



Universidad
Nacional
de Loja

Universidad Nacional de Loja

Facultad de la Energía, las Industrias y los Recursos
Naturales no Renovables

Carrera de Ingeniería en Sistemas

Desarrollo de una aplicación web progresiva para la
gestión y comercialización de productos y servicios para la
empresa “Compumars” de la ciudad de Loja

Trabajo de Titulación previa a la
obtención del título de Ingeniero en
Sistemas

AUTOR:

Víctor Andrés Rojas Curimilma

DIRECTOR:

Ing. Wilman Patricio Chamba Zaragocin, Mg.Sc.

Loja – Ecuador

2023

Certificación

Loja, 09 de septiembre de 2022

Ing. Wilman Chamba Zaragocín.
DIRECTOR DE TRABAJO DE TITULACIÓN

CERTIFICO

Que he revisado y orientado todo el proceso de elaboración del Trabajo de Titulación denominado: **“Desarrollo de una Aplicación Web Progresiva para la gestión y comercialización de productos y servicios para la Empresa “Compumars” de la ciudad de Loja”**, previo a la obtención del título de **Ingeniero en Sistemas**, de la autoría del estudiante **Víctor Andrés Rojas Curimilma**, con **cédula de identidad Nro.1105834939**, una vez que el trabajo cumple con todos los requisitos exigidos por la Universidad Nacional de Loja, para el efecto, autorizo la presentación del mismo para su respectiva sustentación y defensa.

Ing. Wilman Chamba Zaragocín.
DIRECTOR DE TRABAJO DE TITULACIÓN

Autoría

Yo, **Víctor Andrés Rojas Curimilma**, declaro ser autor del presente Trabajo de Titulación y eximo expresamente a la Universidad Nacional de Loja y a sus representantes jurídicos de posibles reclamos y acciones legales por el contenido del mismo. Adicionalmente acepto y autorizo a la Universidad Nacional de Loja la publicación de mí Trabajo de Titulación en el Repositorio Digital Institucional – Biblioteca Virtual.

Firma:

Cédula: 1105834939

Fecha: 6 de marzo de 2023

Correo Electrónico: victor.rojas@unl.edu.ec

Celular: (+593) 96 9328 489

Carta de autorización por parte del autor, para consulta, reproducción parcial o total y/o publicación electrónica del texto completo del Trabajo de Titulación

Yo **Víctor Andrés Rojas Curimilma**, declaro ser autor del Trabajo de Titulación denominado: **Desarrollo de una Aplicación Web Progresiva para la gestión y comercialización de productos y servicios para la Empresa “Compumars” de la ciudad de Loja** como requisito para optar el título de **Ingeniero en Sistemas**, autorizo al sistema Bibliotecario de la Universidad Nacional de Loja para que con fines académicos muestre la producción intelectual de la Universidad, a través de la visibilidad de su contenido de la siguiente manera en el Repositorio Institucional.

Los usuarios pueden consultar el contenido de este trabajo en el Repositorio Institucional, en las redes de información del país y del exterior, con las cuales tenga convenio la Universidad.

La Universidad Nacional de Loja, no se responsabiliza por el plagio o copia del Trabajo de Titulación que realice un tercero.

Para constancia de esta autorización, en la ciudad de Loja, a los seis días del mes de marzo de dos mil veintitrés.

Firma:

Autor: Víctor Andrés Rojas Curimilma

Cédula: 1105834939

Dirección: Av. Eugenio Espejo y Adolfo Valarezo, Loja - Ecuador

Correo Electrónico: victor.rojas@unl.edu.ec

Celular: (+593) 96 9328 489

DATOS COMPLEMENTARIOS:

Director del Trabajo de Titulación: Ing. Wilman Patricio Chamba Zaragocín, Mg.Sc

Dedicatoria

A ti, mamá, en tu ausencia, todo esto sería imposible.

Víctor Rojas

Agradecimiento

Agradezco infinitamente a Dios por permitirme vivir este momento.

A mi familia, por ser mi refugio, mi soporte, mi fortaleza. A mis amigos, amigas, compañeros y compañeras que me brindaron, de algún modo u otro, su apoyo y compañía.

