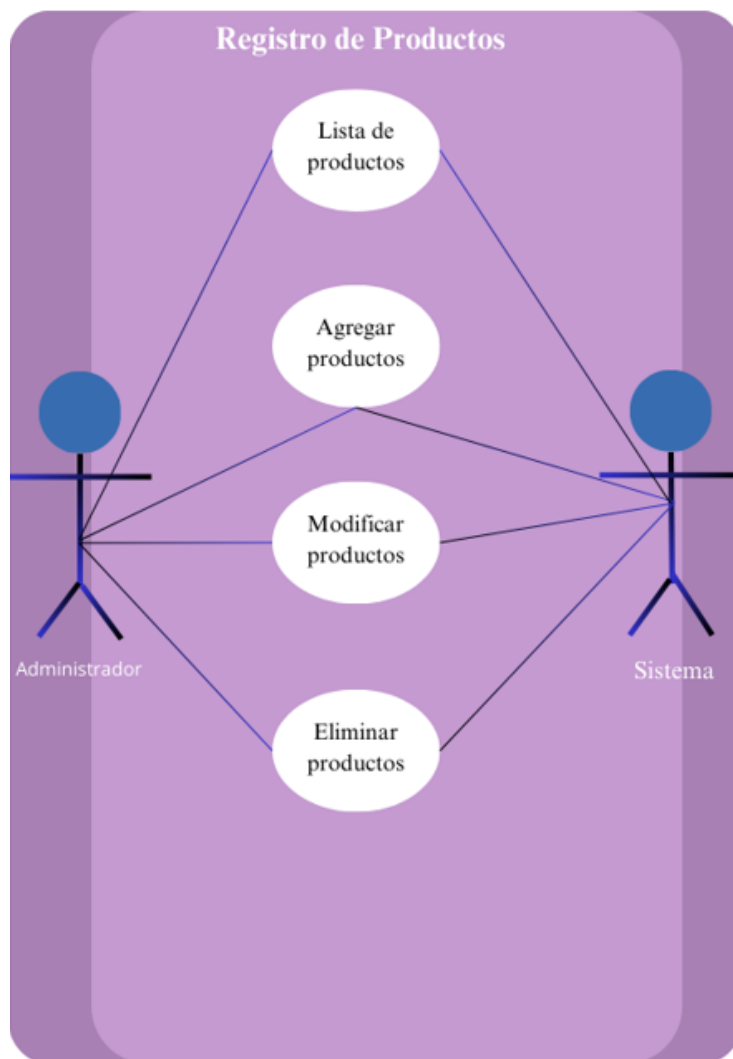


Figura 52. El diagrama de caso de uso pertenece al registro de productos del sistema para la gestión y el control de inventario para lo microempresa de Productos de Concreto Daniel.

Tabla 35

*Registro Productos*

<b>Casos de uso: Registro productos.</b>	
<b>Autora:</b>	Sofia Gamboa Navarro.
<b>Actores:</b>	<ol style="list-style-type: none"> <li>1. Administrador.</li> <li>2. Sistema.</li> </ol>
<b>Observaciones:</b>	<ol style="list-style-type: none"> <li>1. Lista los datos de los productos.</li> <li>2. Agrega nuevos productos.</li> <li>3. Modifica los datos de los productos.</li> <li>4. Elimina los datos de los productos.</li> </ol>
<b>Descripción:</b>	<ol style="list-style-type: none"> <li>1. Acceder a la página de productos.</li> <li>2. Ingresar los datos requeridos del nuevo producto.</li> <li>3. Darle clic al botón de guardar.</li> <li>4. El sistema valida que todos los campos obligatorios estén completos.</li> <li>5. El sistema guarda el nuevo producto en la base de datos.</li> </ol>
<b>Observaciones:</b>	Solo los administradores pueden agregar, modificar o eliminar a un producto.
<b>Mensajes de errores:</b>	<ol style="list-style-type: none"> <li>1. Si el registro es exitoso, el nuevo producto queda registrado en la base de datos y aparece en la lista de productos.</li> </ol>

### *Caso de Uso 4. Registro de Proveedores*

**Figura 53**

*Diagrama de caso de uso - Registro Proveedores*

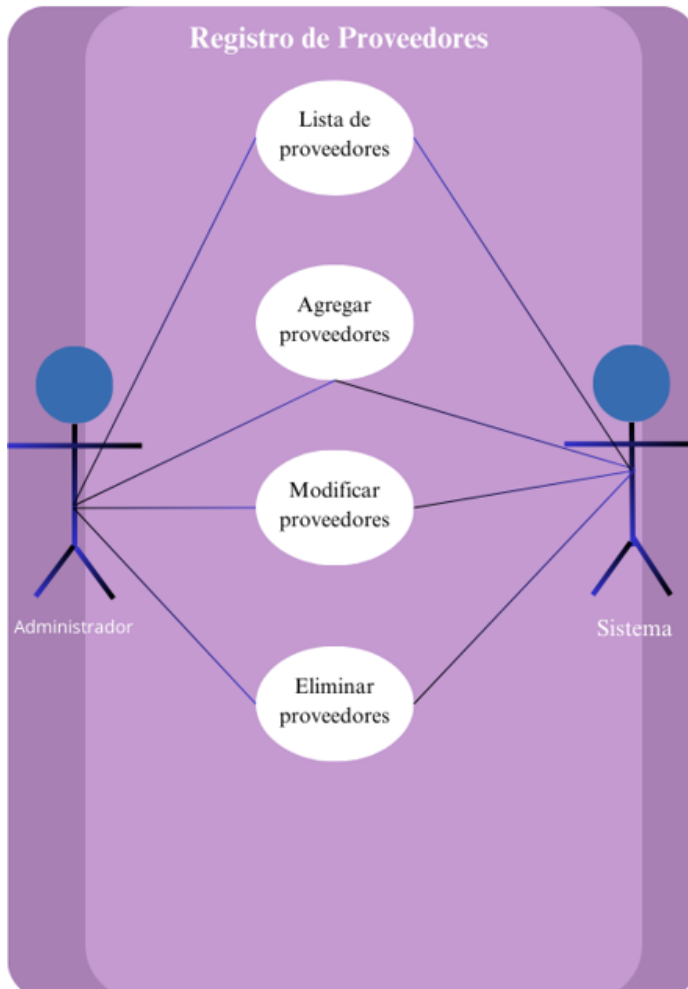


Figura 53. El diagrama de caso de uso pertenece al registro de proveedores del sistema para la gestión y el control de inventario para lo microempresa de Productos de Concreto Daniel.

**Tabla 36***Registro Proveedores*

<b>Casos de uso: Registro proveedores.</b>	
<b>Autora:</b>	Sofia Gamboa Navarro.
<b>Actores:</b>	<ol style="list-style-type: none"> <li>1. Administrador.</li> <li>2. Sistema.</li> </ol>
<b>Observaciones:</b>	<ol style="list-style-type: none"> <li>3. Lista los datos de los proveedores.</li> <li>4. Agrega nuevos proveedores.</li> <li>5. Modifica los datos de los proveedores.</li> <li>6. Elimina los datos de los proveedores.</li> </ol>
<b>Descripción:</b>	<ol style="list-style-type: none"> <li>1. Acceder a la página de proveedores.</li> <li>2. Ingresar los datos requeridos del nuevo proveedor</li> <li>3. Darle clic al botón de guardar.</li> <li>4. El sistema valida que todos los campos obligatorios estén completos.</li> <li>5. El sistema guarda el nuevo proveedor en la base de datos.</li> </ol>
<b>Observaciones:</b>	Solo los administradores pueden agregar, modificar o eliminar a un producto.
<b>Mensajes de errores:</b>	<ol style="list-style-type: none"> <li>1. Si el registro es exitoso, el nuevo proveedor queda registrado en la base de datos y aparece en la lista de proveedor.</li> </ol>

### Caso de Uso 5. Registro de Clientes

Figura 54

Diagrama de caso de uso - Registro Clientes

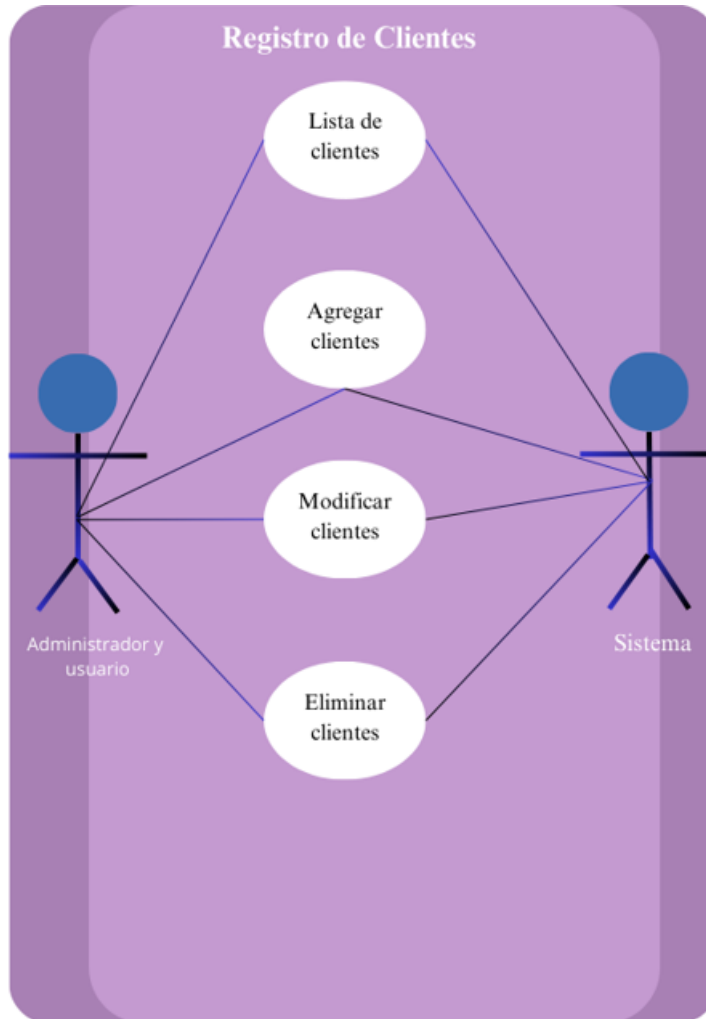


Figura 54. El diagrama de caso de uso pertenece al registro de clientes del sistema para la gestión y el control de inventario para lo microempresa de Productos de Concreto Daniel.

**Tabla 37***Registro Clientes*

<b>Casos de uso: Registro clientes.</b>	
<b>Autora:</b>	Sofia Gamboa Navarro.
<b>Actores:</b>	<ol style="list-style-type: none"> <li>1. Administrador.</li> <li>2. Usuario.</li> <li>3. Sistema.</li> </ol>
<b>Observaciones:</b>	<ol style="list-style-type: none"> <li>1. Lista los datos de los clientes.</li> <li>2. Agrega nuevos clientes.</li> <li>3. Modifica los datos de los clientes.</li> <li>4. Elimina los datos de los clientes.</li> </ol>
<b>Descripción:</b>	<ol style="list-style-type: none"> <li>1. Acceder a la página de clientes.</li> <li>2. Ingresar los datos requeridos del nuevo cliente.</li> <li>3. Darle clic al botón de guardar.</li> <li>4. El sistema valida que todos los campos obligatorios estén completos.</li> <li>5. El sistema guarda el nuevo cliente en la base de datos.</li> </ol>
<b>Observaciones:</b>	Tanto los administradores como los usuarios pueden agregar, modificar o eliminar un cliente.
<b>Mensajes de errores:</b>	<ol style="list-style-type: none"> <li>1. Si el registro es exitoso, el nuevo cliente queda registrado en la base de datos y aparece en la lista de clientes.</li> </ol>

### Caso de Uso 6. Registro de Categorías

Figura 55

Diagrama de caso de uso - Registro Categorías

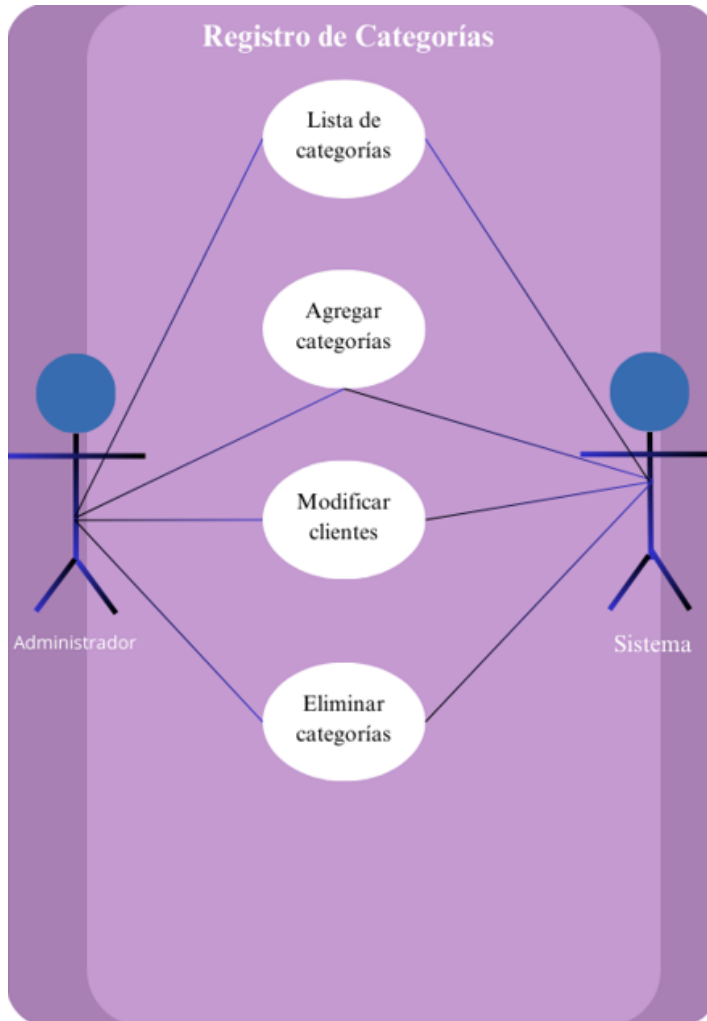


Figura 55. El diagrama de caso de uso pertenece al registro de categorías del sistema para la gestión y el control de inventario para lo microempresa de Productos de Concreto Daniel.

Tabla 38

*Registro Categorías*

<b>Casos de uso: Registro categorías.</b>	
<b>Autora:</b>	Sofía Gamboa Navarro.
<b>Actores:</b>	<ol style="list-style-type: none"> <li>1. Administrador.</li> <li>2. Sistema.</li> </ol>
<b>Observaciones:</b>	<ol style="list-style-type: none"> <li>1. Lista los datos de las categorías.</li> <li>2. Agrega nuevas categorías.</li> <li>3. Modifica los datos de las categorías.</li> <li>4. Elimina los datos de las categorías.</li> </ol>
<b>Descripción:</b>	<ol style="list-style-type: none"> <li>1. Acceder a la página de categorías.</li> <li>2. Ingresar los datos requeridos del nuevo categorías</li> <li>3. Darle clic al botón de guardar.</li> <li>4. El sistema valida que todos los campos obligatorios estén completos.</li> <li>5. El sistema guarda el nuevo categorías en la base de datos.</li> </ol>
<b>Observaciones:</b>	Solo los administradores pueden agregar, modificar o eliminar a unas categorías.
<b>Mensajes de errores:</b>	<ol style="list-style-type: none"> <li>1. Si el registro es exitoso, la nueva categoría queda registrada en la base de datos y aparece en la lista de categorías.</li> </ol>

### *Caso de Uso 7. Registro de Pedidos*

**Figura 56**

*Diagrama de caso de uso - Registro Pedidos*

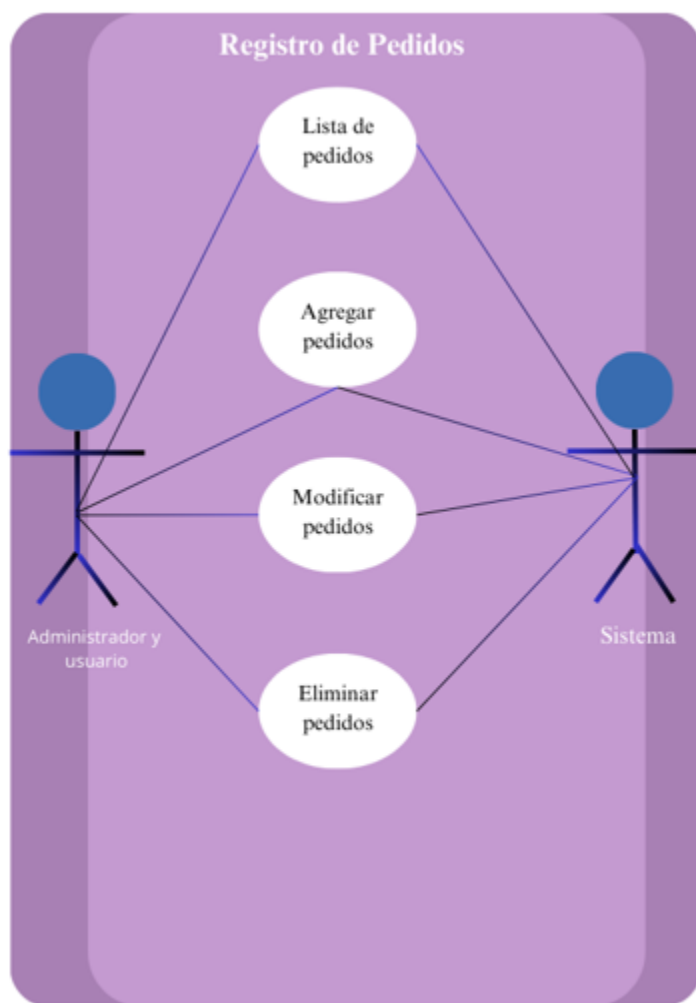


Figura 56. El diagrama de caso de uso pertenece al registro de pedidos del sistema para la gestión y el control de inventario para lo microempresa de Productos de Concreto Daniel.

Tabla 39

*Registro Pedidos*

<b>Casos de uso: Registros pedidos.</b>	
<b>Autora:</b>	Sofia Gamboa Navarro.
<b>Actores:</b>	<ol style="list-style-type: none"> <li>1. Administrador.</li> <li>2. Usuario.</li> <li>3. Sistema.</li> </ol>
<b>Observaciones:</b>	<ol style="list-style-type: none"> <li>1. Lista los datos de los pedidos.</li> <li>2. Agrega nuevos pedidos.</li> <li>3. Modifica los datos de los pedidos.</li> <li>4. Elimina los datos de los pedidos.</li> </ol>
<b>Descripción:</b>	<ol style="list-style-type: none"> <li>1. Acceder a la página de pedidos.</li> <li>2. Ingresar los datos requeridos del nuevo pedido.</li> <li>3. Darle clic al botón de guardar.</li> <li>4. El sistema valida que todos los campos obligatorios estén completos.</li> <li>5. El sistema guarda el nuevo pedido en la base de datos.</li> </ol>
<b>Observaciones:</b>	Tanto los administradores como los usuarios pueden agregar, modificar o eliminar un pedido.
<b>Mensajes de errores:</b>	<ol style="list-style-type: none"> <li>1. Si el registro es exitoso, el nuevo pedido queda registrado en la base de datos y aparece en la lista de pedidos.</li> </ol>

### *Caso de Uso 8. Registro de Inventarios*

**Figura 57**

*Diagrama de caso de uso - Registro Inventarios*

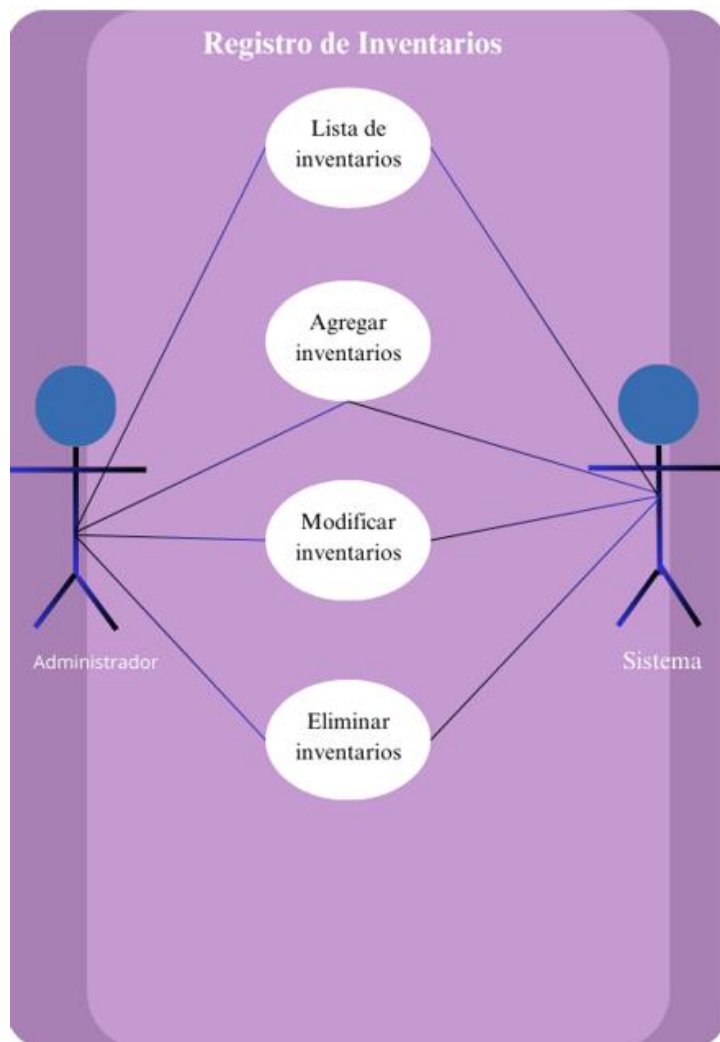


Figura 57. El diagrama de caso de uso pertenece al registro de inventarios del sistema para la gestión y el control de inventario para lo microempresa de Productos de Concreto Daniel.