A todas esas personas tan cercanas a mí que estuvieron cuando más lo necesitaba. Quiero también extender mi más sincero agradecimiento al Ing. Wilman Chamba, por la confianza, por brindarme sus conocimientos y su gran apoyo en todo este proceso de investigación.

Agradecer también a todos los docentes por ayudarme a crecer día a día. y, cómo no, dar las gracias a la Universidad Nacional de Loja por tan bonita experiencia que me permitió vivir a lo largo de todos estos años.

Víctor Rojas

Índice de Contenidos

Portada	i
Certificación	ii
Autoría	iii
Carta de autorización	iv
Dedicatoria	v
Agradecimiento	vi
Índice de Contenidos	vii
Índice de tablas.....	ix
Índice de figuras.....	x
Índice de anexos.....	xv
1. Título	1
2. Resumen	2
2.1. Abstract.....	3
3. Introducción	4
4. Marco Teórico	6
4.1. Antecedentes.....	6
4.1.1. Comercio electrónico.....	6
4.1.2. Técnicas de manejo de inventario.....	7
4.2. Ingeniería de Software.....	8
4.2.1. Metodología ICONIX.....	8
4.2.2. Aplicaciones Web Progresivas.....	11
4.3. Tecnologías.....	12
4.3.1. JavaScript.....	12
4.3.2. MongoDB.....	14
4.4. Trabajos relacionados.....	15
4.4.1. Desarrollo de una aplicación WEB para pedidos en línea en la casa comercial Calva & Calva.....	15
4.4.2. Diseño e implementación de una tienda virtual para la venta de equipos informáticos para la empresa "SETCOMPC" de la ciudad de Loja.....	16
4.4.3. Creación de una tienda virtual para compras de abarrotes en línea utilizando metodología de desarrollo SCRUM.....	17
4.4.4. Desarrollo de un software para la creación de tiendas virtuales orientadas a b2c que facilite el ingreso de pequeñas y medianas empresas ecuatorianas al comercio electrónico.....	17
5. Metodología	18
5.1. Área de estudio.....	18
5.2. Procedimiento.....	18

5.2.1.	Construir el modelo de arquitectura física y lógica en base a los requerimientos del usuario utilizando el Modelamiento de desarrollo de software dirigido por ICONIX.	18
5.2.2.	Implementar la arquitectura de la solución informática en una Aplicación Web Progresiva (PWA) mediante el uso de tecnologías JavaScript y la metodología ICONIX.	19
5.2.3.	Comparar los procesos de gestión y comercialización tradicional y web en un ambiente de pruebas simulado para determinar los beneficios de la actualización.	19
5.3.	Recursos.....	19
5.3.1.	Recursos Científicos.....	20
5.3.2.	Recursos de Hardware y Software	20
5.3.3.	Recursos Técnicos	20
6.	Resultados	22
6.1.	Objetivo 1: Construir el modelo de arquitectura física y lógica en base a los requerimientos del usuario utilizando el Modelamiento de desarrollo de software dirigido por ICONIX.....	22
6.1.1.	Análisis y diseño preliminar.	22
6.1.2.	Diseño de la arquitectura física y lógica.	27
6.2.	Objetivo 2: Implementar la arquitectura de la solución informática en una Aplicación Web Progresiva (PWA) mediante el uso de tecnologías JavaScript y la metodología ICONIX.	48
6.2.1.	Implementación.	48
6.2.2.	Realizar pruebas de funcionamiento del software.....	63
6.3.	Objetivo 3: Comparar los procesos de gestión y comercialización tradicional y web en un ambiente de pruebas simulado para determinar los beneficios de la actualización. . .	65
6.3.1.	Planificación del caso de prueba para la simulación manual versus web.	65
6.3.2.	Ejecución de la simulación.	68
6.3.3.	Establecer un análisis cuantitativo de los resultados.	74
7.	Discusión	77
7.1.	Objetivo 1: Construir el modelo de arquitectura física y lógica en base a los requerimientos del usuario utilizando el Modelamiento de desarrollo de software dirigido por ICONIX.....	77
7.2.	Objetivo 2: Implementar la arquitectura de la solución informática en una Aplicación Web Progresiva (PWA) mediante el uso de tecnologías JavaScript y la metodología ICONIX.	78
7.3.	Objetivo 3: Comparar los procesos de gestión y comercialización tradicional y web en un ambiente de pruebas simulado para determinar los beneficios de la actualización. . .	79
8.	Conclusiones.....	81
9.	Recomendaciones.....	82

9.1. Trabajos Futuros.....	82
10. Bibliografía.....	84
11. Anexos	88

Índice de tablas:

Tabla 1. Preguntas relevantes de la entrevista para la obtención de requerimientos	22
Tabla 2. Glosario de términos.....	23
Tabla 3. Requerimientos funcionales más relevantes	24
Tabla 4. Requerimientos no funcionales.....	25
Tabla 5. Especificación del caso de uso Iniciar sesión.....	31
Tabla 6. Especificación del caso de uso Registrar cuentas de usuario.....	31
Tabla 7. Especificación de caso de uso Realizar órdenes	32
Tabla 8. Especificación de caso de uso Registrar órdenes de venta	33
Tabla 9. Especificación de caso de uso Registrar pago para la orden de venta	34
Tabla 10. Especificación de caso de uso Validar pagos de órdenes.....	35
Tabla 11. Especificación de caso de uso Registrar productos	36
Tabla 12. Especificación de caso de uso Registrar órdenes de compra	37
Tabla 13. Módulos de software a implementar	48
Tabla 14. Comité del proyecto	65
Tabla 15. Cronograma de actividades	66
Tabla 16. Mitigación de riesgos	67
Tabla 17. Lista de preguntas de la encuesta de satisfacción	71
Tabla 18. Escenarios de prueba para el proceso tradicional.....	72
Tabla 19. Escenarios de prueba para el proceso web.....	72
Tabla 20. Modelo de caso de prueba para el proceso tradicional	73
Tabla 21. Modelo de caso de prueba para el proceso web	73
Tabla 22. Resumen de resultados de la Encuesta de Satisfacción.....	74
Tabla 23. Requerimientos funcionales de Todos los usuarios	94
Tabla 24. Requerimientos funcionales del Administrador.....	95
Tabla 25. Requerimientos funcionales del Cliente.....	95
Tabla 26. Requerimientos funcionales del Cajero	96
Tabla 27. Requerimientos funcionales del Bodeguero	96
Tabla 28. Requerimientos funcionales del Operador.....	97
Tabla 29. Requerimientos no funcionales.....	97
Tabla 30. Especificación del caso de uso Iniciar Sesión	102
Tabla 31. Especificación del caso de uso Buscar productos por categoría.....	103
Tabla 32. Especificación del caso de uso Buscar cuentas de usuario	104
Tabla 33. Especificación del caso de uso Registrar cuentas de usuario.....	105
Tabla 34. Especificación del caso de uso Modificar cuentas de usuario.....	105
Tabla 35. Especificación del caso de uso Dar de baja cuentas de usuario.....	106
Tabla 36. Especificación del caso de uso Realizar órdenes.....	108
Tabla 37. Especificación del caso de uso Buscar órdenes de venta	108
Tabla 38. Especificación del caso de uso Registrar órdenes de venta	109
Tabla 39. Especificación del caso de uso Completar órdenes	110
Tabla 40. Especificación del caso de uso Registrar pago para la orden de venta	111
Tabla 41. Especificación del caso de uso Visualizar pagos de órdenes	112
Tabla 42. Especificación del caso de uso Validar pagos de órdenes.....	112
Tabla 43. Especificación del caso de uso Buscar productos.....	114