**Tabla 40***Registro Inventarios*

<b>Casos de uso: Registro inventarios.</b>	
<b>Autora:</b>	Sofia Gamboa Navarro.
<b>Actores:</b>	<ol style="list-style-type: none"> <li>1. Administrador.</li> <li>2. Sistema.</li> </ol>
<b>Observaciones:</b>	<ol style="list-style-type: none"> <li>1. Lista los datos de los inventarios.</li> <li>2. Agrega nuevos inventarios.</li> <li>3. Modifica los datos de los inventarios.</li> <li>4. Elimina los datos de los inventarios.</li> </ol>
<b>Descripción:</b>	<ol style="list-style-type: none"> <li>1. Acceder a la página de inventarios.</li> <li>2. Ingresar los datos requeridos del nuevo inventario</li> <li>3. Darle clic al botón de guardar.</li> <li>4. El sistema valida que todos los campos obligatorios estén completos.</li> <li>5. El sistema guarda el nuevo inventario en la base de datos.</li> </ol>
<b>Observaciones:</b>	Solo los administradores pueden agregar, modificar o eliminar a un producto.
<b>Mensajes de errores:</b>	<ol style="list-style-type: none"> <li>1. Si el registro es exitoso, el nuevo inventario queda registrado en la base de datos y aparece en la lista de inventarios.</li> </ol>

### Caso de Uso 9. Registro de Detalles Pedidos

Figura 58

Diagrama de caso de uso - Registro Detalles Pedidos

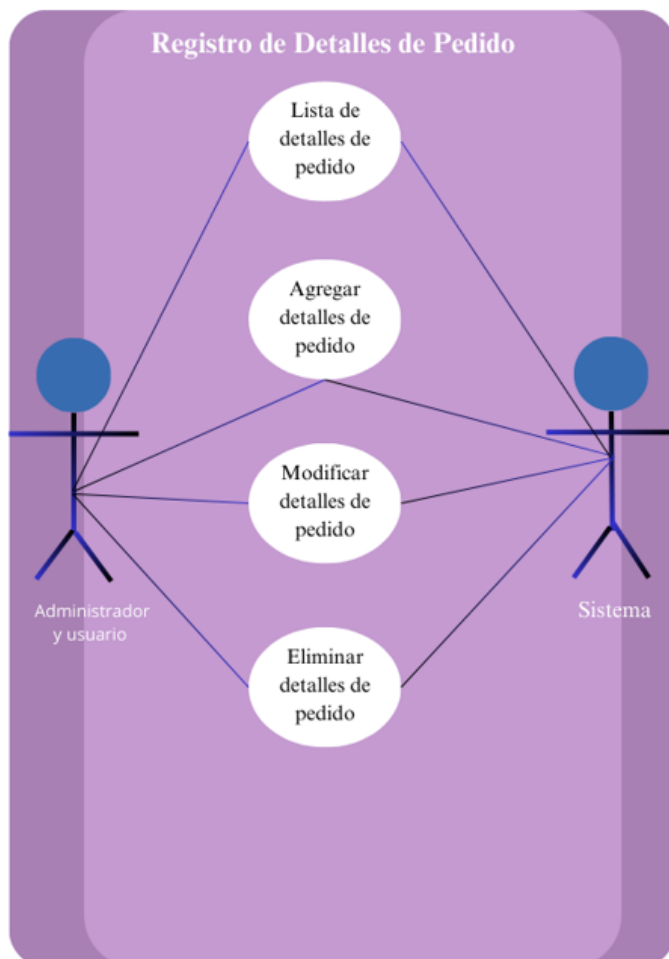


Figura 58. El diagrama de caso de uso pertenece al registro de detalles pedidos del sistema para la gestión y el control de inventario para lo microempresa de Productos de Concreto Daniel.

**Tabla 41***Registro Detalles Pedidos*

<b>Casos de uso: Registros detalles pedidos.</b>	
<b>Autora:</b>	Sofia Gamboa Navarro.
<b>Actores:</b>	<ol style="list-style-type: none"> <li>1. Administrador.</li> <li>2. Usuario.</li> <li>3. Sistema.</li> </ol>
<b>Observaciones:</b>	<ol style="list-style-type: none"> <li>1. Lista los datos de los detalles de pedido.</li> <li>2. Agrega nuevos detalles de pedido.</li> <li>3. Modifica los datos de los detalles de pedido.</li> <li>4. Elimina los datos de los detalles de pedido.</li> </ol>
<b>Descripción:</b>	<ol style="list-style-type: none"> <li>1. Acceder a la página de detalles de pedido.</li> <li>2. Ingresar los datos requeridos del nuevo detalle de pedido.</li> <li>3. Darle clic al botón de guardar.</li> <li>4. El sistema valida que todos los campos obligatorios estén completos.</li> <li>5. El sistema guarda el nuevo detalle de pedido en la base.</li> </ol>
<b>Observaciones:</b>	Tanto los administradores como los usuarios pueden agregar, modificar o eliminar un detalle de pedido.
<b>Mensajes de errores:</b>	<ol style="list-style-type: none"> <li>1. Si el registro es exitoso, el nuevo detalle de pedido queda registrado en la base de datos y aparece en la lista de detalles de pedido.</li> </ol>

### Caso de Uso 10. Registro de suministros

Figura 59

Diagrama de caso de uso - Registro Suministro

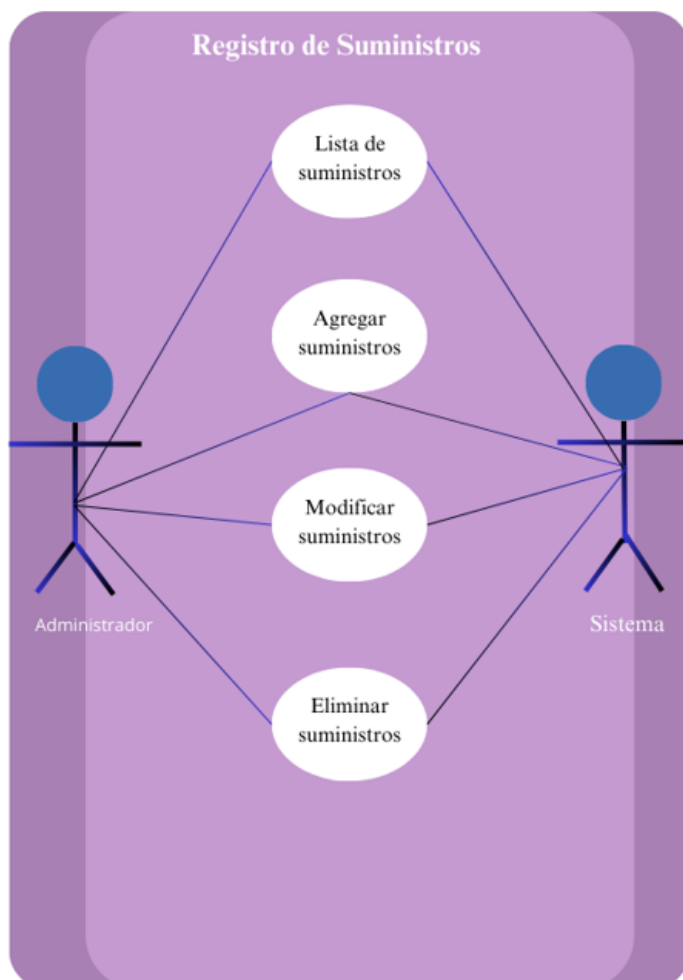


Figura 59. El diagrama de caso de uso pertenece al registro de suministro del sistema para la gestión y el control de inventario para la microempresa de Productos de Concreto Daniel.

Tabla 42

*Registro Suministro*

<b>Casos de uso: Registro suministros.</b>	
<b>Autora:</b>	Sofia Gamboa Navarro.
<b>Actores:</b>	<ol style="list-style-type: none"> <li>1. Administrador.</li> <li>2. Sistema.</li> </ol>
<b>Observaciones:</b>	<ol style="list-style-type: none"> <li>1. Lista los datos de los suministros.</li> <li>2. Agrega nuevos suministros.</li> <li>3. Modifica los datos de los suministros.</li> <li>4. Elimina los datos de los suministros.</li> </ol>
<b>Descripción:</b>	<ol style="list-style-type: none"> <li>1. Acceder a la página de suministros.</li> <li>2. Ingresar los datos requeridos del nuevo suministro.</li> <li>3. Darle clic al botón de guardar.</li> <li>4. El sistema valida que todos los campos obligatorios estén completos.</li> <li>5. El sistema guarda el nuevo suministro en la base de datos.</li> </ol>
<b>Observaciones:</b>	Solo los administradores pueden agregar, modificar o eliminar un suministro.
<b>Mensajes de errores:</b>	<ol style="list-style-type: none"> <li>1. Si el registro es exitoso, el nuevo suministro queda registrado en la base de datos y aparece en la lista de suministros.</li> </ol>

*Caso de Uso 11. Control de Bitácora Sesiones*

**Figura 60**

*Diagrama de caso de uso - Control de Bitácora Sesión*

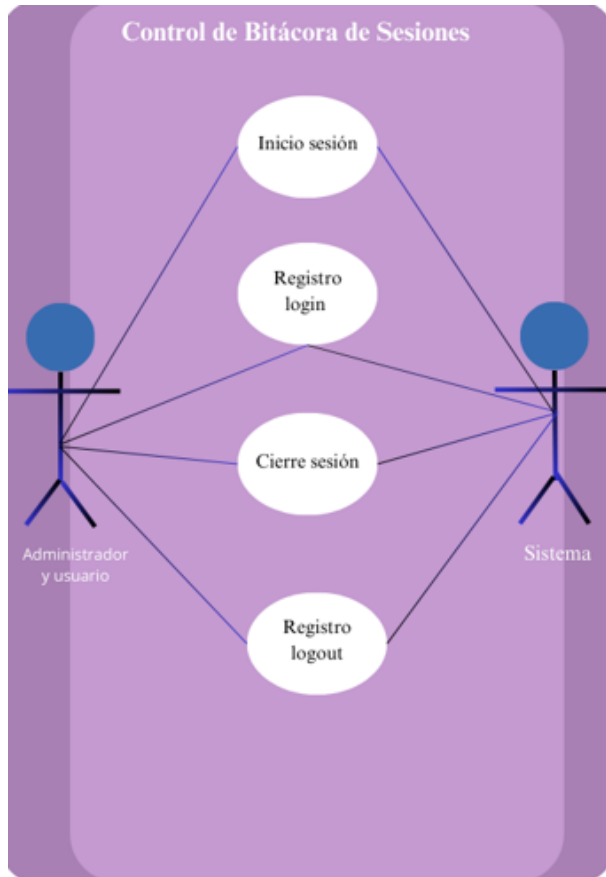


Figura 60. El diagrama de caso de uso pertenece al control de bitácora sesión del sistema para la gestión y el control de inventario para lo microempresa de Productos de Concreto Daniel.

**Tabla 43***Control de Bitácora Sesión*

<b>Casos de uso: Control de bitácora Sesión.</b>	
<b>Autora:</b>	Sofia Gamboa Navarro.
<b>Actores:</b>	<ol style="list-style-type: none"> <li>1. Administrador.</li> <li>2. Sistema.</li> </ol>
<b>Observaciones:</b>	Lista los datos de la bitácora Sesión.
<b>Descripción:</b>	<ol style="list-style-type: none"> <li>1. El usuario accede a la página de login e ingresa su correo electrónico y contraseña.</li> <li>2. El usuario presiona el botón “Inicio Sesión”.</li> <li>3. El sistema verifica las credenciales.</li> <li>4. Si las credenciales son correctas, el sistema obtiene el correo del usuario y lo redirige se al menú principal del sistema según el rol que tenga.</li> <li>5. El sistema registra la información en la tabla bitácora sesión con los siguientes datos, id del usuario evento, fecha y hora de inicio de sesión</li> <li>6. El sistema permite al usuario acceder a la plataforma.</li> <li>7. El usuario de la al botón de “Cerrar sesión”.</li> <li>8. El sistema se cierra y registra los eventos en la tabla bitácora sesión.</li> </ol>
<b>Observaciones:</b>	Solo los administradores pueden ver la información sobre la bitácora sesión.
<b>Mensajes de errores:</b>	N/A

### Caso de Uso 12. Dashboard

Figura 61

Diagrama de caso de uso - Control de Datos Dashboard

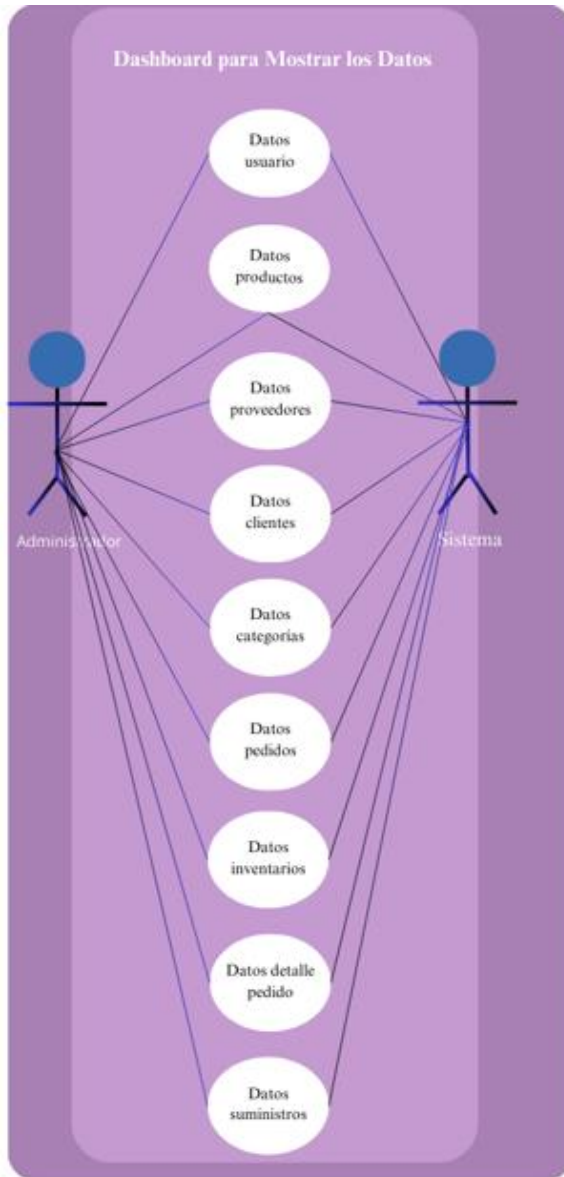


Figura 61. El diagrama de caso de uso pertenece al control de datos dashboard del sistema para la gestión y el control de inventario para lo microempresa de Productos de Concreto Daniel.

**Tabla 44***Control de Datos Dashboard*

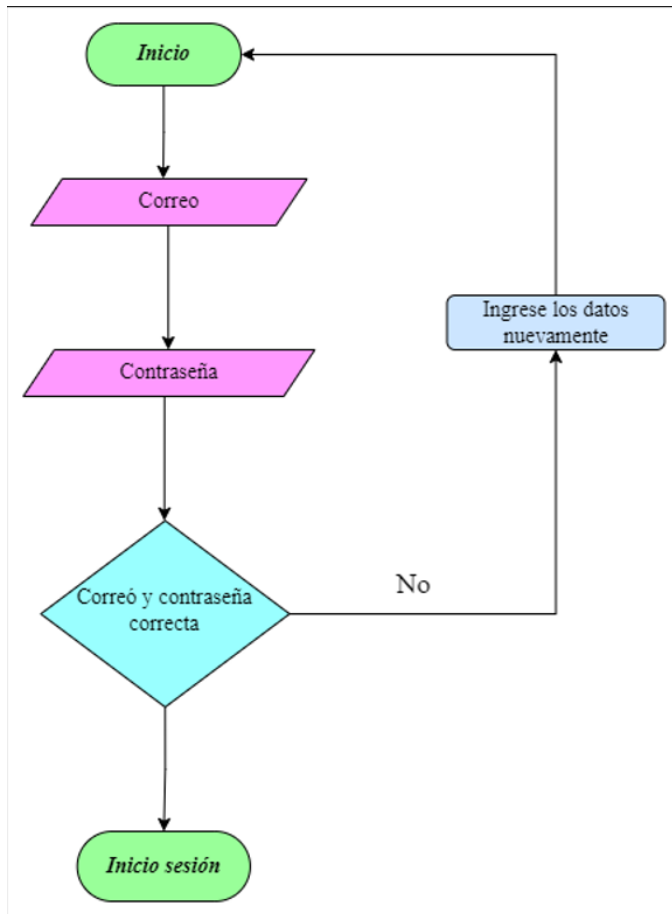
<b>Casos de uso: Control de Dashboard.</b>	
<b>Autora:</b>	Sofia Gamboa Navarro.
<b>Actores:</b>	<ol style="list-style-type: none"> <li>1. Administrador.</li> <li>2. Sistema.</li> </ol>
<b>Observaciones:</b>	Vista de dashboard.
<b>Descripción:</b>	<ol style="list-style-type: none"> <li>1. El administrador accede al sistema mediante su cuenta.</li> <li>2. El administrador puede ver el dashboard de datos apenas ingresa al menú principal.</li> </ol>
<b>Observaciones:</b>	Solo los administradores pueden ver la información sobre el dashboard.
<b>Mensajes de errores:</b>	N/A

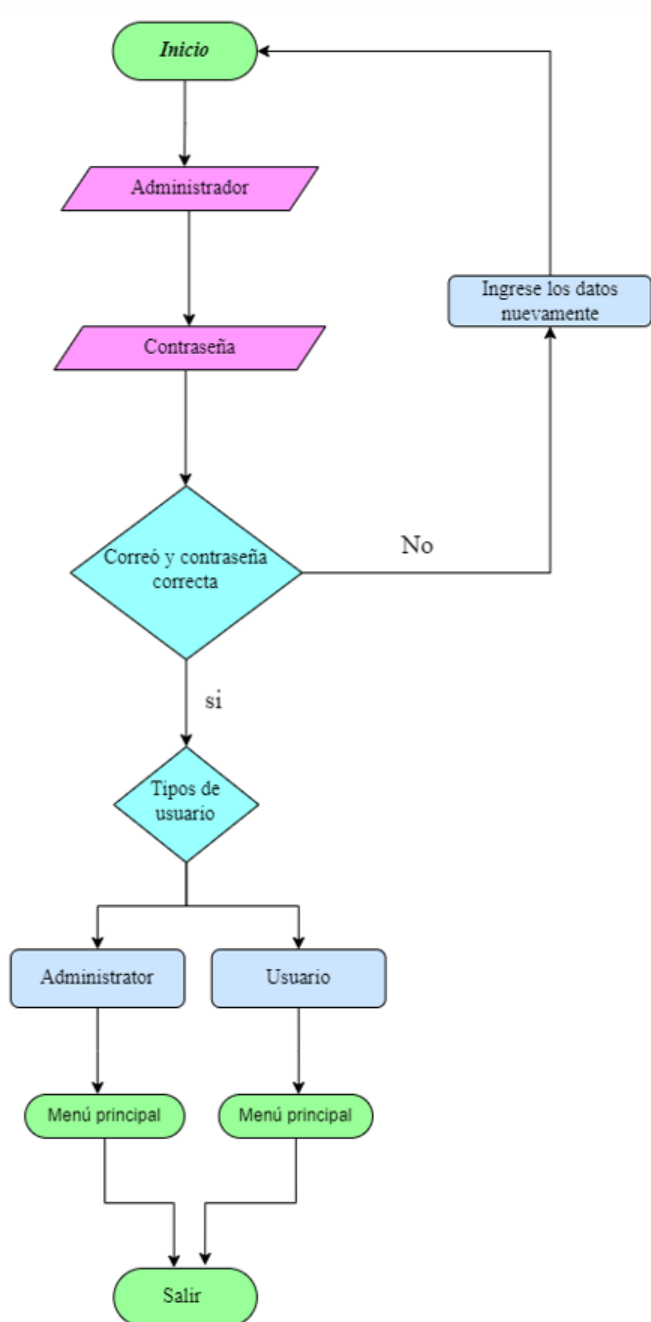
## Diagramas de flujo

### *Login*

**Figura 62**

*Diagrama de Flujo Login*

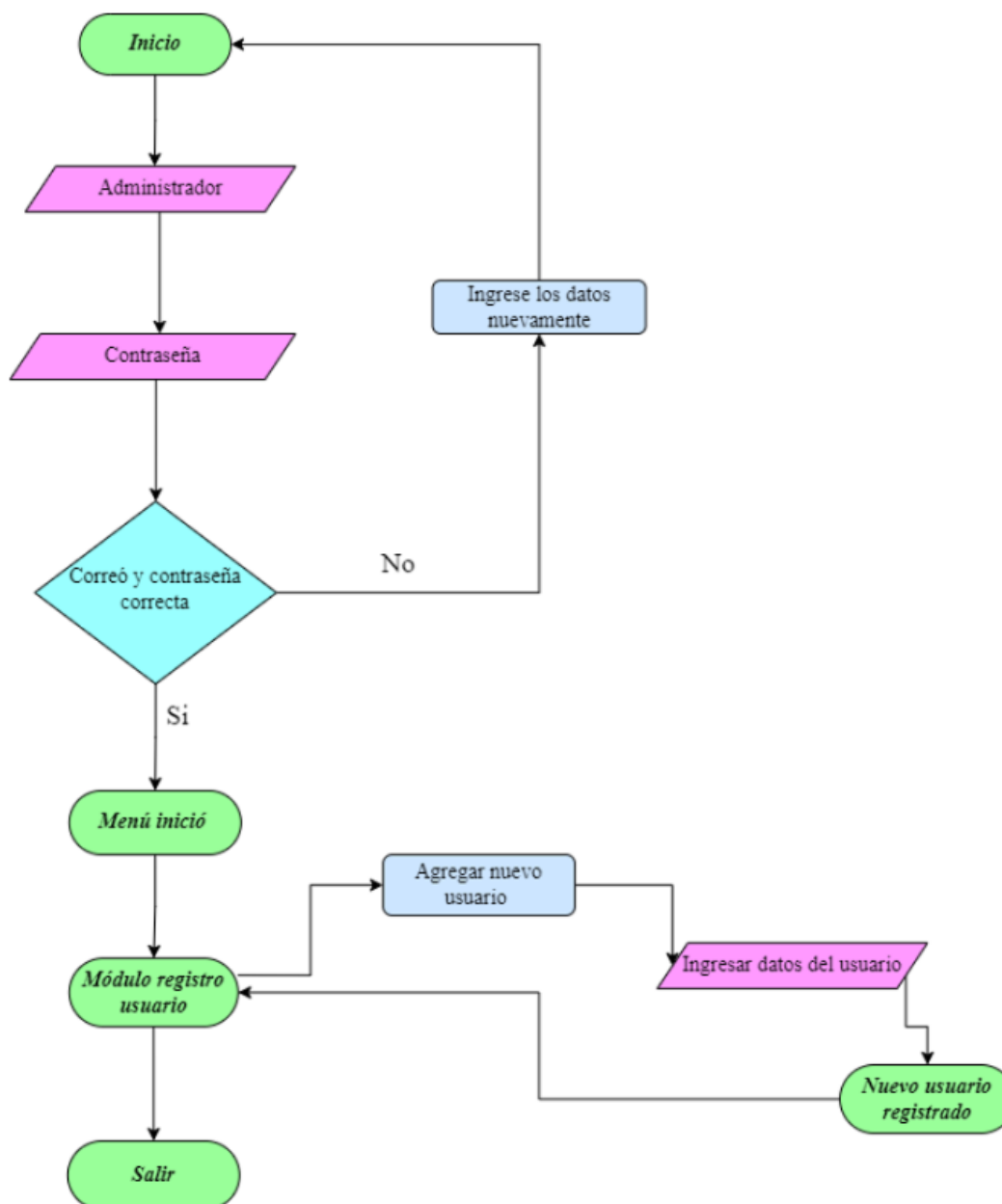


*Menú***Figura 63***Diagrama de Flujo Menú*

## Registro Usuario

Figura 64

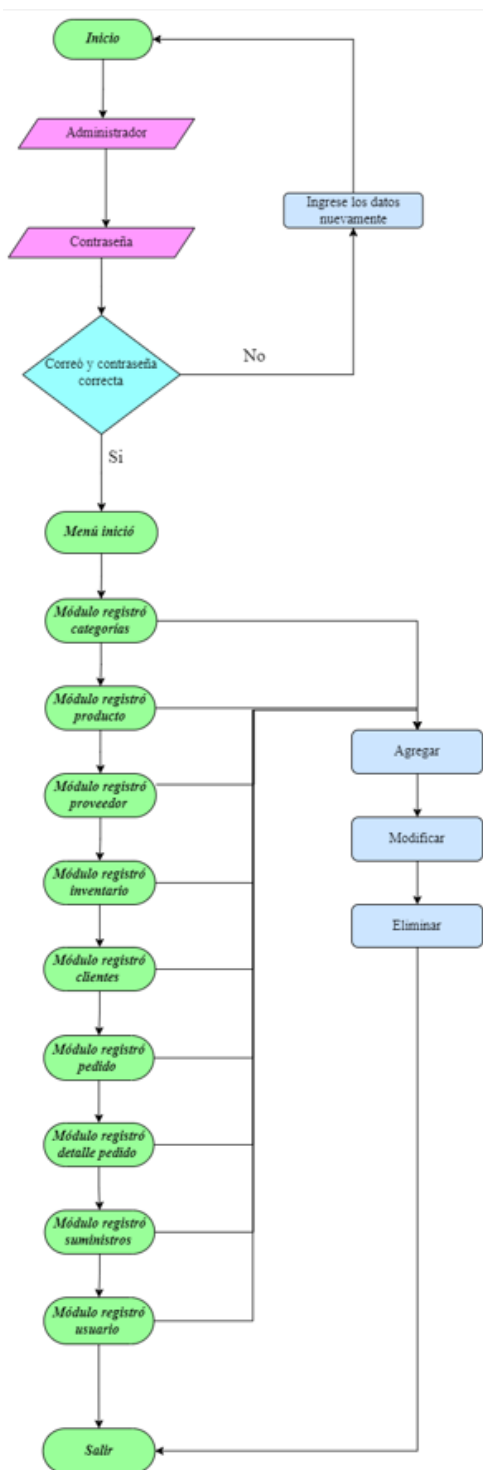
Diagrama de Flujo Registro Usuario



## Mantenimiento

Figura 65

Diagrama de Flujo Mantenimiento

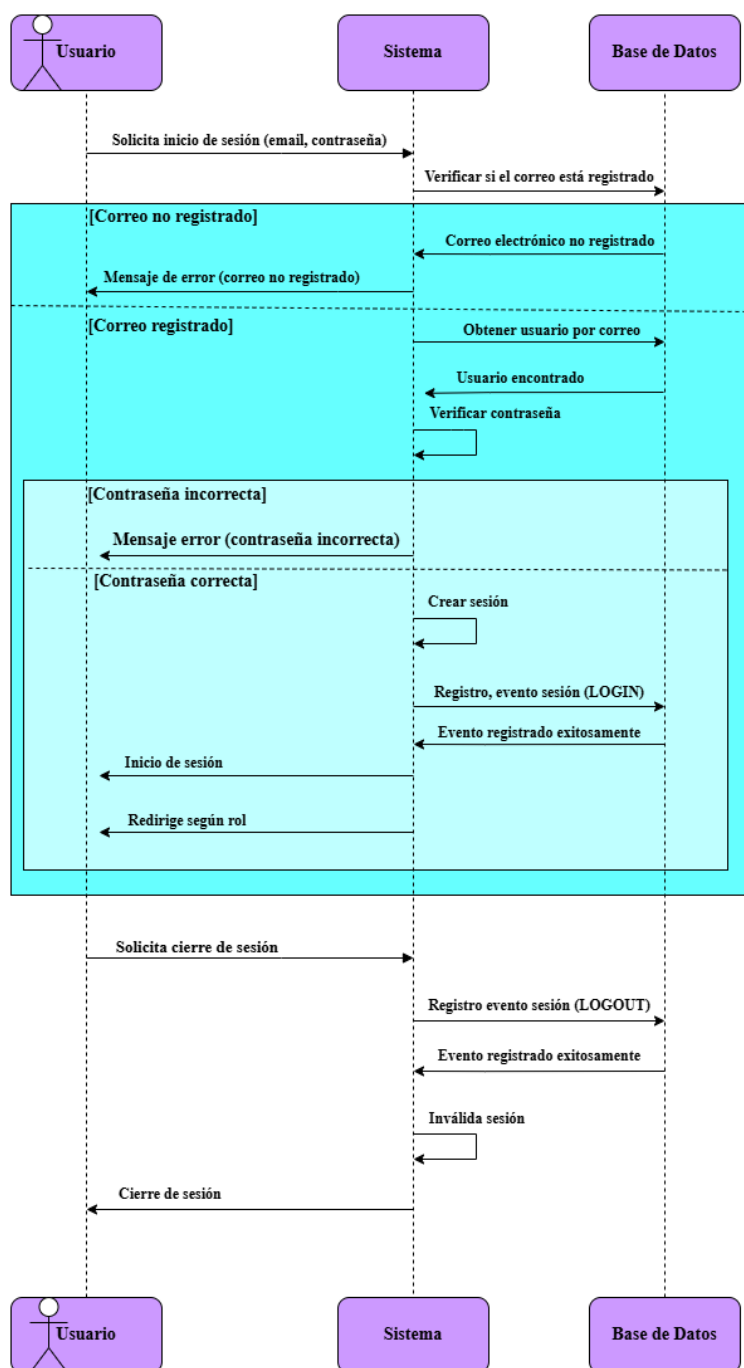


## Diagramas de secuencia

### Login

Figura 66

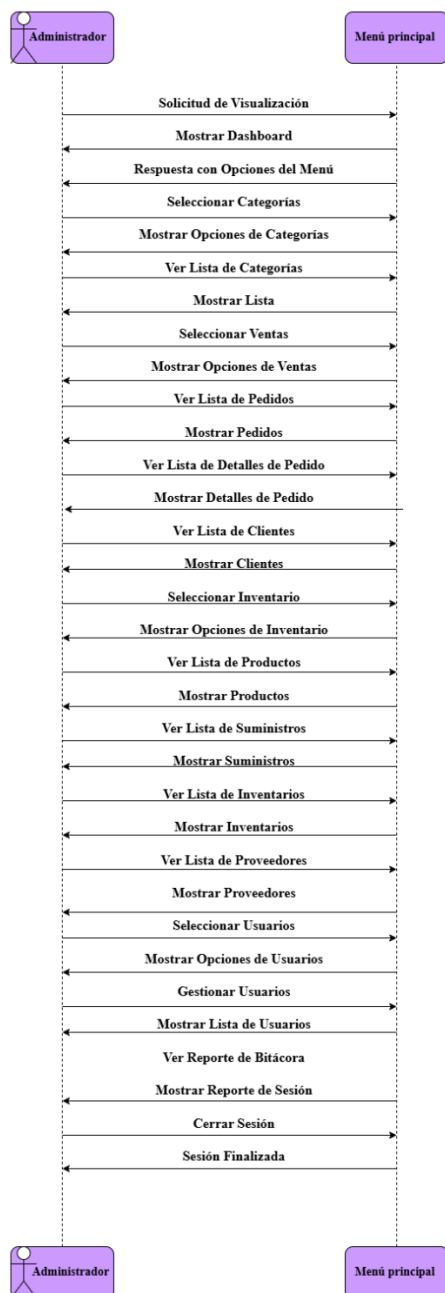
Diagrama de secuencia Login



## Menú Principal

Figura 67

### Diagrama de Secuencia Menú Principal



## Mantenimientos y Búsqueda

Figura 68

### Diagrama de Secuencia Mantenimiento Registro Usuarios

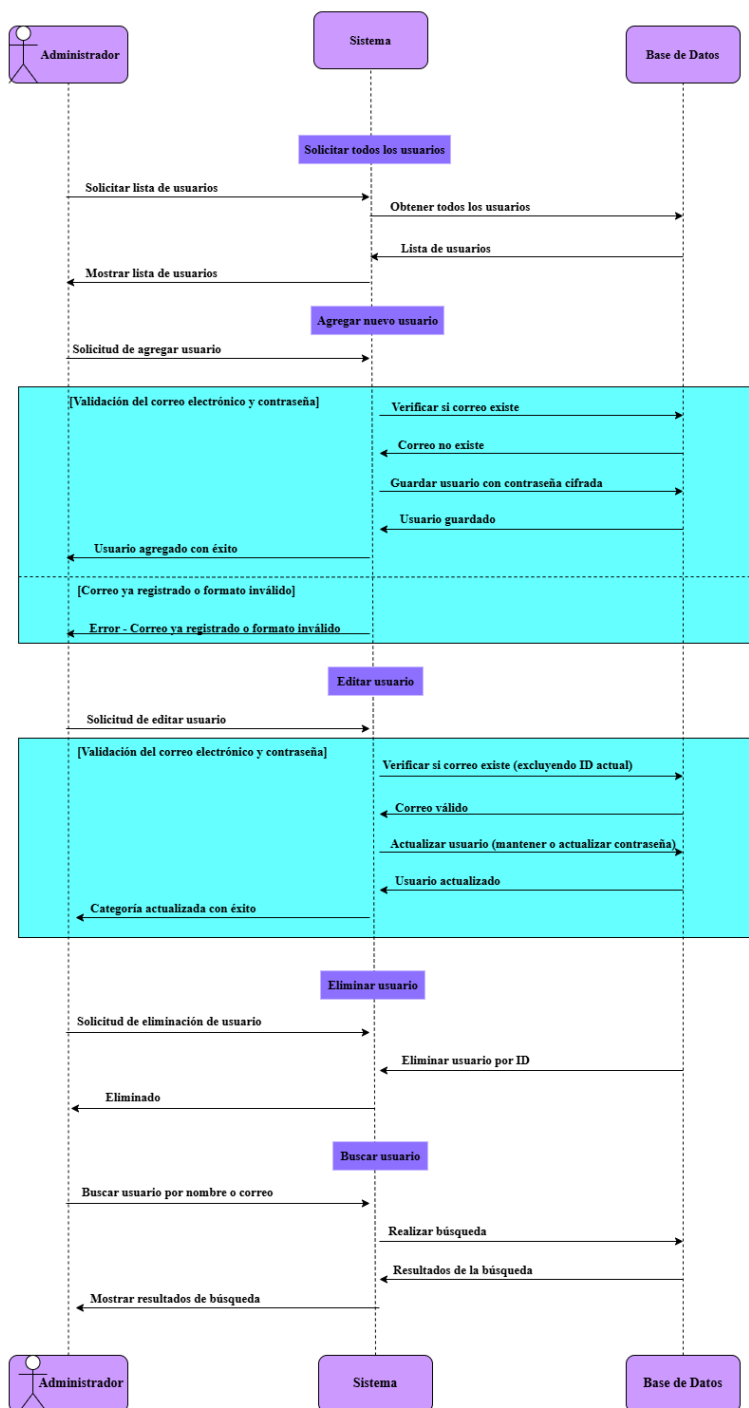


Figura 69

## Diagrama de Secuencia Mantenimiento de Categorías

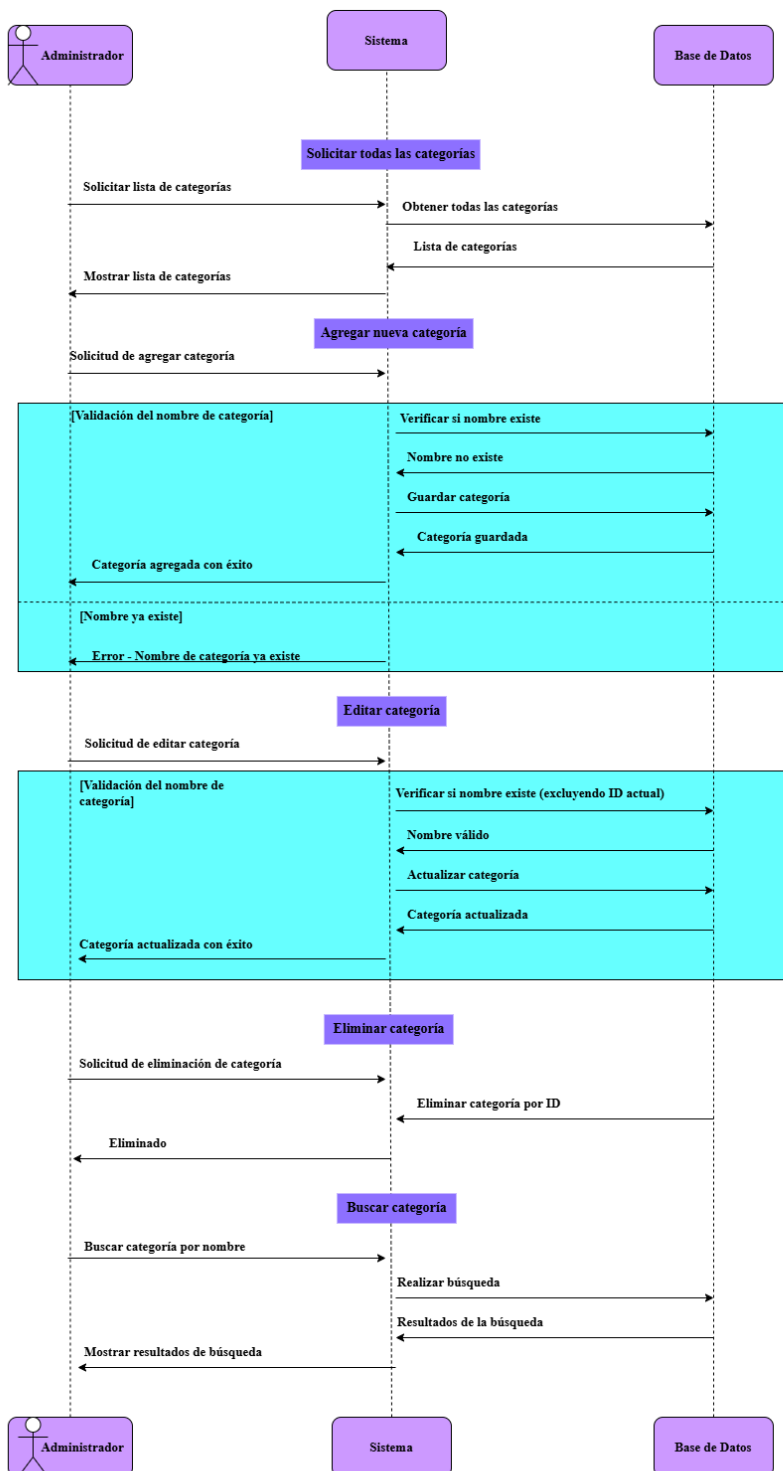