Tabla 44. Especificación del caso de uso Registrar productos	115
Tabla 45. Especificación del caso de uso Modificar productos	115
Tabla 46. Especificación del caso de uso Dar de baja productos	116
Tabla 47. Especificación del caso de uso Buscar órdenes de compra.....	117
Tabla 48. Especificación del caso de uso Registrar órdenes de compra	118
Tabla 49. Participantes y resultados de la Encuesta de Satisfacción.....	172
Tabla 50. Resumen de resultados de la Encuesta de Satisfacción.....	180
Tabla 51. Componentes de la página de direcciones del usuario	194
Tabla 52. Componentes de la página de ordenes realizadas por el usuario.....	195
Tabla 53. Componentes de la página de ordenes realizadas por el usuario.....	197
Tabla 54. Componentes de la página de configuraciones generales del sistema.....	199
Tabla 55. Componentes de la página de ordenes realizadas por el usuario.....	200
Tabla 56. Componentes de la página de ordenes realizadas por el usuario.....	201
Tabla 57. Componentes de la página de ver detalles de un producto.....	202
Tabla 58. Componentes del menú lateral del carrito de compras	203
Tabla 59. Componentes de la página de carrito de compras	204
Tabla 60. Componentes de la página de ver detalles de una orden	205
Tabla 61. Componentes de la página de inventario de productos.....	206
Tabla 62. Componentes de la página de detalles del inventario	207
Tabla 63. Componentes de la página de órdenes de venta	207
Tabla 64. Componentes de la página de órdenes de venta	208
Tabla 65. Componentes del formulario de nueva orden de venta – Paso 1	209
Tabla 66. Componentes del formulario de nueva orden de venta – Paso 2.....	210
Tabla 67. Componentes de la página de proveedores.....	213
Tabla 68. Componentes de la página de categorías	215
Tabla 69. Componentes de la página de marcas	216
Tabla 70. Componentes de la página de productos	218
Tabla 71. Componentes de la página de órdenes de compra	220
Tabla 72. Componentes del formulario de nueva orden de compra – Paso 2.....	221
Tabla 73. Componentes de la página de impuestos.....	223

Índice de figuras:

Figura 1. Flujo de trabajo de la Metodología ICONIX.....	9
Figura 2. Diagrama del dominio de Usuarios	26
Figura 3. Diagrama del dominio de Órdenes realizadas por el cliente.....	26
Figura 4. Diagrama del dominio de Inventario.....	27
Figura 5. Diagrama de casos de uso General.....	28
Figura 6. Diagrama de casos de uso de Administrar cuentas de usuario	28
Figura 7. Diagrama de casos de uso de Administrar órdenes de venta.....	29
Figura 8. Diagrama de casos de uso de Administrar pagos de órdenes.....	29
Figura 9. Diagrama de casos de uso de Administrar productos.....	30
Figura 10. Diagrama de casos de uso de Administrar órdenes de compra	30
Figura 11. <i>Wireframe</i> de Realizar orden	32
Figura 12. <i>Wireframe</i> de Registrar órdenes de venta	33
Figura 13. <i>Wireframe</i> de Validar pagos de órdenes.....	35
Figura 14. <i>Wireframe</i> de Registrar órdenes de compra	37
Figura 15. Diagrama de secuencia de Iniciar sesión.....	38
Figura 16. Diagrama de secuencia de Registrar cuentas de usuario.....	39