Figura 70

## Diagrama de Secuencia Mantenimiento Pedidos

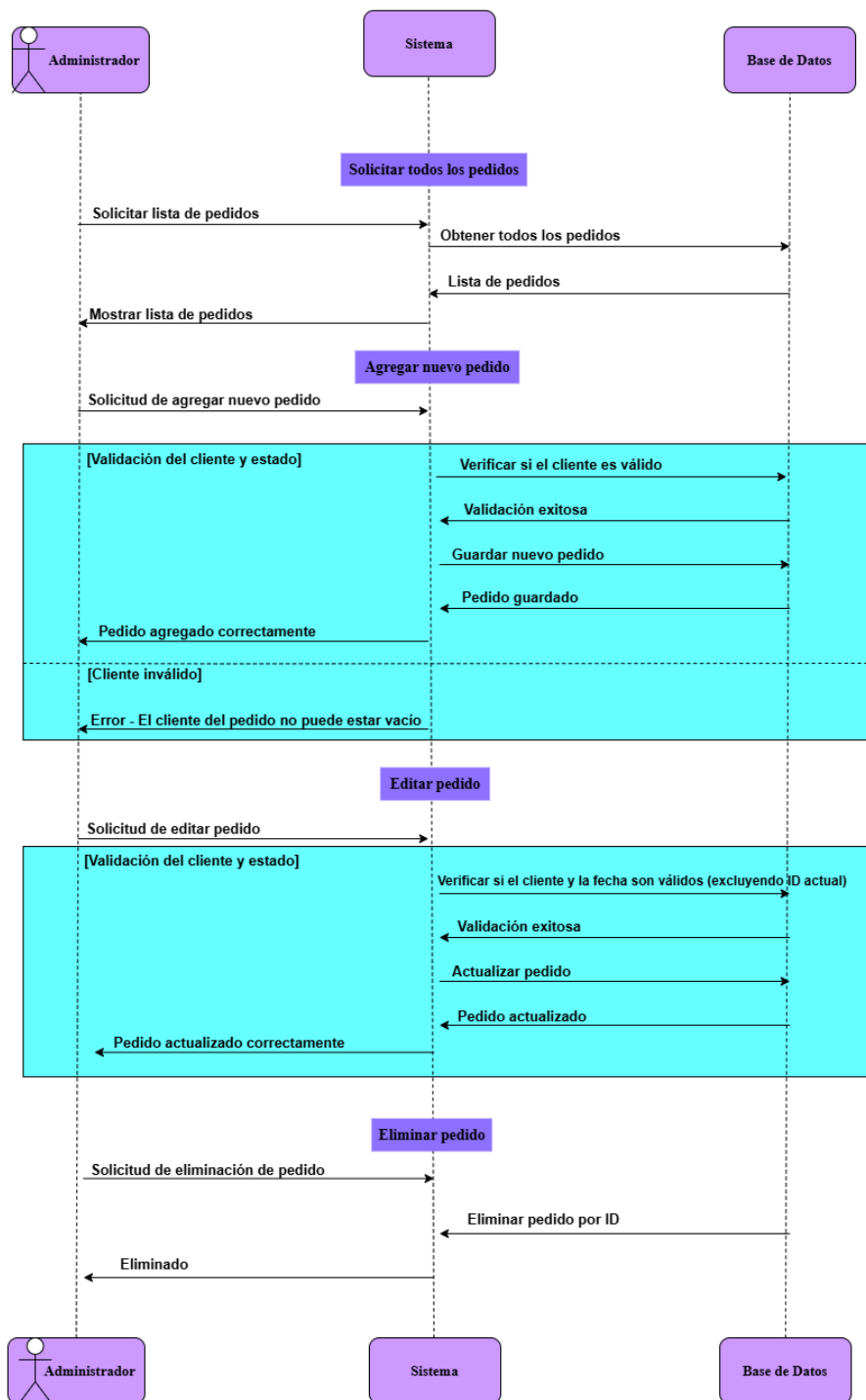


Figura 71

## Diagrama de Secuencia Mantenimiento Detalles Pedidos

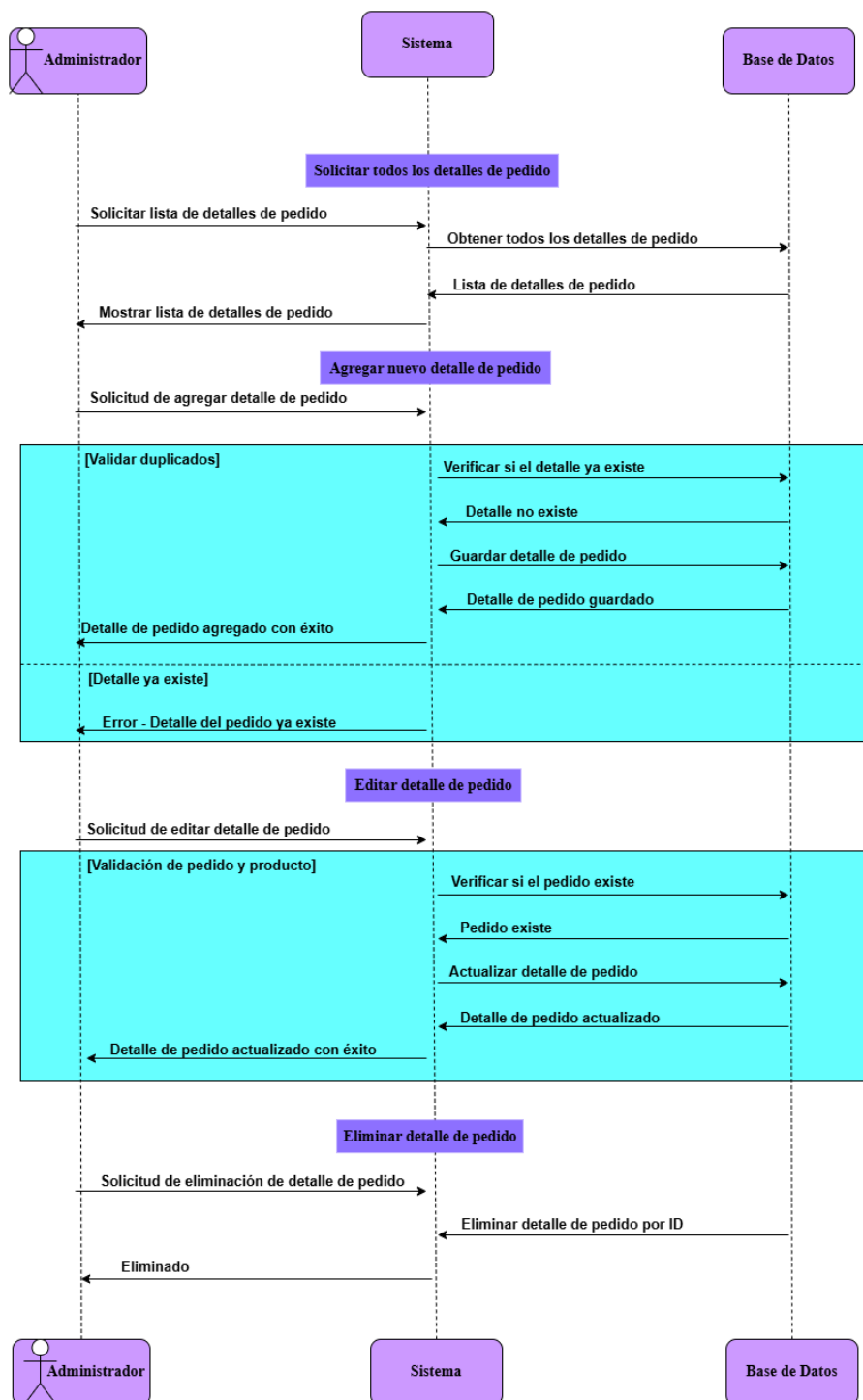


Figura 72

## Diagrama de Secuencia Mantenimiento Clientes

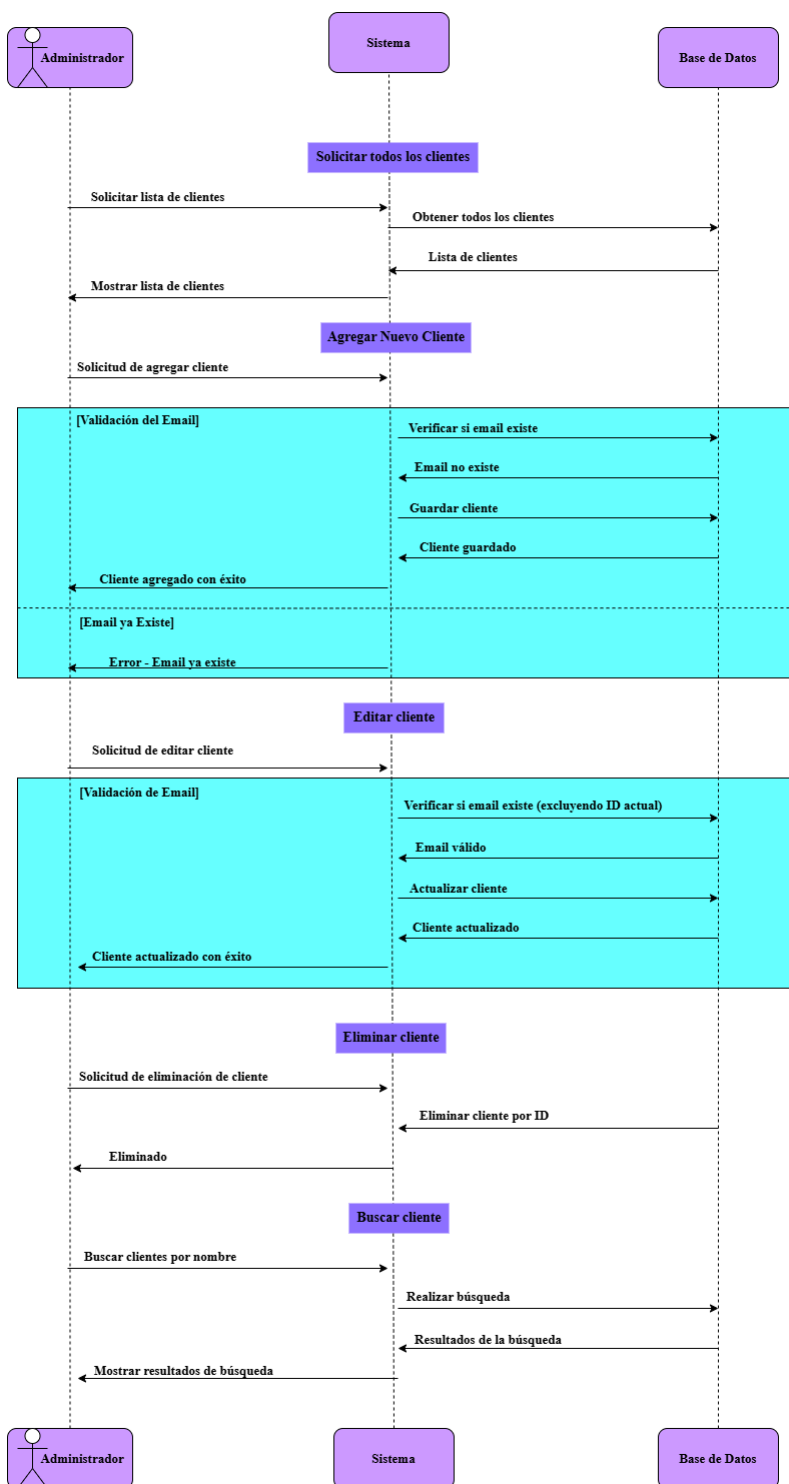


Figura 73

## Diagrama de Secuencia Mantenimiento Productos

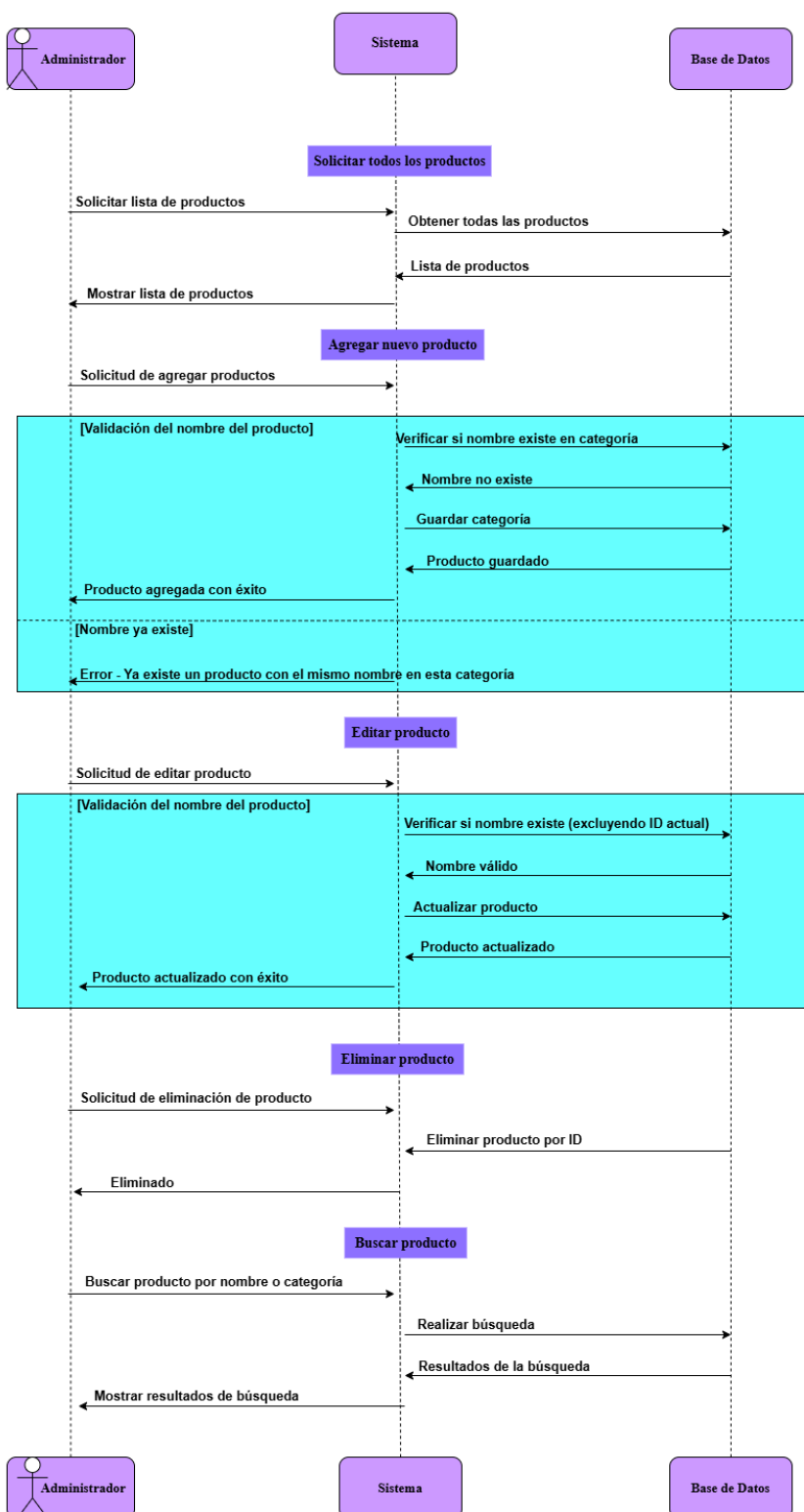


Figura 74

## Diagrama de Secuencia Mantenimiento Inventario

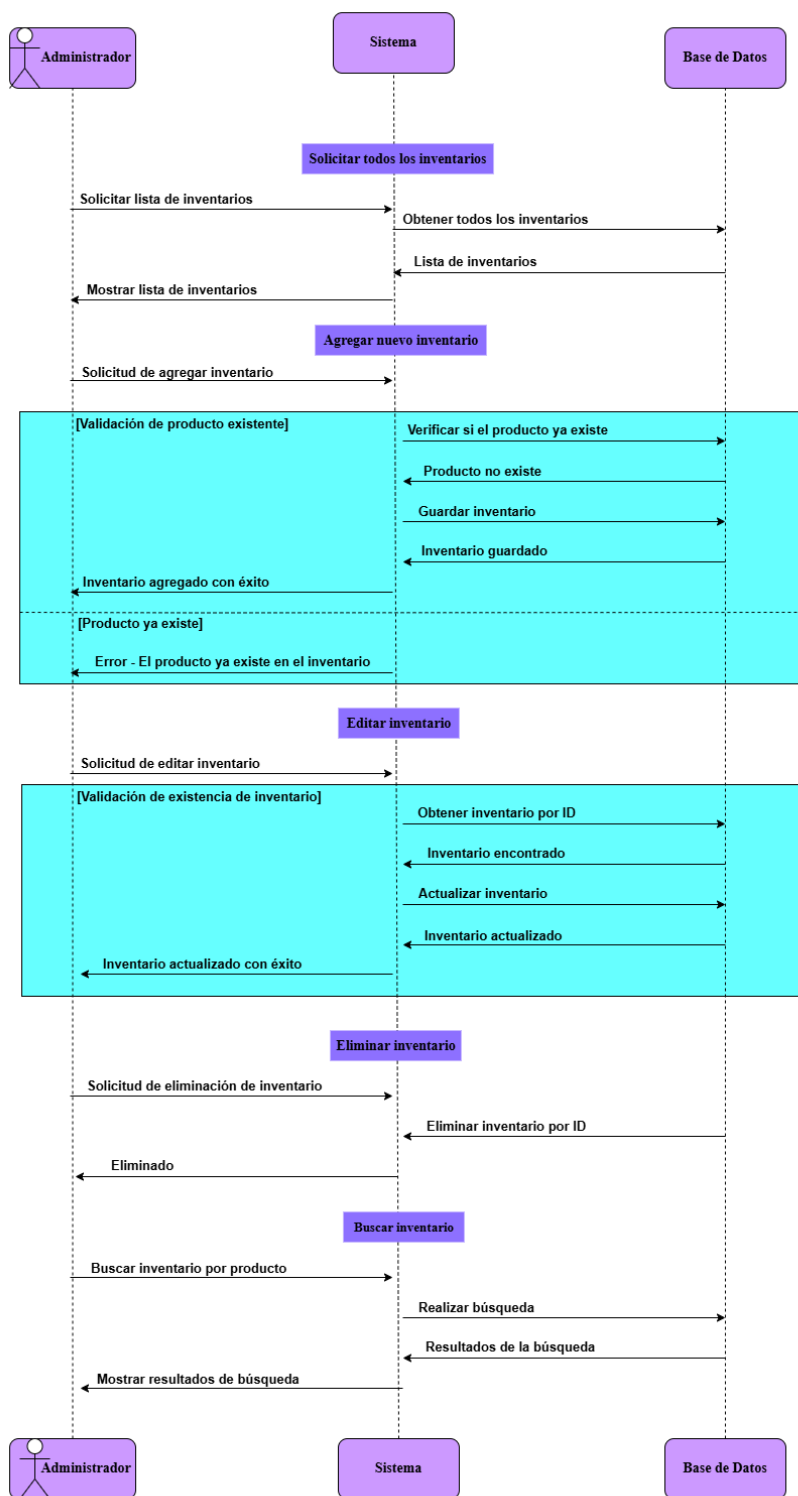


Figura 75

## Diagrama de Secuencia Mantenimiento Suministros

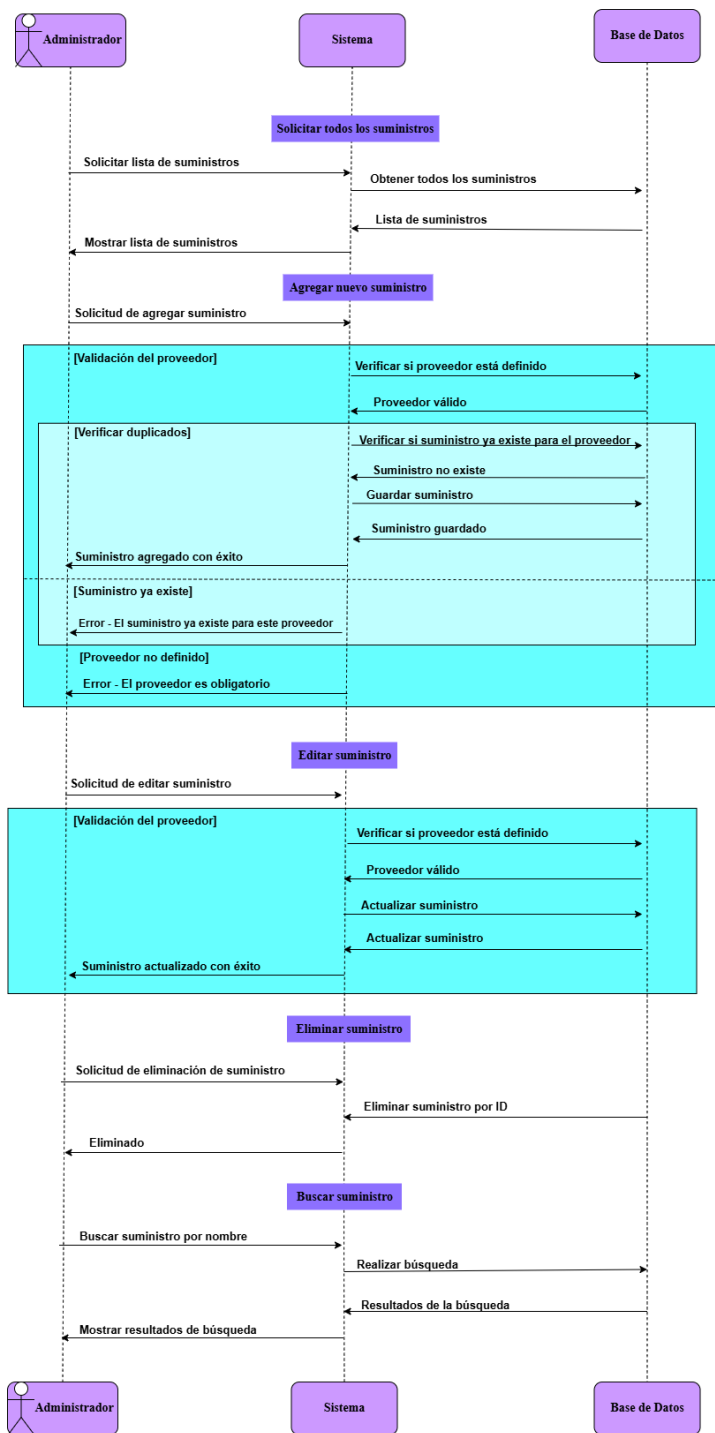


Figura 76

## Diagrama de Secuencia Mantenimiento Proveedores

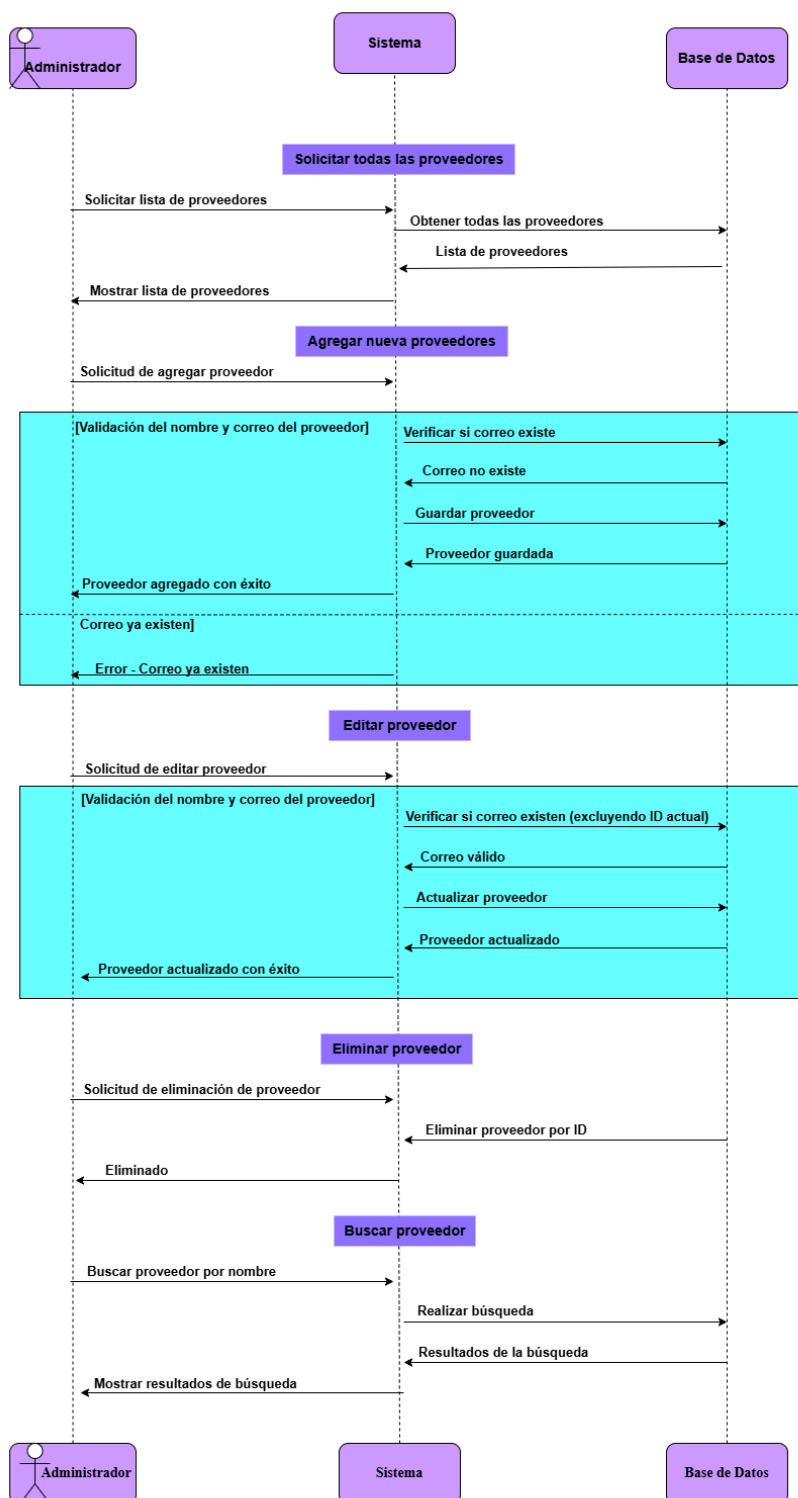
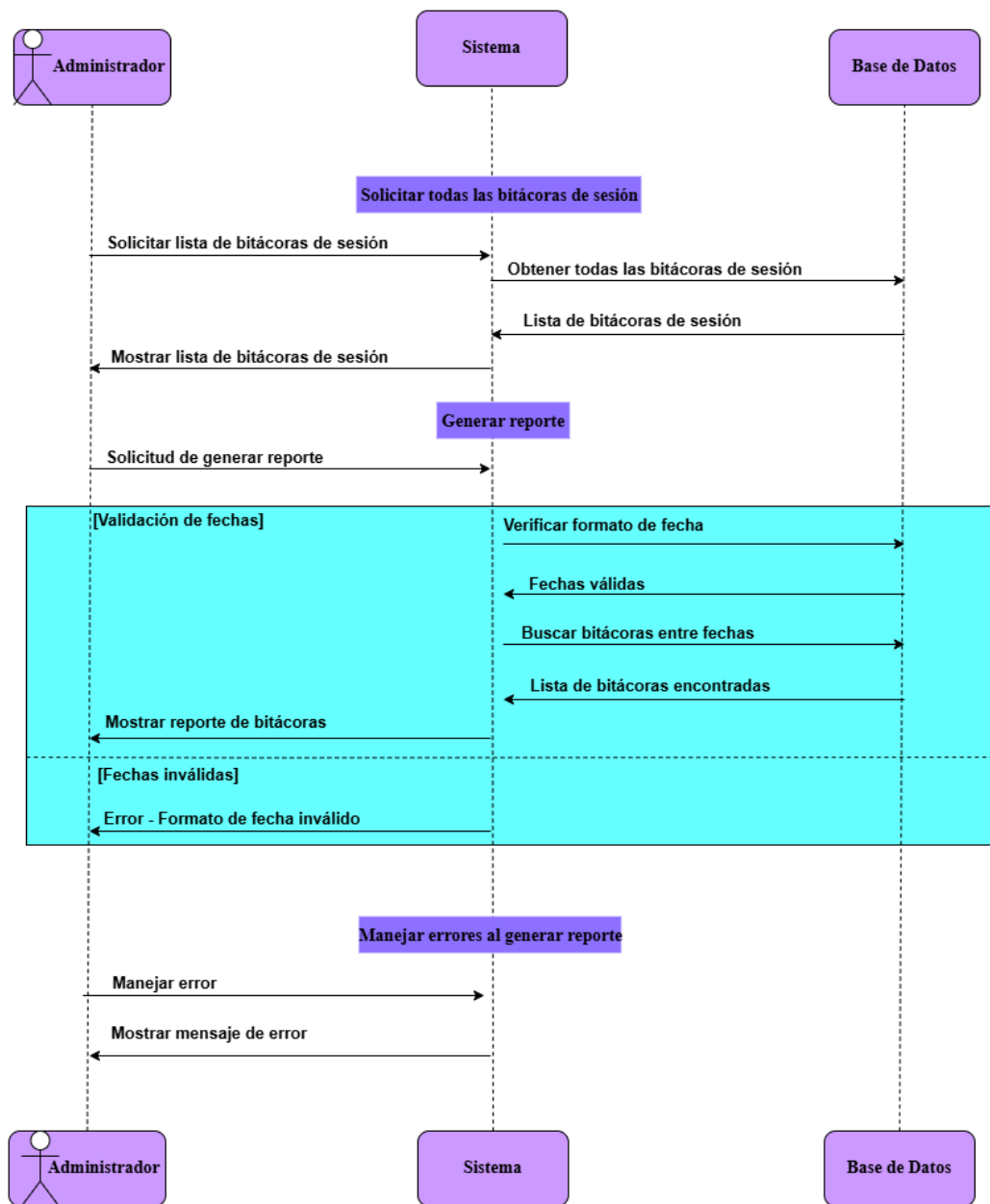
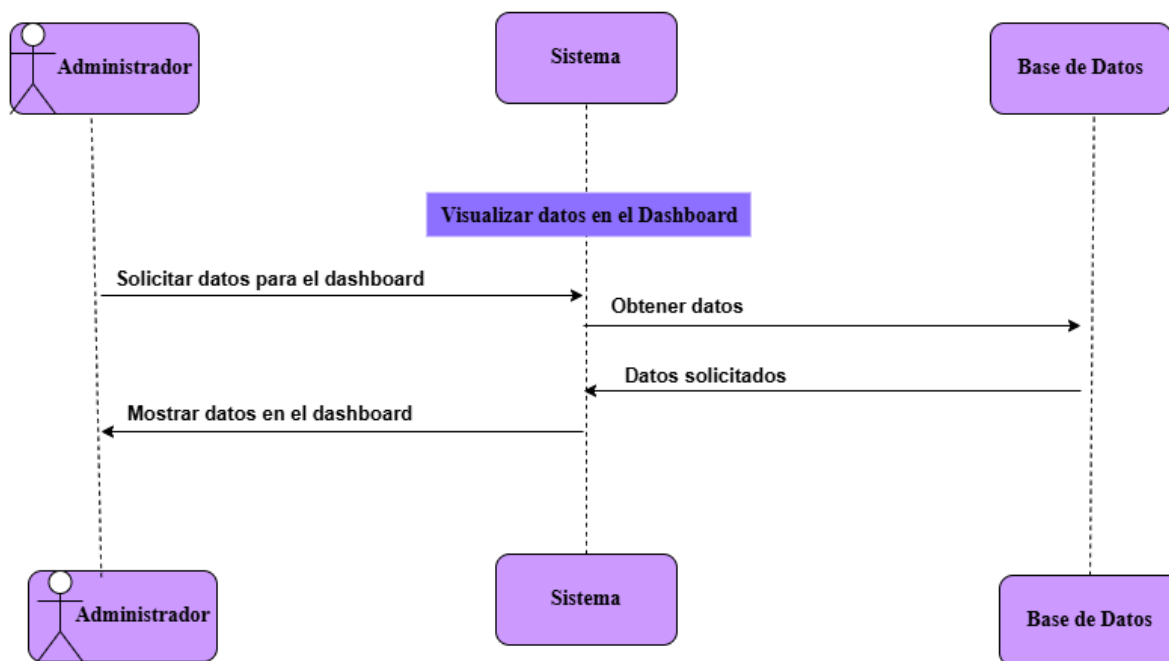


Figura 77

## Diagrama de Secuencia Bitácora Sesión

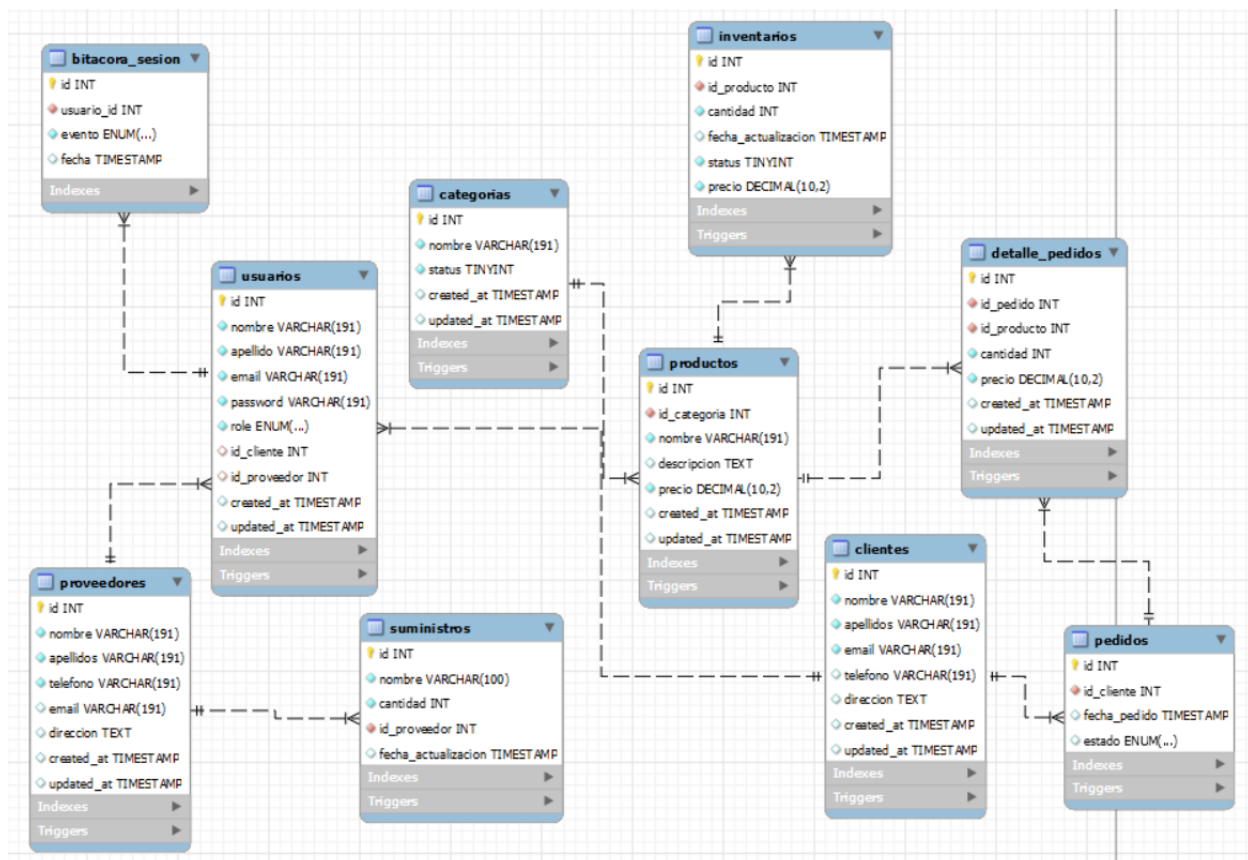


*Datos Dashboard***Figura 78***Diagrama de Secuencia Dashboard*

## Modelo Entidad Relación

Figura 79

Modelo Relacional



## Diccionario de datos

**Tabla 45**

*Tabla Categorías*

Columna	Tipo	Nulo	Predeterminado
<b>id (Primaria)</b>	int (11)	No	
<b>nombre</b>	varchar (191)	No	
<b>status</b>	tinyint	No	1
<b>created_at</b>	timestamp	No	current_timestamp ()
<b>updated_at</b>	timestamp	No	current_timestamp ()

**Tabla 46**

*Tabla Proveedores*

Columna	Tipo	Nulo	Predeterminado
<b>id (Primaria)</b>	int (11)	No	
<b>nombre</b>	varchar (191)	No	
<b>apellidos</b>	varchar (191)	No	
<b>teléfono</b>	varchar (191)	No	
<b>email</b>	varchar (191)	No	
<b>dirección</b>	text	Sí	
<b>created_at</b>	timestamp	No	current_timestamp ()
<b>updated_at</b>	timestamp	No	current_timestamp ()

**Tabla 47***Tabla Productos*

Columna	Tipo	Nulo	Predeterminado
<b>id (Primaria)</b>	int (11)	No	
<b>id_categoria</b>	int (11)	No	
<b>nombre</b>	Varchar (191)	No	
<b>descripción</b>	text	Sí	
<b>precio</b>	Decimal (10, 2)	No	
<b>created_at</b>	timestamp	No	current_timestamp ()
<b>updated_at</b>	timestamp	No	current_timestamp ()

**Tabla 48***Tabla Inventarios*

Columna	Tipo	Nulo	Predeterminado
<b>id (Primaria)</b>	int (11)	No	
<b>id_producto</b>	int (11)	No	
<b>cantidad</b>	int	No	0
<b>status</b>	tinyint	No	1
<b>precio</b>	decimal (10, 2)	No	0.00
<b>fecha_actualizacion</b>	timestamp	No	current_timestamp ()

**Tabla 49***Tabla Clientes*

Columna	Tipo	Nulo	Predeterminado
<b>id (Primaria)</b>	int (11)	No	
<b>nombre</b>	varchar (191)	No	
<b>apellidos</b>	varchar (191)	No	
<b>email</b>	varchar (191)	No	
<b>telefono</b>	varchar (191)	Sí	
<b>dirección</b>	text	Sí	
<b>created_at</b>	timestamp	No	current_timestamp ()
<b>updated_at</b>	timestamp	No	current_timestamp ()

**Tabla 50***Tabla Pedidos*

Columna	Tipo	Nulo	Predeterminado
<b>id (Primaria)</b>	int (11)	No	
<b>id_cliente</b>	int (11)	No	
<b>fecha_pedido</b>	timestamp	No	current_timestamp ()
<b>estado</b>	enum	No	'Pendiente'

**Tabla 51***Tabla Detalles de Pedidos*

Columna	Tipo	Nulo	Predeterminado
<b>id (Primaria)</b>	int (11)	No	
<b>id_pedido</b>	int (11)	No	
<b>id_producto</b>	int (11)	No	
<b>cantidad</b>	int	No	1
<b>precio</b>	decimal (10, 2)	No	0.00
<b>created_at</b>	timestamp	No	current_timestamp ()
<b>updated_at</b>	timestamp	No	current_timestamp ()

**Tabla 52***Tabla Suministros*

Columna	Tipo	Nulo	Predeterminado
<b>id (Primaria)</b>	int (11)	No	
<b>nombre</b>	varchar (100)	No	
<b>cantidad</b>	int	No	0
<b>id_proveedor</b>	int (11)	No	
<b>fecha_actualizacion</b>	timestamp	No	current_timestamp ()

**Tabla 53***Tabla Usuarios*

Columna	Tipo	Nulo	Predeterminado
<b>id (Primaria)</b>	int (11)	No	
<b>nombre</b>	varchar (191)	No	
<b>apellido</b>	varchar (191)	No	
<b>email</b>	varchar (191)	No	
<b>password</b>	Varchar (191)	No	
<b>role</b>	enum	No	'user'
<b>id_cliente</b>	int (11) NULL	Sí	
<b>id_proveedor</b>	int (11) NULL	Sí	
<b>created_at</b>	timestamp	No	current_timestamp ()
<b>updated_at</b>	timestamp	No	current_timestamp ()

**Tabla 54***Tabla Bitácora de Sesión*

Columna	Tipo	Nulo	Predeterminado
<b>id (Primaria)</b>	int (11)	No	
<b>usuario_id</b>	int (11)	No	
<b>evento</b>	enum	No	'LOGIN'
<b>fecha</b>	timestamp	No	current_timestamp ()

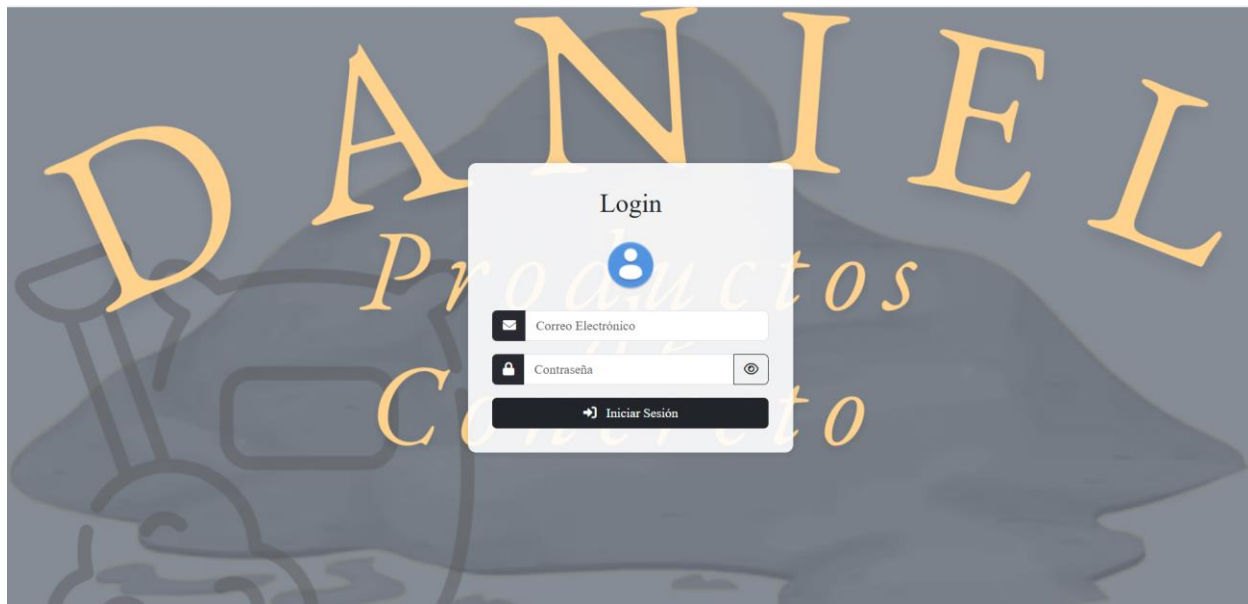
## Desarrollo del sistema

### *Pantalla del Login*

Pantalla para logearse

### **Figura 80**

*Login*

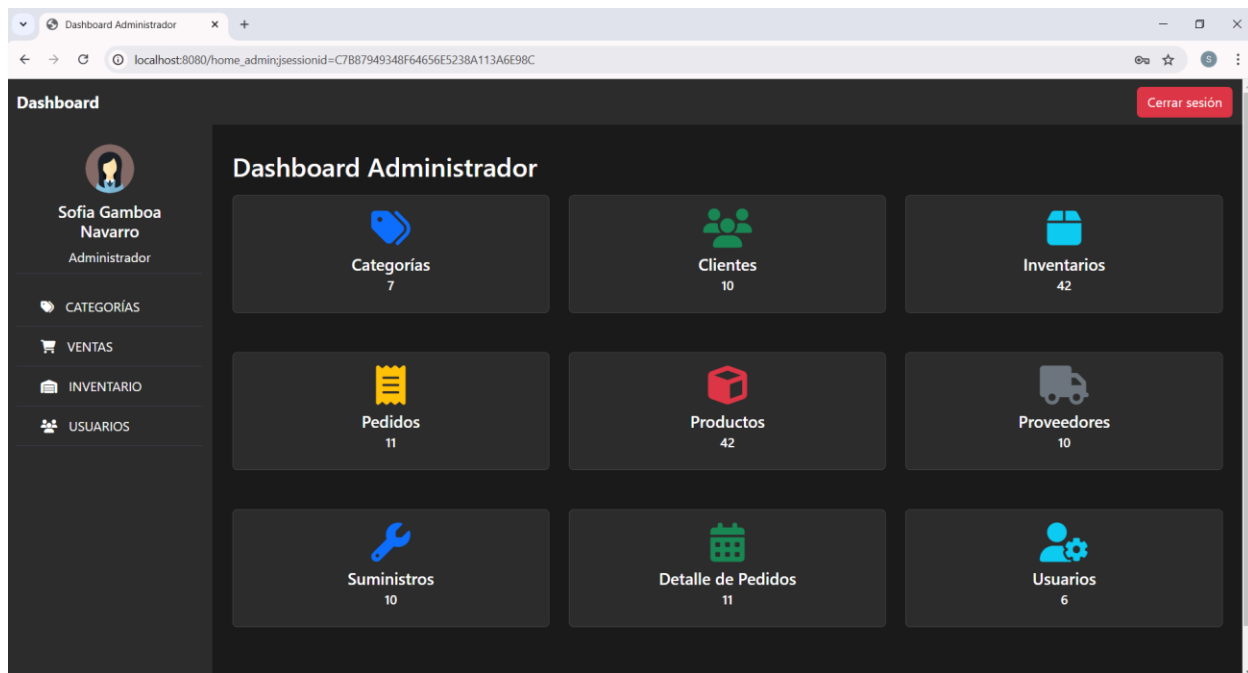


## *Pantalla de ingreso al sistema*

Pantalla que ve el usuario al ingresar al sistema.

### **Figura 81**

## *Pantalla Principal*

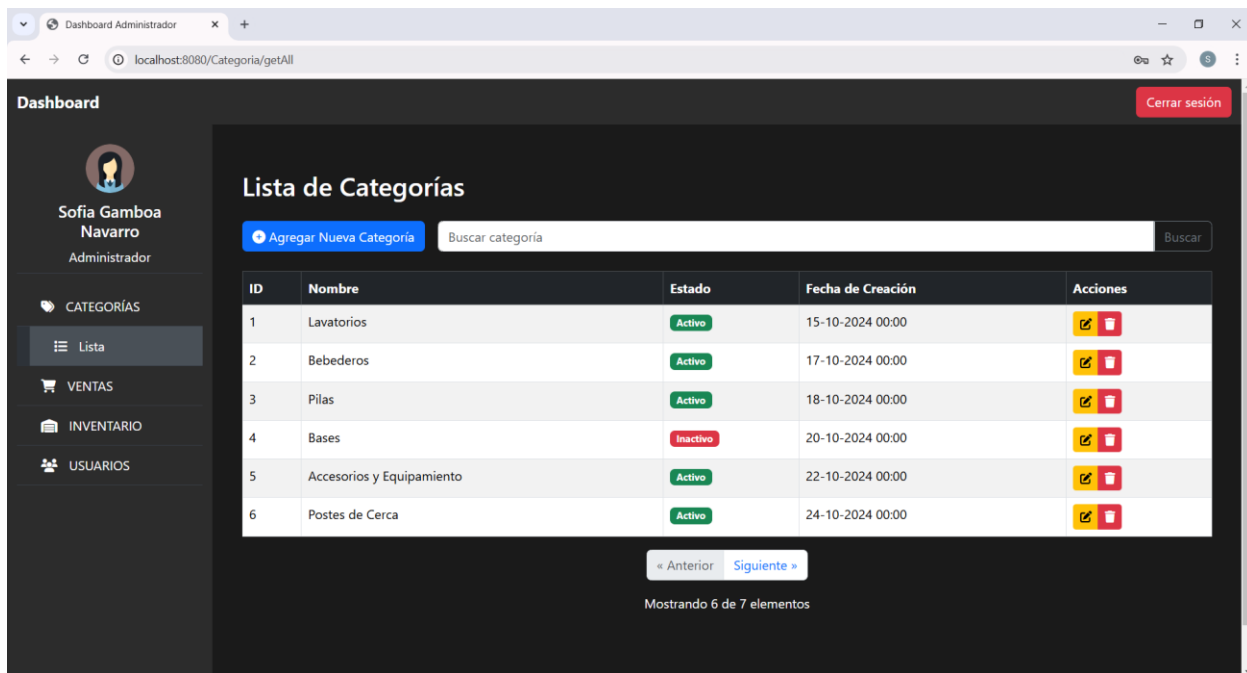


## Pantallas de Categoría

Pantalla en la cual se muestra el mantenimiento de categoría.

### Figura 82

#### Pantalla Listar Categorías



Dashboard Administrator

localhost:8080/Categoria/getAll

Dashboard Cerrar sesión













Sofía Gamboa Navarro Administrador

CATEGORÍAS

- Lista
- VENTAS
- INVENTARIO
- USUARIOS

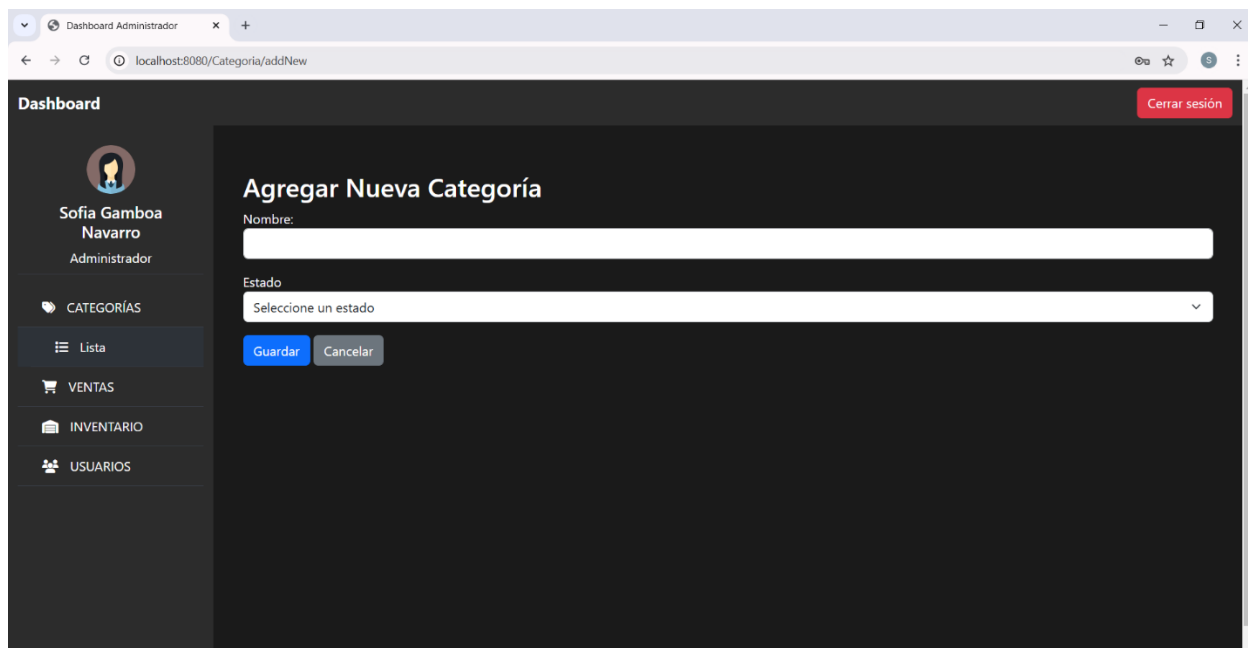
### Lista de Categorías

[+ Agregar Nueva Categoría](#)  Buscar

ID	Nombre	Estado	Fecha de Creación	Acciones
1	Lavatorios	Activo	15-10-2024 00:00	 
2	Bebederos	Activo	17-10-2024 00:00	 
3	Pilas	Activo	18-10-2024 00:00	 
4	Bases	Inactivo	20-10-2024 00:00	 
5	Accesorios y Equipamiento	Activo	22-10-2024 00:00	 
6	Postes de Cerca	Activo	24-10-2024 00:00	 

« Anterior [Siguiente](#) »

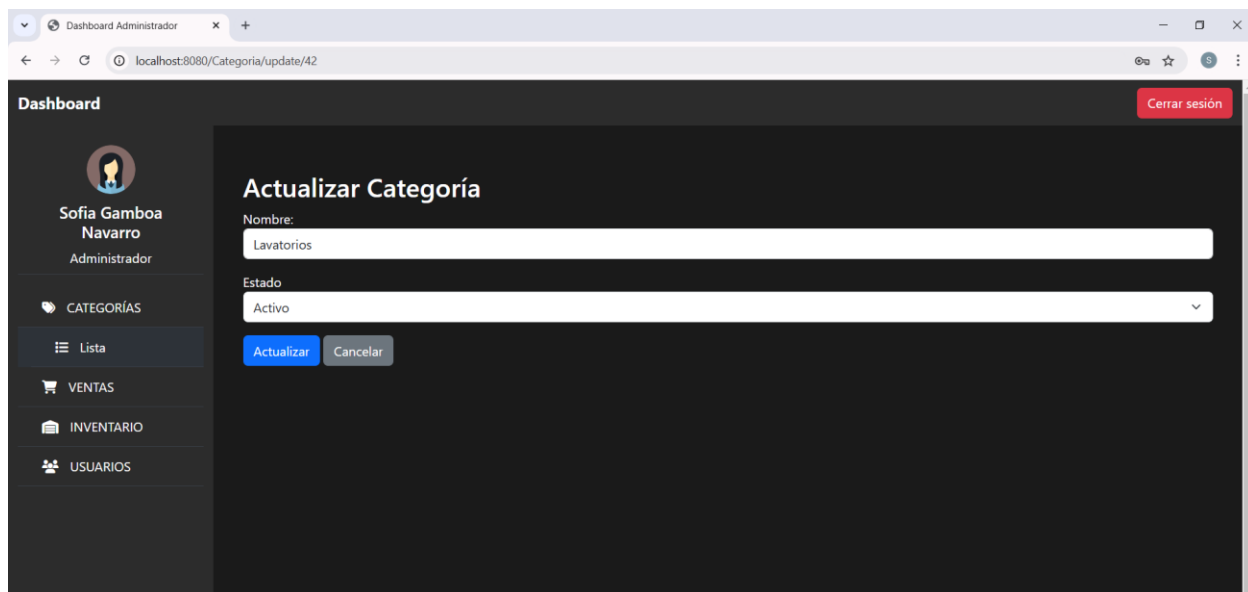
Mostrando 6 de 7 elementos

**Figura 83***Pantalla Agregar Categorías*

The screenshot shows a web browser window with the URL `localhost:8080/Categoria/addNew`. The page is titled "Agregar Nueva Categoría" and is part of a dashboard for "Sofia Gamboa Navarro, Administrador". The dashboard sidebar includes a menu with "CATEGORÍAS" (selected), "Lista", "VENTAS", "INVENTARIO", and "USUARIOS". The main content area contains a form with the following fields:

- Nombre:** A text input field.
- Estado:** A dropdown menu with the selected option "Seleccione un estado".
- Buttons:** "Guardar" (blue) and "Cancelar" (grey).

A "Cerrar sesión" button is located in the top right corner of the dashboard.

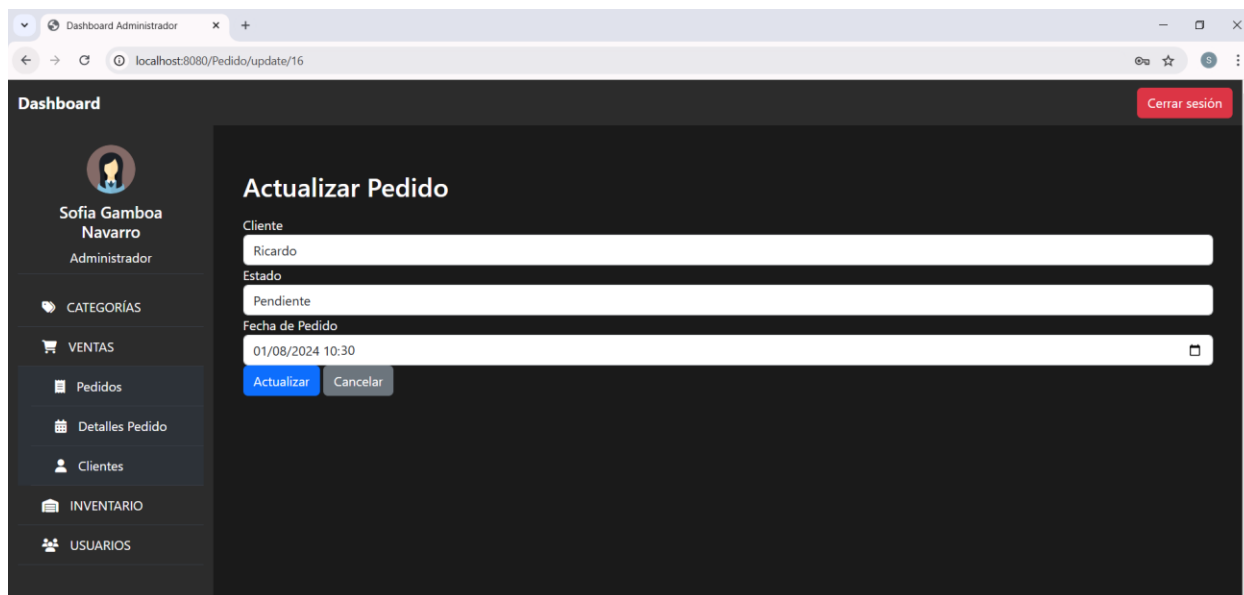
**Figura 84***Pantalla Actualizar Categorías*

The screenshot shows a web browser window with the URL `localhost:8080/Categoria/update/42`. The page is titled "Actualizar Categoría" and is part of the same dashboard as Figure 83. The sidebar menu is identical, with "CATEGORÍAS" selected. The main content area contains a form with the following fields:

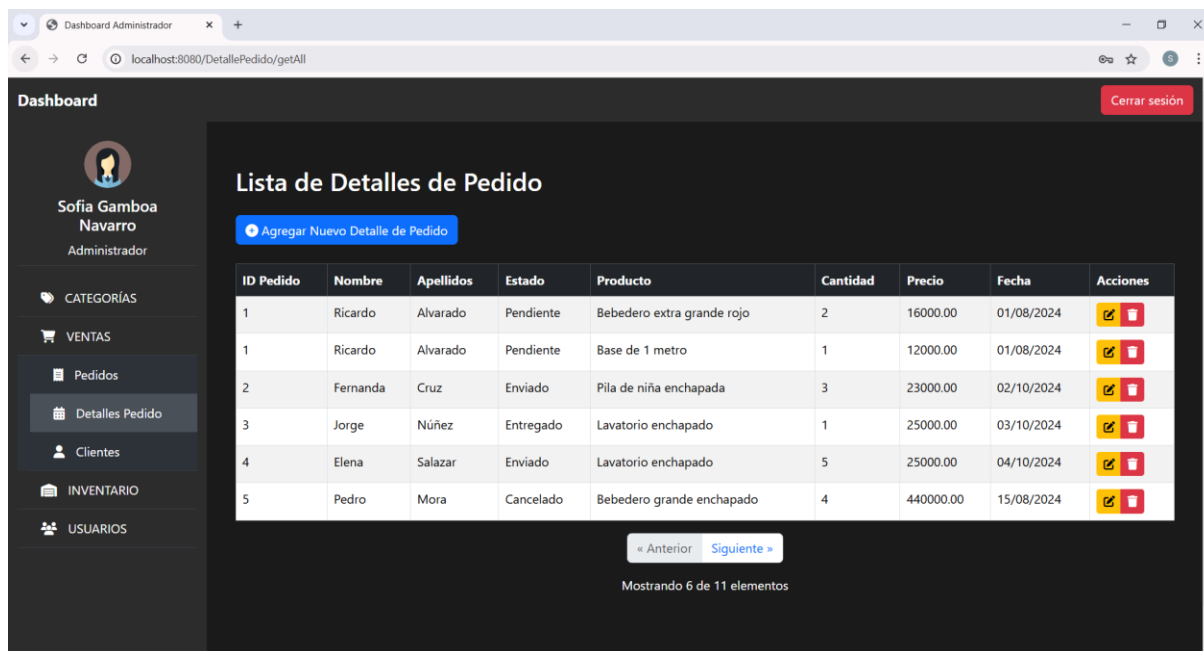
- Nombre:** A text input field containing the value "Lavatorios".
- Estado:** A dropdown menu with the selected option "Activo".
- Buttons:** "Actualizar" (blue) and "Cancelar" (grey).

A "Cerrar sesión" button is located in the top right corner of the dashboard.



**Figura 87***Pantalla Actualizar Pedidos**Pantallas de Detalles de los Pedidos*

Pantalla en la cual se muestra el mantenimiento de detalles de los pedidos.

**Figura 88***Pantalla Listar Detalles de Pedidos*

**Figura 89***Pantalla Agregar Detalles de Pedidos*

The screenshot shows a web browser window with the URL `localhost:8080/DetallePedido/addNew`. The dashboard header includes the name **Sofia Gamboa Navarro** and the role **Administrador**. A sidebar menu on the left contains the following items: **CATEGORÍAS**, **VENTAS**, **Pedidos**, **Detalles Pedido**, **Cientes**, **INVENTARIO**, and **USUARIOS**. The main content area is titled **Agregar Nuevo Detalle de Pedido** and contains the following form fields: **Pedido** (dropdown menu), **Producto** (dropdown menu), **Cantidad** (text input), **Precio Total** (text input), and **Fecha de Pedido** (date picker). At the bottom of the form are **Guardar** and **Cancelar** buttons. A **Cerrar sesión** button is located in the top right corner of the dashboard.

**Figura 90***Pantalla Actualizar Detalles de Pedidos*

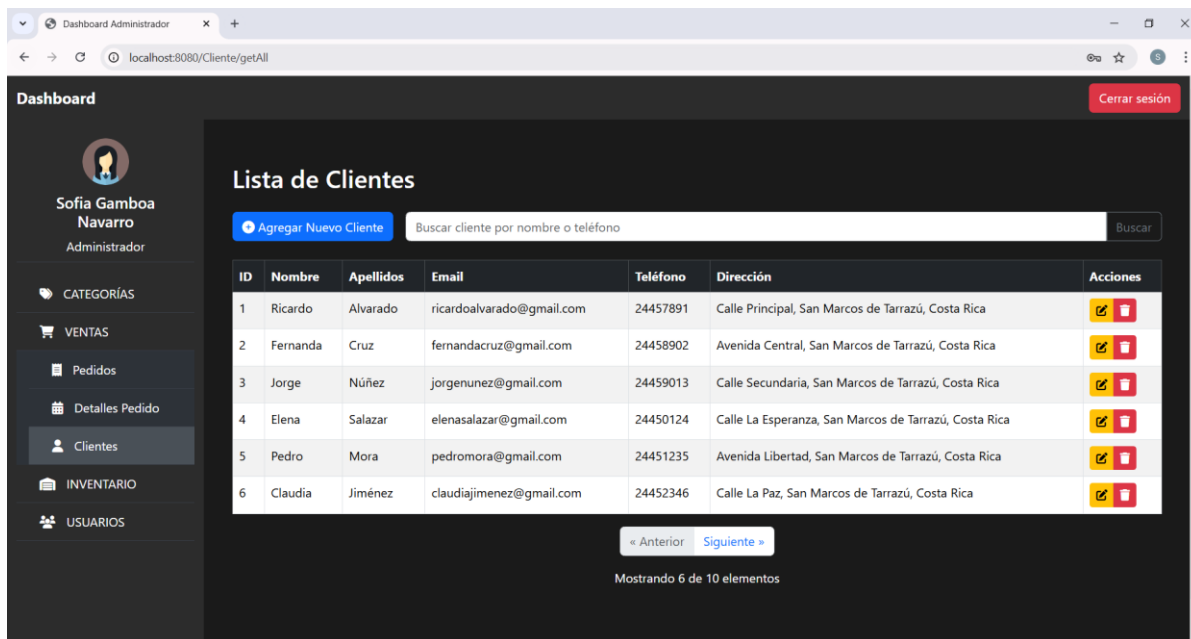
The screenshot shows a web browser window with the URL `localhost:8080/DetallePedido/update/27`. The dashboard header and sidebar menu are identical to Figure 89. The main content area is titled **Actualizar Detalle de Pedido** and contains the following form fields: **Pedido** (dropdown menu with value 17), **Producto** (dropdown menu with value Pila de niña enchapada), **Cantidad** (text input with value 3), **Precio** (text input with value 23000.00), and **Fecha de Pedido** (date picker). At the bottom of the form are **Actualizar** and **Cancelar** buttons. A **Cerrar sesión** button is located in the top right corner of the dashboard.

## Pantallas de Clientes

Pantalla en la cual se muestra el mantenimiento de Clientes.

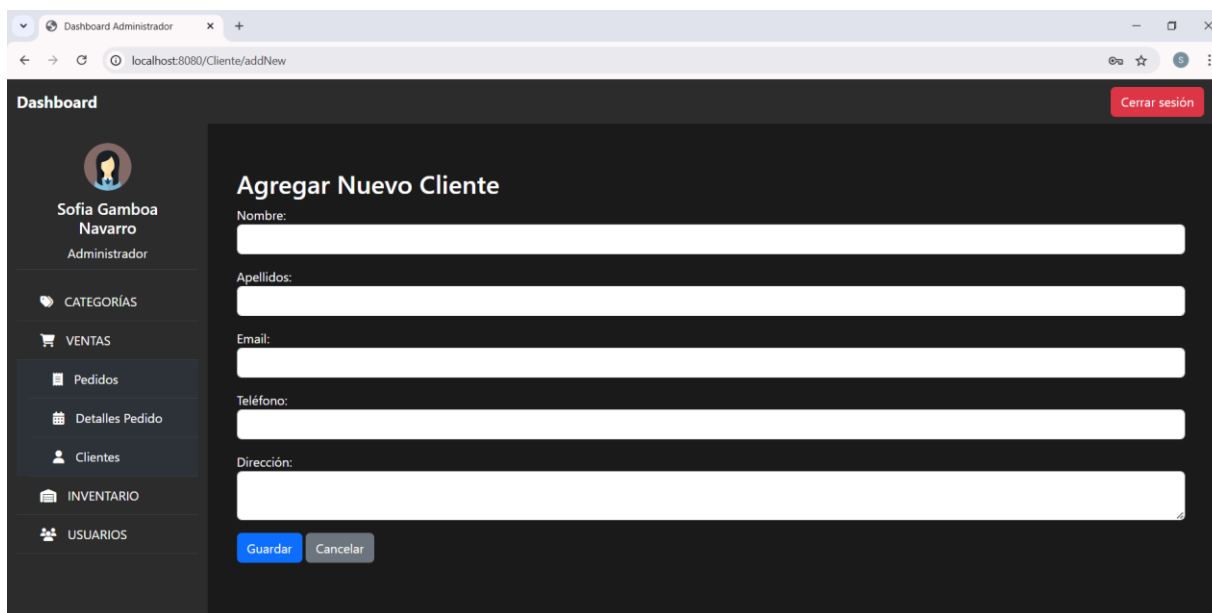
### Figura 91

#### Pantalla Listar Clientes



### Figura 92

#### Pantalla Agregar Clientes



## Figura 93

### *Pantalla Actualizar Clientes*

Dashboard Cerrar sesión

**Actualizar Cliente**

Nombre:

Apellidos:

Email:

Teléfono:

Dirección:

### *Pantallas de Productos*

Pantalla en la cual se muestra el mantenimiento de Productos.

## Figura 94

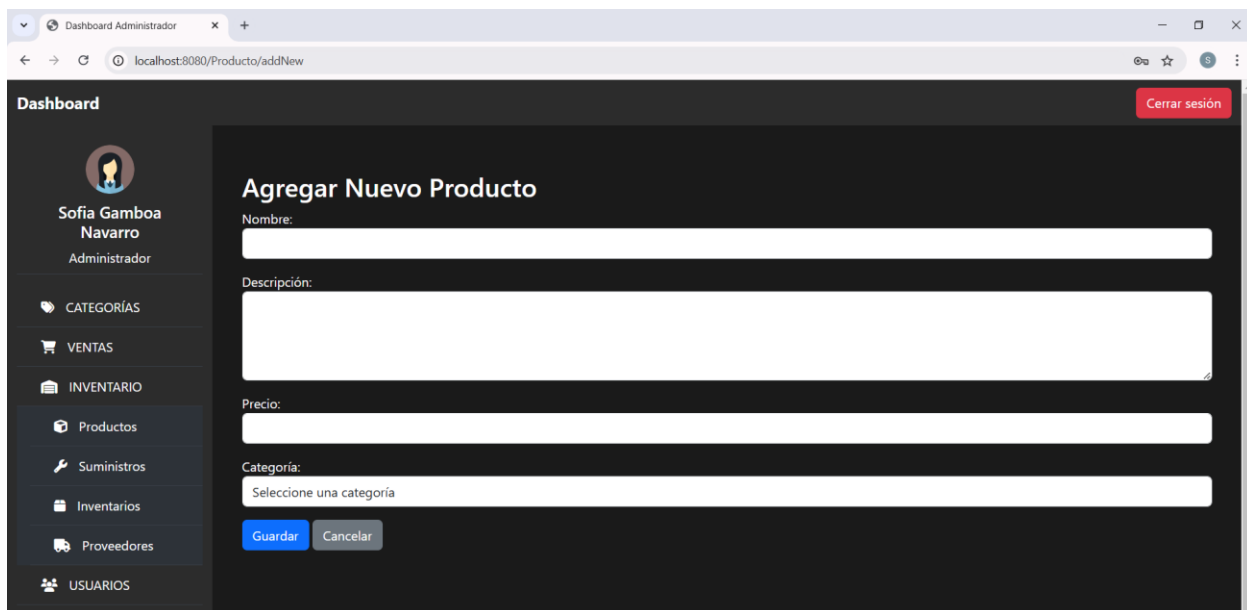
### *Pantalla Listar Productos*

Dashboard Cerrar sesión

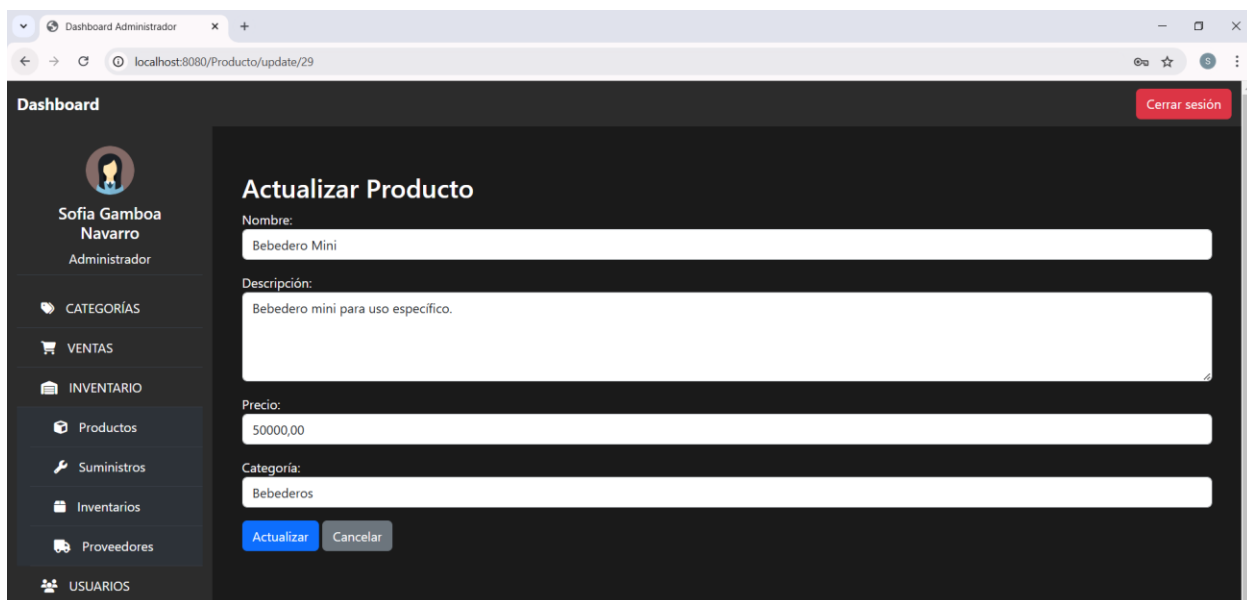
**Lista de Productos**

ID	Nombre	Descripción	Precio	Categoría	Acciones
1	Lavatorio enchapado	Lavatorio enchapado de alta calidad.	100000.00	Lavatorios	<input type="button" value="✍"/> <input type="button" value="🗑"/>
2	Lavatorio en ocre	Lavatorio en color ocre.	90000.00	Lavatorios	<input type="button" value="✍"/> <input type="button" value="🗑"/>
3	Bebedero extra grande rojo	Bebedero extra grande en color rojo.	120000.00	Bebederos	<input type="button" value="✍"/> <input type="button" value="🗑"/>
4	Bebedero extra grande enchapado	Bebedero extra grande enchapado.	130000.00	Bebederos	<input type="button" value="✍"/> <input type="button" value="🗑"/>
5	Bebedero Mini	Bebedero mini para uso específico.	50000.00	Bebederos	<input type="button" value="✍"/> <input type="button" value="🗑"/>
6	Bebedero grande enchapado	Bebedero grande enchapado de calidad.	110000.00	Bebederos	<input type="button" value="✍"/> <input type="button" value="🗑"/>

Mostrando 6 de 42 elementos

**Figura 95***Pantalla Agregar Productos*

The screenshot shows a web browser window with the URL `localhost:8080/Producto/addNew`. The page is titled "Dashboard" and features a sidebar with the user profile "Sofia Gamboa Navarro, Administrador" and a menu with items: CATEGORÍAS, VENTAS, INVENTARIO, Productos, Suministros, Inventarios, Proveedores, and USUARIOS. The main content area is titled "Agregar Nuevo Producto" and contains a form with the following fields: "Nombre:" (empty), "Descripción:" (empty), "Precio:" (empty), and "Categoría:" (dropdown menu with "Seleccione una categoría" selected). At the bottom of the form are "Guardar" and "Cancelar" buttons. A "Cerrar sesión" button is located in the top right corner of the dashboard.

**Figura 96***Pantalla Actualizar Productos*

The screenshot shows a web browser window with the URL `localhost:8080/Producto/update/29`. The page is titled "Dashboard" and features the same sidebar as Figure 95. The main content area is titled "Actualizar Producto" and contains a form with the following fields: "Nombre:" (filled with "Bebedero Mini"), "Descripción:" (filled with "Bebedero mini para uso específico."), "Precio:" (filled with "50000,00"), and "Categoría:" (dropdown menu with "Bebederos" selected). At the bottom of the form are "Actualizar" and "Cancelar" buttons. A "Cerrar sesión" button is located in the top right corner of the dashboard.

## Pantallas de Suministros

Pantalla en la cual se muestra el mantenimiento de Suministros.

### Figura 97

#### Pantalla Listar Suministros

Dashboard Administrator

localhost:8080/Suministro/getAll

Cerrar sesión

Dashboard

Sofía Gamboa Navarro  
Administrador

CATEGORÍAS

VENTAS

INVENTARIO

Productos

Suministros

Inventarios

Proveedores













USUARIOS

### Lista de Suministros

Agregar Nuevo Suministro

Buscar suministro por nombre

Buscar

ID	Nombre	Cantidad	Proveedor	Fecha	Acciones
11	Cerámica	100	Juan	2024-10-29	 
12	Pintura enchapada	50	María	2024-10-29	 
13	Pegamento	75	Luis	2024-10-29	 
14	Plástico	200	Ana	2024-10-29	 
15	Colorante rojo	30	Carlos	2024-10-29	 
16	Madera	150	Laura	2024-10-29	 

« Anterior [Siguiente](#) »

Mostrando 6 de 10 elementos

### Figura 98

#### Pantalla Agregar Suministros

Dashboard Administrator

localhost:8080/Suministro/addNew

Cerrar sesión

Dashboard

Sofía Gamboa Navarro  
Administrador

CATEGORÍAS

VENTAS

INVENTARIO

Productos

Suministros

Inventarios

Proveedores

USUARIOS

### Agregar Nuevo Suministro

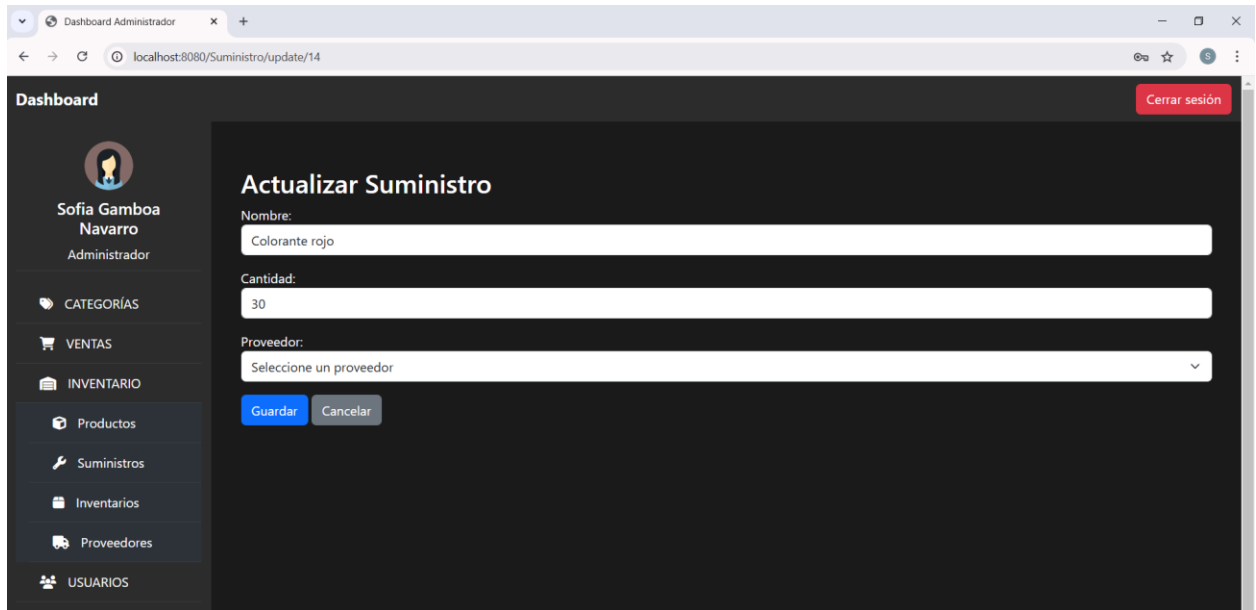
Nombre:

Cantidad:

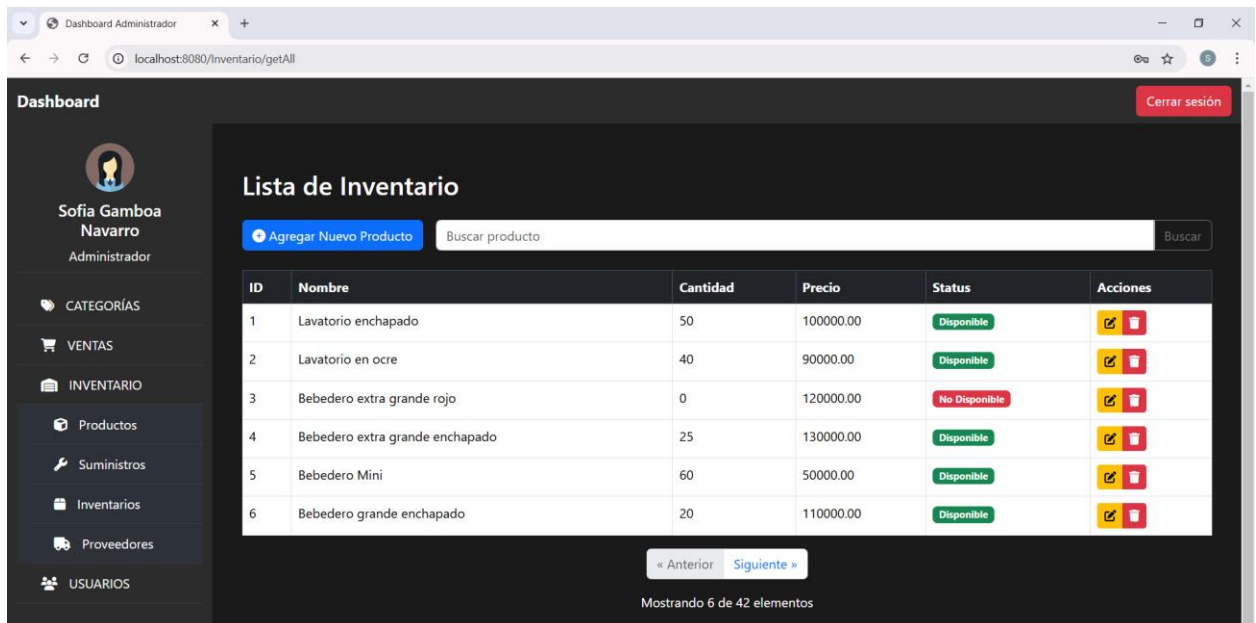
Proveedor:

Seleccione un proveedor

Guardar Cancelar

**Figura 99***Pantalla Actualizar Suministros**Pantallas de Inventario*

Pantalla en la cual se muestra el mantenimiento de Inventario.

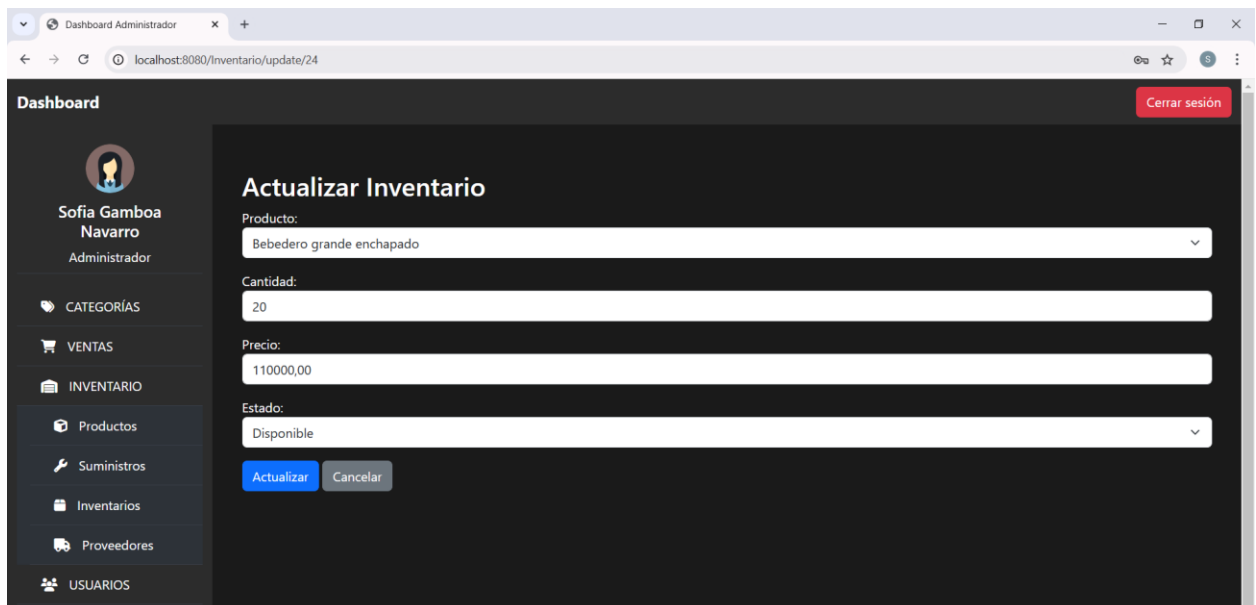
**Figura 100***Pantalla Listar Inventario*

**Figura 101***Pantalla Agregar Inventario*

The screenshot shows a web browser window with the URL `localhost:8080/inventario/addNew`. The page is titled "Agregar Nuevo Inventario" and is part of a dashboard for "Sofia Gamboa Navarro, Administrador". The dashboard sidebar includes menu items: CATEGORÍAS, VENTAS, INVENTARIO, Productos, Suministros, Inventarios, Proveedores, and USUARIOS. The main form contains the following fields:

- Producto:** A dropdown menu with the placeholder text "Seleccione un producto".
- Cantidad:** A text input field.
- Precio:** A text input field.
- Estado:** A dropdown menu with the placeholder text "Seleccione un estado".

At the bottom of the form are two buttons: "Guardar" (highlighted in blue) and "Cancelar". A "Cerrar sesión" button is located in the top right corner of the dashboard.

**Figura 102***Pantalla Actualizar Inventario*

The screenshot shows a web browser window with the URL `localhost:8080/inventario/update/24`. The page is titled "Actualizar Inventario" and is part of the same dashboard as Figure 101. The main form contains the following fields:

- Producto:** A dropdown menu with the value "Bebedero grande enchapado".
- Cantidad:** A text input field with the value "20".
- Precio:** A text input field with the value "110000,00".
- Estado:** A dropdown menu with the value "Disponible".

At the bottom of the form are two buttons: "Actualizar" (highlighted in blue) and "Cancelar". A "Cerrar sesión" button is located in the top right corner of the dashboard.

## Pantallas de Proveedores

Pantalla en la cual se muestra el mantenimiento de Proveedores.

### Figura 103

#### Pantalla Listar Proveedores

Dashboard Administrator

localhost:8080/Proveedor/getAll

Dashboard

Cerrar sesión

Sofía Gamboa Navarro  
Administrador

CATEGORÍAS

VENTAS

INVENTARIO

Productos

Suministros

Inventarios

Proveedores

USUARIOS

### Lista de Proveedores

Agregar Nuevo Proveedor

Buscar proveedor por nombre

ID	Nombre	Apellidos	Teléfono	Email	Dirección	Acciones
1	Juan	Pérez	24456789	juanperez@gmail.com	Calle San Marcos, San Marcos de Tarrazú, Costa Rica	
2	Maria	González	24457890	mariagonzalez@gmail.com	Calle Los Ángeles, San Marcos de Tarrazú, Costa Rica	
3	Luis	Rodríguez	24458901	luisrodriguez@gmail.com	Avenida Tarrazú, San Marcos de Tarrazú, Costa Rica	
4	Ana	Martinez	24459012	anamartinez@gmail.com	Calle La Loma, San Marcos de Tarrazú, Costa Rica	
5	Carlos	Sánchez	24450123	carlossanchez@gmail.com	Calle El Carmen, San Marcos de Tarrazú, Costa Rica	
6	Laura	García	24451234	lauragarcia@gmail.com	Avenida La Paz, San Marcos de Tarrazú, Costa Rica	

« Anterior    Siguiente »

Mostrando 6 de 10 elementos

### Figura 104

#### Pantalla Agregar Proveedores

Dashboard Administrator

localhost:8080/Proveedor/addNew

Dashboard

Cerrar sesión

Sofía Gamboa Navarro  
Administrador

CATEGORÍAS

VENTAS

INVENTARIO

Productos

Suministros

Inventarios

Proveedores

USUARIOS

### Agregar Nuevo Proveedor

Nombre:

Apellidos:

Teléfono:

Email:

Dirección:

Guardar    Cancelar

**Figura 105***Pantalla Actualizar Proveedores*

**Actualizar Proveedor**

Nombre:

Apellidos:

Teléfono:

Email:

Dirección:

*Pantallas de Usuarios*

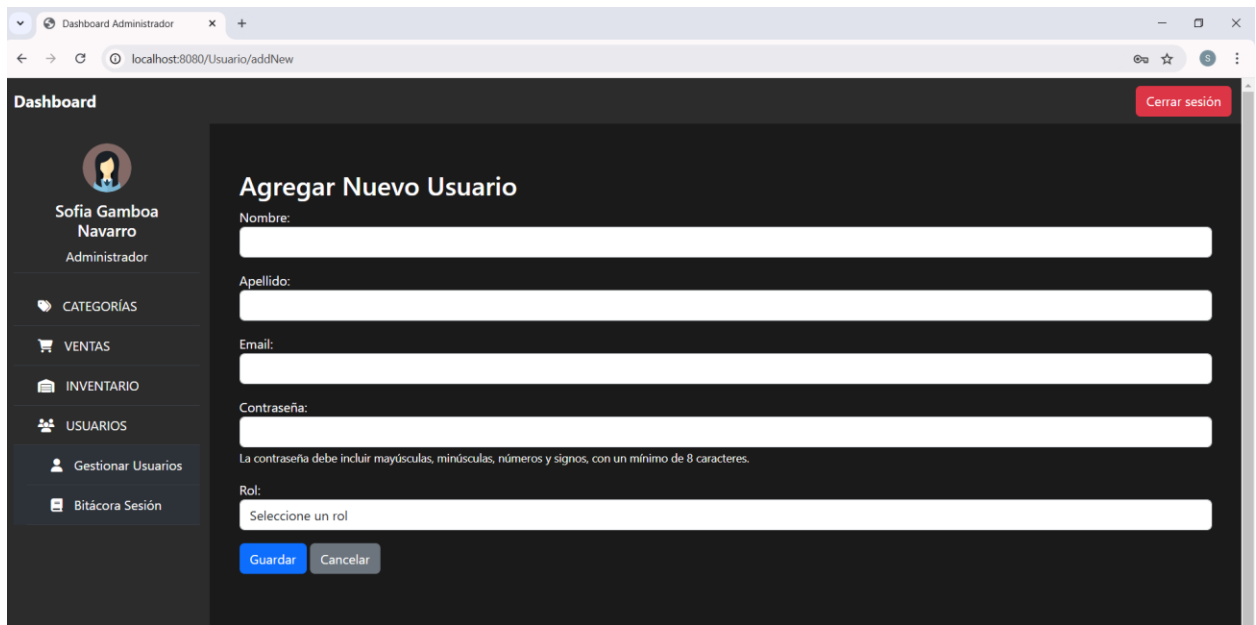
Pantalla en la cual se muestra el mantenimiento de Usuarios.

**Figura 106***Pantalla Listar Usuarios*

**Lista de Usuarios**

ID	Nombre	Apellido	Email	Rol	Acciones
1	Sofia	Gamboa Navarro	sofi_gana12@gmail.com	Admin	<input type="button" value="Editar"/> <input type="button" value="Eliminar"/>
2	Maria	Perez Osa	per_123@gmail.com	Usuario	<input type="button" value="Editar"/> <input type="button" value="Eliminar"/>
16	Maria	Romero Martines	Maroma@hotmail.com	Admin	<input type="button" value="Editar"/> <input type="button" value="Eliminar"/>
20	Sofia	Barrantes Solis	sofi_asdadad@gmail.com	Usuario	<input type="button" value="Editar"/> <input type="button" value="Eliminar"/>
29	Juliana	Blanco Barrantes	Jublaba@hotmail.com	Usuario	<input type="button" value="Editar"/> <input type="button" value="Eliminar"/>
31	Maria	Gamboa Navarro	Magana@gmail.com	Admin	<input type="button" value="Editar"/> <input type="button" value="Eliminar"/>

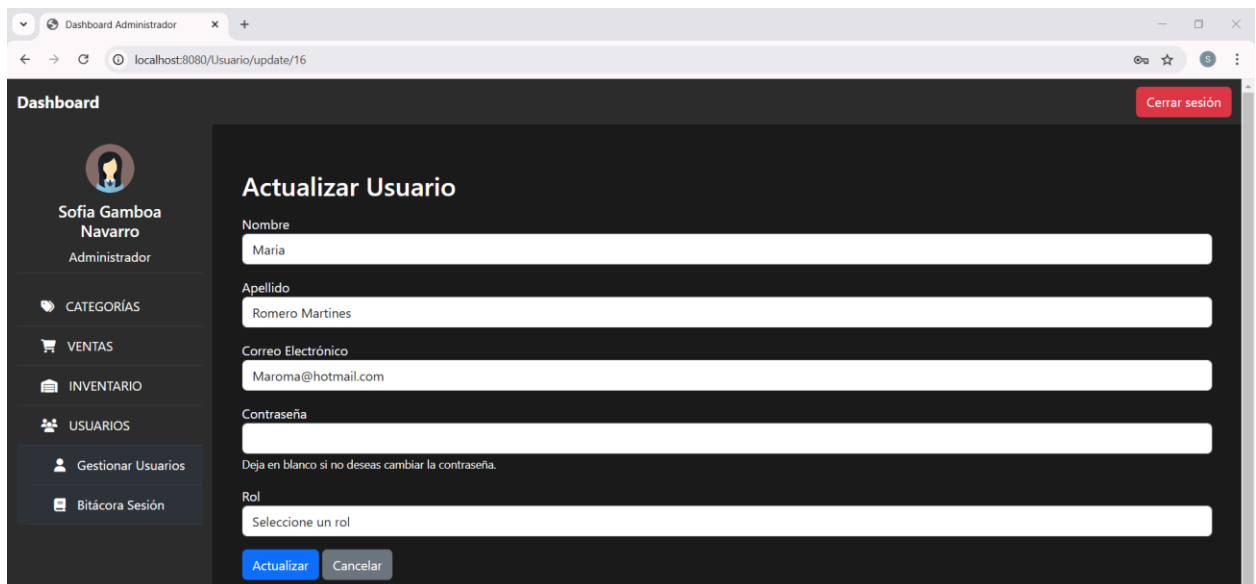
Mostrando 6 de 6 elementos

**Figura 107***Pantalla Agregar Usuarios*

The screenshot shows a web browser window with the URL `localhost:8080/Usuario/addNew`. The page is titled "Agregar Nuevo Usuario" and is part of a dashboard for "Sofia Gamboa Navarro, Administrador". The form contains the following fields:

- Nombre:
- Apellido:
- Email:
- Contraseña:   
La contraseña debe incluir mayúsculas, minúsculas, números y signos, con un mínimo de 8 caracteres.
- Rol:

At the bottom of the form are two buttons: "Guardar" (highlighted in blue) and "Cancelar". A "Cerrar sesión" button is located in the top right corner of the dashboard.

**Figura 108***Pantalla Actualizar Usuarios*

The screenshot shows a web browser window with the URL `localhost:8080/Usuario/update/16`. The page is titled "Actualizar Usuario" and is part of the same dashboard for "Sofia Gamboa Navarro, Administrador". The form contains the following fields:

- Nombre:
- Apellido:
- Correo Electrónico:
- Contraseña:   
Deja en blanco si no deseas cambiar la contraseña.
- Rol:

At the bottom of the form are two buttons: "Actualizar" (highlighted in blue) and "Cancelar". A "Cerrar sesión" button is located in the top right corner of the dashboard.

## *Pantallas de Bitácora Sesión*

Pantalla en la cual se muestra el mantenimiento de Bitácora Sesión.

### *Pantalla Vista Bitácora Sesión*

## **Figura 109**

### *Pantalla Vista Bitácora Sesión*

Dashboard

Sofia Gamboa Navarro  
Administrador

CATEGORÍAS

VENTAS

INVENTARIO

USUARIOS

Gestionar Usuarios

Bitácora Sesión

Reporte de Bitácora de Sesión

dd/mm/aaaa --:-- dd/mm/aaaa --:-- Generar Reporte

ID	Usuario	Evento	Fecha
1	Sofia	LOGIN	2024-10-29 21:20:53
2	Sofia	LOGOUT	2024-10-29 21:26:57
3	Sofia	LOGIN	2024-10-29 21:48:02
4	Sofia	LOGOUT	2024-10-29 21:49:12
5	Sofia	LOGIN	2024-10-29 21:50:49
6	Sofia	LOGOUT	2024-10-29 21:51:08

« Anterior Siguiente »

Mostrando 6 de 49 elementos

## **Figura 110**

### *Pantalla Reporte de Bitácora Sesión*

Dashboard

Sofia Gamboa Navarro  
Administrador

CATEGORÍAS

VENTAS

INVENTARIO

USUARIOS

Gestionar Usuarios

Bitácora Sesión

Reporte de Bitácora de Sesión


23/10/2024 22:53 24/10/2024 12:25 Generar Reporte


ID	Usuario	Evento	Fecha
21	Sofia	LOGIN	2024-10-23 22:53:04
22	Sofia	LOGIN	2024-10-23 22:53:57
23	Sofia	LOGOUT	2024-10-23 22:54:32
24	Sofia	LOGIN	2024-10-24 12:20:37

« Anterior Siguiente »

Mostrando 4 de 49 elementos

## Referencias Bibliográficas

 *¿Qué es la Programación orientada a objetos?* | EDteam. (s. f.). Recuperado el 20 de agosto de 2024, de <https://ed.team/comunidad/que-es-la-programacion-orientada-a-objetos>

 *Editor vs IDE* | EDteam. (s. f.). Recuperado el 26 de agosto de 2024, de <https://ed.team/comunidad/editor-vs-ide>

*3.1 Requerimientos funcionales y no funcionales.* (s. f.). Recuperado el 22 de agosto de 2024, de <https://view.genially.com/659329ab9c403800147f96f2/interactive-content-31-requerimientos-funcionales-y-no-funcionales>

*8 ejemplos de diagramas de casos de uso para inspirarse* | Boardmix. (s. f.). Recuperado el 22 de agosto de 2024, de <https://boardmix.com/es/articles/use-case-examples/>

*(25) Explorando las Similitudes entre Metodologías Ágiles y Desarrollo Tradicional.* | LinkedIn. (s. f.). Recuperado el 26 de agosto de 2024, de <https://www.linkedin.com/pulse/explorando-las-similitudes-entre-metodolog%C3%ADas-%C3%A1giles-y-perea-r/>

*¡Aprende a distinguir entre fuentes primarias, secundarias y terciarias!* (s. f.). Recuperado el 25 de septiembre de 2024, de <https://tesisdeceroa100.com/aprende-a-distinguir-entre-fuentes-primarias-secundarias-y-terciarias/>

*Arquitectura de Componentes - DEV Community.* (s. f.). Recuperado el 22 de agosto de 2024, de <https://dev.to/vanessamarely/arquitectura-de-componentes-283p>

*Arquitectura de datos: ¿qué es? Importancia, tipos y usos.* (s. f.). Recuperado el 22 de agosto de 2024, de <https://www.astera.com/es/type/blog/data-architecture/>

*Arquitectura de software: ¿Qué es y qué tipos hay?* (s. f.). Recuperado el 22 de agosto de 2024, de <https://www.gluo.mx/blog/arquitectura-de-software-que-es-y-que-tipos-hay>

*Bases de datos y su importancia en las empresas-Gestión de base de datos.* (s. f.). Recuperado el 26 de agosto de 2024, de <https://visualizamarketingonline.com.ar/blog/bases-datos-importancia-las-empresas/>

*Bootstrap Blog · Official blog for the Bootstrap framework.* (s. f.). Recuperado el 26 de agosto de 2024, de <https://blog.getbootstrap.com/>

*Bootstrap: qué es, para qué sirve y cómo usarlo | Blog de Arsys.* (s. f.). Recuperado el 26 de agosto de 2024, de <https://www.arsys.es/blog/guia-completa-sobre-bootstrap>

*Casos Uso Farmacia | PDF | Farmacia | Caso de uso.* (s. f.). Recuperado el 26 de septiembre de 2024, de <https://es.scribd.com/doc/64216512/Casos-uso-farmacia>

*¿Cómo aprender a programar desde cero? | EDteam.* (s. f.). Recuperado el 22 de agosto de 2024, de <https://ed.team/blog/como-aprender-programar-desde-cero>

*¿Cómo funciona un sistema de inventario? | Defontana Chile.* (s. f.). Recuperado el 4 de agosto de 2024, de <https://www.defontana.com/blog/cl/como-funciona-un-sistema-de-inventario>

Computacionales, S. (s. f.). *Universidad Estatal De Milagro Facultad Ciencias E Ingeniería Trabajo De Integración Curricular Previo A La Obtención Del Título De Ingeniera En.*

Comunicación, I. Y. (2023). *Propuesta de mejora en el sistema de gestión de inventarios de roles en bodega central de suministro, por medio del uso de herramientas de la calidad que permitan apoyar la reducción de costos operativos del proceso, en departamento y área seleccionada en Ingenio Taboga, año 2022.*  
<https://repositorio.ulatina.ac.cr/handle/20.500.12411/2321>

*Conceptos básicos de HTML - Aprende desarrollo web | MDN.* (s. f.). Recuperado el 26 de agosto de 2024, de [https://developer.mozilla.org/es/docs/Learn/Getting\\_started\\_with\\_the\\_web/HTML\\_basics](https://developer.mozilla.org/es/docs/Learn/Getting_started_with_the_web/HTML_basics)

*Conditionals in Thymeleaf | Baeldung.* (s. f.). Recuperado el 26 de agosto de 2024, de <https://www.baeldung.com/spring-thymeleaf-conditionals>

Corella-Parra, L. M., Olea-Miranda, J., Corella-Parra, L. M. y Olea-Miranda, J. (2023). Desarrollo de un sistema de control de inventario para una empresa comercializadora de sistemas de riego. *Ingeniería, investigación y tecnología*, 24(1), 1–10. <https://doi.org/10.22201/FI.25940732E.2023.24.1.006>

*Crear un sitio web html usando html, css y javascript.* (s. f.). Recuperado el 26 de agosto de 2024, de [https://de.fiverr.com/umar\\_daraz\\_/create-html-website-using-html-css-and-javascript](https://de.fiverr.com/umar_daraz_/create-html-website-using-html-css-and-javascript)

*Cuadro de Operacionalizacion de La Variable | PDF | Logística | Inventario.* (s. f.). Recuperado el 25 de septiembre de 2024, de <https://es.scribd.com/doc/127011876/CUADRO-DE-OPERACIONALIZACION-DE-LA-VARIABLE-xlsx>

*Curso: MySQL (Comienza gratis) | EDteam.* (s. f.). Recuperado el 26 de agosto de 2024, de <https://ed.team/cursos/mysql>

*Definición de casos de uso - Documentación de IBM.* (s. f.-a). Recuperado el 22 de agosto de 2024, de <https://www.ibm.com/docs/es/product-master/12.0.0?topic=processes-defining-use-cases>

*Definición de casos de uso - Documentación de IBM.* (s. f.-b). Recuperado el 26 de septiembre de 2024, de <https://www.ibm.com/docs/es/product-master/12.0.0?topic=processes-defining-use-cases>

*Desarrollo de un software web para la generación de planes de gestión de riesgos de software.* (s. f.). Recuperado el 27 de septiembre de 2024, de [https://www.scielo.cl/scielo.php?script=sci\\_arttext&pid=S0718-07642020000300135](https://www.scielo.cl/scielo.php?script=sci_arttext&pid=S0718-07642020000300135)

*Diagrama de casos de usos | PPT.* (s. f.). Recuperado el 26 de septiembre de 2024, de <https://es.slideshare.net/slideshow/diagrama-de-casos-de-usos/13589232#1>

*Diagrama de componentes UML: Qué es y cómo hacerlo / Miro.* (s. f.). Recuperado el 22 de agosto de 2024, de <https://miro.com/es/diagrama/que-es-diagrama-componentes-uml/>

*Diseño Integral - FasterCapital.* (s. f.). Recuperado el 22 de agosto de 2024, de <https://fastercapital.com/es/palabra-clave/dise%C3%B1o-integral.html>

*Dos siglos de historia del lenguaje de programación.* (s. f.). Recuperado el 22 de julio de 2024, de <https://www.revistaonoff.es/lenguaje-de-programacion-una-historia-con-mas-de-200-anos/>

*El ciclo SDLC en 7 fases - Viewnext.* (s. f.). Recuperado el 18 de septiembre de 2024, de <https://www.viewnext.com/el-ciclo-sdlc-en-7-fases/>

*El concepto del sistema de control de inventario, el gerente profesional y el trabajador están verificando el suministro de bienes y existencias, la gestión de inventario con la demanda de bienes 12494550 Vector en Vecteezy.* (s. f.). Recuperado el 4 de agosto de 2024, de <https://es.vecteezy.com/arte-vectorial/12494550-el-concepto-del-sistema-de-control-de-inventario-el-gerente-profesional-y-el-trabajador-estan-verificando-el-suministro-de-bienes-y-existencias-la-gestion-de-inventario-con-la-demanda-de-bienes>

*El portal de la tesis.* (s. f.). Recuperado el 24 de septiembre de 2024, de <https://recursos.ucoj.mx/tesis/investigacion.php>

En, M., De Proyectos, G., Aguilar, A. C., Alajuela, P. y Del, J. (2021). *Plan de gestión del proyecto Inventario Forestal Nacional de Costa Rica.* <https://repositoriotec.tec.ac.cr/handle/2238/13426>

*Estrategias para gestionar y priorizar requisitos de usuario en proyectos UX.* (s. f.). Recuperado el 6 de octubre de 2024, de <https://keepcoding.io/blog/requisitos-de-usuario-en-proyectos-ux/>

*Estudio de factibilidad - Qué es y qué tipos hay.* (s. f.). Recuperado el 27 de septiembre de 2024, de <https://economipedia.com/definiciones/estudio-de-factibilidad.html>

*Explorar las características del editor de código - Visual Studio (Windows) / Microsoft Learn.* (s. f.). Recuperado el 26 de agosto de 2024, de <https://learn.microsoft.com/es-es/visualstudio/ide/writing-code-in-the-code-and-text-editor?view=vs-2022>

*Fundamentos de JavaScript - Aprende desarrollo web / MDN.* (s. f.). Recuperado el 26 de agosto de 2024, de [https://developer.mozilla.org/es/docs/Learn/Getting\\_started\\_with\\_the\\_web/JavaScript\\_basics](https://developer.mozilla.org/es/docs/Learn/Getting_started_with_the_web/JavaScript_basics)

García-Pacheco, M. C. y Andrés-Laz, E. M. S. (2021). Diseño De Un Sistema De Gestión Por Procesos Para El Manejo De Inventarios. Caso: Ferretería Quiroz. *Revista Científica Multidisciplinaria Arbitrada Yachasun - ISSN: 2697-3456*, 5(9 Ed. esp.), 180–203. <https://doi.org/10.46296/YC.V5I9EDESPSOCT.0118>

*Historia de Bases de Datos: Evolución y Orígenes - jhonmosquera.com.* (s. f.). Recuperado el 26 de agosto de 2024, de <https://jhonmosquera.com/bases-de-datos-historia/>

*Historia de los lenguajes de programación - Administración de Sistemas.* (s. f.). Recuperado el 19 de julio de 2024, de <https://administraciondesistemas.com/historia-de-los-lenguajes-de-programacion/>

*IDE: ¿Qué es un entorno de desarrollo integrado?* (s. f.). Recuperado el 26 de agosto de 2024, de <https://datascientest.com/es/ide>

*IDEs: Qué son, para qué sirven y los más populares - Blog de SW Hosting.* (s. f.). Recuperado el 26 de agosto de 2024, de <https://www.swhosting.com/es/blog/ides-que-son-para-que-sirven-y-los-mas-populares>

*Introducción a la programación funcional con Java | by Johan Del Valle | Medium.* (s. f.). Recuperado el 20 de agosto de 2024, de <https://johandvv.medium.com/introducci%C3%B3n-a-la-programaci%C3%B3n-funcional-con-java-89c58e0972bc>

*Investigación cualitativa / QuestionPro.* (s. f.). Recuperado el 24 de septiembre de 2024, de <https://www.questionpro.com/es/investigacion-cualitativa.html>

*Java: ¿Qué es y para qué sirve?* (s. f.). Recuperado el 26 de agosto de 2024, de <https://www.godaddy.com/resources/latam/stories/java-que-es-y-para-que-sirve>

*Las mejores bases de datos open source - Codemotion.* (s. f.). Recuperado el 22 de agosto de 2024, de <https://www.codemotion.com/magazine/es/backend-es/las-5-mejores-bases-de-datos-open-source/>

*Lenguajes de Alto nivel vs de Bajo nivel / EDteam.* (s. f.). Recuperado el 19 de agosto de 2024, de <https://ed.team/blog/lenguajes-de-alto-nivel-vs-de-bajo-nivel-b8806b09-73ac-4d83-9047-b487dae2cab2>

*Lenguajes de bajo nivel: aprende qué y cuáles son.* (s. f.). Recuperado el 19 de agosto de 2024, de <https://keepcoding.io/blog/que-son-lenguajes-de-bajo-nivel-cuales-hay/>

*Lenguajes de programación: una breve introducción contextualizada / Alura Cursos Online.* (s. f.). Recuperado el 19 de agosto de 2024, de <https://www.aluracursos.com/blog/lenguajes-de-programacion>

*Los 17 mejores editores de código.* (s. f.). Recuperado el 26 de agosto de 2024, de <https://www.hostinger.es/tutoriales/editores-de-codigo>

*Los enfoques cuantitativo y cualitativo en la investigación / PPT.* (s. f.). Recuperado el 24 de septiembre de 2024, de <https://es.slideshare.net/slideshow/los-enfoques-cuantitativo-y-cualitativo-en-la-investigacin/30112318#6>

*Los sistemas de información de una empresa.* (s. f.). Recuperado el 18 de septiembre de 2024, de <https://informatecdigital.com/productividad/los-sistemas-de-informacion-de-una-empresa/>

*Mapa conceptual fundamentos de programación por Christopher Diego Jimenez Almaraz en Prezi.* (s. f.). Recuperado el 19 de agosto de 2024, de <https://prezi.com/bqomkgmholfj/mapa-conceptual-fundamentos-de-programacion/>

*Matriz de riesgo: Importancia y ejemplos | SafetyCulture.* (s. f.). Recuperado el 27 de septiembre de 2024, de <https://safetyculture.com/es/temas/evaluacion-de-riesgos/matriz-de-riesgo/>

*Matriz de riesgos: cómo evaluar los riesgos para lograr el éxito del proyecto [2024] • Asana.* (s. f.). Recuperado el 27 de septiembre de 2024, de <https://asana.com/es/resources/risk-matrix-template>

*Método UEPS (last in, first out): usos y ejemplos.* (s. f.). Recuperado el 4 de agosto de 2024, de <https://www.mecalux.com.mx/blog/metodo-ueps>

*Metodología de la Investigación: Marco Metodológico.* (s. f.). Recuperado el 24 de septiembre de 2024, de <https://uftmetodologia.blogspot.com/2019/09/marco-metodologico.html>

*Metodología en cascada: Ventajas e inconvenientes | SafetyCulture.* (s. f.). Recuperado el 26 de agosto de 2024, de <https://safetyculture.com/es/temas/metodologia-en-cascada/>

*Metodología Kanban: en qué consiste y cómo utilizarla | APD.* (s. f.). Recuperado el 26 de agosto de 2024, de <https://www.apd.es/metodologia-kanban/>

*Perfil y Salario de un Perfil de Desarrollador Web en México en 2024 - Hireline México.* (s. f.). Recuperado el 19 de septiembre de 2024, de <https://hireline.io/mx/enciclopedia-de-perfiles-de-tecnologia/desarrollador-web>

*Plan de pruebas de software: 8 pasos para realizarlas - Icaria Technology.* (s. f.). Recuperado el 26 de septiembre de 2024, de <https://icariatechnology.com/plan-de-pruebas-de-software/>

*Programación imperativa: ventajas y desventajas del paradigma - IONOS.* (s. f.). Recuperado el 20 de agosto de 2024, de <https://www.ionos.com/es-us/digitalguide/paginas-web/desarrollo-web/programacion-imperativa/>

*Prototipo – el papel del prototipado en el proceso de desarrollo de productos - Copymate.* (s. f.). Recuperado el 26 de agosto de 2024, de <https://copymate.app/es/blog/multi/prototipo-el-papel-del-prototipado-en-el-proceso-de-desarrollo-de-productos/>

*Publicaes.* (s. f.). Recuperado el 24 de septiembre de 2024, de [https://www.publicaes.com/?fbclid=IwY2xjawFhs5RleHRuA2FlbQIxMAABHW9ksVxMxQ0Z0RqdzjYUF9\\_XfQUV-CstGjDKRgpQiWr2oMGKDco-o\\_RNJQ\\_aem\\_\\_o-l3yxTNupy11q76mSoZw](https://www.publicaes.com/?fbclid=IwY2xjawFhs5RleHRuA2FlbQIxMAABHW9ksVxMxQ0Z0RqdzjYUF9_XfQUV-CstGjDKRgpQiWr2oMGKDco-o_RNJQ_aem__o-l3yxTNupy11q76mSoZw)

*¿Qué es CSS y para qué sirve? - Webempresa.* (s. f.). Recuperado el 26 de agosto de 2024, de <https://www.webempresa.com/blog/que-es-css-y-para-que-sirve.html>

*¿Qué es el desarrollo incremental? | phoenixNAP Glosario de TI.* (s. f.). Recuperado el 26 de agosto de 2024, de <https://phoenixnap.mx/glosario/desarrollo-incremental>

*¿Qué es frontend y backend? - Ecdisis Estudio.* (s. f.). Recuperado el 19 de septiembre de 2024, de <https://ecdisis.com/que-es-frontend-y-backend/>

*¿Qué es Interfaz Gráfica de Usuario y para qué sirve? - Neo Wiki.* (s. f.). Recuperado el 22 de agosto de 2024, de <https://neoattack.com/neowiki/interfaz-grafica-de-usuario/>

*¿Qué es la programación extrema (XP)? [2024] • Asana.* (s. f.). Recuperado el 26 de agosto de 2024, de <https://asana.com/es/resources/extreme-programming-xp>

*¿Qué es la programación, sus lenguajes y tipos de enfoque? - Tiffin University.* (s. f.-a). Recuperado el 19 de agosto de 2024, de <https://global.tiffin.edu/blog/que-es-programacion>

*¿Qué es la programación, sus lenguajes y tipos de enfoque? - Tiffin University.* (s. f.-b).

Recuperado el 19 de agosto de 2024, de <https://global.tiffin.edu/blog/que-es-programacion>

*¿Qué es MySQL? Explicación y características | Blog de Arsys.* (s. f.). Recuperado el 26 de agosto

de 2024, de <https://www.arsys.es/blog/mysql>

*¿Qué es Spring Boot y cómo se desarrollan aplicaciones? (s. f.).* Recuperado el 26 de agosto de

2024, de <https://keepcoding.io/blog/que-es-spring-boot/>

*Qué es Thymeleaf | OpenWebinars.* (s. f.). Recuperado el 26 de agosto de 2024, de

<https://openwebinars.net/blog/que-es-thymeleaf/>

*¿Qué es un lenguaje de programación de alto nivel? Y por qué saberlo.* (s. f.). Recuperado el 19

de agosto de 2024, de [https://impulso06.com/que-es-un-lenguaje-de-programacion-de-alto-](https://impulso06.com/que-es-un-lenguaje-de-programacion-de-alto-nivel-y-porque-deberias-saberlo/)

[nivel-y-porque-deberias-saberlo/](https://impulso06.com/que-es-un-lenguaje-de-programacion-de-alto-nivel-y-porque-deberias-saberlo/)

*¿Qué es una interfaz? | KeepCoding Bootcamps.* (s. f.). Recuperado el 22 de agosto de 2024, de

<https://keepcoding.io/blog/que-es-una-interfaz/>

*¿Qué es Visual Studio Code y cuáles son sus ventajas? | Blog de Arsys.* (s. f.). Recuperado el 26

de agosto de 2024, de [https://www.arsys.es/blog/que-es-visual-studio-code-y-cuales-son-sus-](https://www.arsys.es/blog/que-es-visual-studio-code-y-cuales-son-sus-ventajas)

[ventajas](https://www.arsys.es/blog/que-es-visual-studio-code-y-cuales-son-sus-ventajas)

*¿Qué son los lenguajes tipados y no tipados? (Explicación sencilla) | EDteam.* (s. f.). Recuperado

el 19 de agosto de 2024, de [https://ed.team/blog/que-son-los-lenguajes-tipados-y-no-tipados-](https://ed.team/blog/que-son-los-lenguajes-tipados-y-no-tipados-explicacion-sencilla)

[explicacion-sencilla](https://ed.team/blog/que-son-los-lenguajes-tipados-y-no-tipados-explicacion-sencilla)

*¿Qué son los paradigmas de programación? | EDteam.* (s. f.). Recuperado el 20 de agosto de 2024,

de [https://ed.team/blog/que-son-los-paradigmas-de-programacion-f57de068-6426-4332-](https://ed.team/blog/que-son-los-paradigmas-de-programacion-f57de068-6426-4332-b5dd-dbdcff1aaec)

[b5dd-dbdcff1aaec](https://ed.team/blog/que-son-los-paradigmas-de-programacion-f57de068-6426-4332-b5dd-dbdcff1aaec)

¿Qué son los paradigmas de programación? - Platzi. (s. f.). Recuperado el 22 de julio de 2024, de <https://platzi.com/clases/2211-historia-programacion/35072-que-son-los-paradigmas-de-programacion/>

Rodríguez, Á., Martínez, L., Carolina, M., López, R. y Ramón, M. (2023). *Propuesta de un sistema de gestión y control de inventarios en la bodega de almacenamiento a partir de un estudio de rotación y trazabilidad de los productos comercializados por la Distribuidora La Ruta Costa Rica durante el periodo 2021-2022*. Puntarenas, Costa Rica. <https://hdl.handle.net/20.500.13077/838>

Rueda Vera, G., Avendaño Castro, W. R. y Parada Trujillo, A. E. (2022). Sistemas de información y control de inventarios en Micro Pequeñas y Medianas Empresas - MiPyMEs de la ciudad de Cúcuta, Colombia. *Saber, Ciencia y Libertad*, 17(2), 328–351. <https://doi.org/10.18041/2382-3240/saber.2022v17n2.9295>

*Scrum: conceptos clave y cómo se aplica en la gestión de proyectos [2024]* • Asana. (s. f.). Recuperado el 26 de agosto de 2024, de <https://asana.com/es/resources/what-is-scrum>

Silva Polo, C. R. y Ramírez García, V. J. (2021). Desarrollo de un sistema de gestión de activos para mejorar su control en el departamento de Tecnología de Información de la empresa Industrias Plásticas Reunidas S. A. C. *Repositorio Institucional – UCS*. <https://doi.org/10.21142/TB.2020.2126>

*Sistema de inventario: Toma el control de tu almacén (2023) | Colombia*. (s. f.). Recuperado el 4 de agosto de 2024, de <https://blog.alegra.com/colombia/sistema-de-inventarios/>

*Sistema de inventarios: qué es y cuáles son sus tipos*. (s. f.). Recuperado el 4 de agosto de 2024, de <https://www.doofinder.com/es/blog/sistema-de-inventarios>

Sistemas Computacionales, I. E., Milagro, E. DE e Izquierdo Albarracín Miguel Rolando Sr Shigla

Morocho Alex Paul, S. (2021). Desarrollo de un Sistema WEB para el Control y Gestión de

Reactivos e Insumos de los Laboratorios de Ingeniería de la Universidad Estatal de Milagro.

*Repositorio de la Universidad Estatal de Milagro.*

<http://repositorio.unemi.edu.ec/xmlui/handle/123456789/6025>

*Sistemas de información en empresas: definición, tipos y ejemplos.* (s. f.). Recuperado el 18 de

septiembre de 2024, de <https://blog.hubspot.es/marketing/sistema-informacion#que>

*Spring-boot-logo | yes.* (s. f.). Recuperado el 26 de agosto de 2024, de

<https://www.yessinfotech.com/spring-boot/spring-boot-logo/>

*Tipos de lenguajes de programación | EDteam.* (s. f.). Recuperado el 19 de agosto de 2024, de

<https://ed.team/blog/tipos-de-lenguajes-de-programacion>

*Tipos y clasificación de inventarios existentes.* (s. f.). Recuperado el 14 de agosto de 2024, de

<https://www.xamai.com/blog/clasificacion-de-inventarios>

*Tipos y niveles de lenguajes de programación ejemplos 2024 | AAU.* (s. f.). Recuperado el 19 de

agosto de 2024, de <https://www.auniv.com/s/blog/lenguajes-de-programacion/>

*Un poco de historia sobre las bases de datos - ADR Formación.* (s. f.). Recuperado el 26 de agosto

de 2024, de [https://www.adrformacion.com/knowledge/programacion/un\\_poco\\_de\\_h](https://www.adrformacion.com/knowledge/programacion/un_poco_de_historia_sobre_las_bases_de_datos.html)

[istoria\\_sobre\\_las\\_bases\\_de\\_datos.html](https://www.adrformacion.com/knowledge/programacion/un_poco_de_historia_sobre_las_bases_de_datos.html)

*Visual Studio Code - Code Editing. Redefined.* (s. f.). Recuperado el 26 de agosto de 2024,

de <https://code.visualstudio.com/>