Figura 17. Realizar orden	39
Figura 18. Registrar órdenes de venta	40
Figura 19. Registrar pago para la orden de venta	40
Figura 20. Validar pagos de órdenes	41
Figura 21. Registrar Productos.....	41
Figura 22. Registrar órdenes de compra	42
Figura 23. Diagrama de paquetes	43
Figura 24. Diagrama de Clases.....	44
Figura 25. Diagrama de componentes	45
Figura 26. Diagrama de despliegue	46
Figura 27. Estructura general de carpetas del proyecto.....	49
Figura 28. Estructura de directorios del Cliente	50
Figura 29. Middleware de autenticación en el cliente.....	50
Figura 30. Código de configuración general del cliente y la PWA.....	51
Figura 31. Estructura de directorios del Servidor	52
Figura 32. Función para el envío de correos electrónicos desde el servidor	52
Figura 33. Funcionalidad para generar y comparar un hash de una contraseña.....	53
Figura 34. Estructura de archivos de manifiesto para la infraestructura de la aplicación	53
Figura 35. Configuración del acceso externo a los servicios de la aplicación.....	54
Figura 36. Manifiesto con la configuración de un <i>deployment</i>	55
Figura 37. Manifiesto con la configuración de <i>redis</i>	55
Figura 38. Página de Listar Usuarios de Compumars.....	56
Figura 39. Página de Listar Proveedores	57
Figura 40. Página de Listar Direcciones del Usuario	57
Figura 41. Página de Listar Productos de Compumars.....	58
Figura 42. Página de Listar Categorías.....	58
Figura 43. Página de Listar Marcas.....	59
Figura 44. Página de Listar Servicios.....	59
Figura 45. Página del Inventario de Compumars	60
Figura 46. Página de Kardex de un producto.....	60
Figura 47. Página de Listar Órdenes de Compra.....	61
Figura 48. Página de Listar Órdenes de Venta	61
Figura 49. Página de Carrito de Compras.....	62
Figura 50. Página de Listar órdenes realizadas por un usuario	62
Figura 51. Página de registrar constancia del pago de una orden de venta	63
Figura 52. Estructura del directorio de pruebas unitarias.....	63
Figura 53. Estructura del directorio de pruebas de integración.....	64
Figura 54. Estructura del directorio para la configuración de pruebas con <i>Jest</i>	64
Figura 55. Resultado de las pruebas unitarias y de integración de la Aplicación Web Progresiva.....	64
Figura 56. Resultado de las pruebas unitarias y de integración de la Aplicación Web Progresiva.....	69
Figura 57. Alerta para la instalación de la PWA en ordenadores.....	70
Figura 58. Alerta para la instalación de la PWA en dispositivos móviles	70
Figura 59. Resumen de resultados de la Encuesta de Satisfacción	75
Figura 60. Tiempo empleado al Registrar productos en el inventario (Tradicional vs Web)	75
Figura 61. Tiempo empleado al Realizar una nueva venta (Tradicional vs Web).....	76
Figura 62. Tiempo empleado al Registrar y validar pago de una venta Tradicional vs Web	76
Figura 63. Diagrama del dominio de Usuarios	98
Figura 64. Diagrama del dominio de Orden	98
Figura 65. Diagrama del dominio de Inventario.....	98

Figura 66. Diagrama de casos de uso General	99
Figura 67. Diagrama de casos de uso de Administrar cuentas de usuario	100
Figura 68. Diagrama de casos de uso de Administrar órdenes de venta	100
Figura 69. Diagrama de casos de uso de Administrar pagos de órdenes	101
Figura 70. Diagrama de casos de uso de Administrar productos	101
Figura 71. Diagrama de casos de uso de Administrar órdenes de compra	102
Figura 72. <i>Wireframe</i> de Buscar productos por categoría	103
Figura 73. <i>Wireframe</i> de Administrar cuentas de usuario	104
Figura 74. <i>Wireframe</i> de Administrar órdenes de venta	107
Figura 75. <i>Wireframe</i> de Realizar órdenes	107
Figura 76. <i>Wireframe</i> de Registrar órdenes de venta	109
Figura 77. <i>Wireframe</i> de Administrar pagos de órdenes	111
Figura 78. <i>Wireframe</i> de Administrar productos	113
Figura 79. <i>Wireframe</i> de Inventario	114
Figura 80. <i>Wireframe</i> de Administrar órdenes de compra	117
Figura 81. <i>Wireframe</i> de Registrar órdenes de compra	118
Figura 82. Diagrama de secuencia de Iniciar sesión	119
Figura 83. Diagrama de secuencia de Buscar productos por categoría	120
Figura 84. Diagrama de secuencia de Buscar cuentas de usuario	120
Figura 85. Diagrama de secuencia de Registrar cuentas de usuario	121
Figura 86. Diagrama de secuencia de Modificar cuentas de usuario	121
Figura 87. Diagrama de secuencia de Dar de baja cuentas de usuario	122
Figura 88. Diagrama de secuencia de Realizar órdenes	122
Figura 89. Diagrama de secuencia de Buscar órdenes de venta	123
Figura 90. Diagrama de secuencia de Registrar órdenes de venta	123
Figura 91. Diagrama de secuencia de Completar órdenes	124
Figura 92. Diagrama de secuencia de Registrar pago para la orden de venta	124
Figura 93. Diagrama de secuencia de Visualizar pagos de órdenes	125
Figura 94. Diagrama de secuencia de Validar pagos de órdenes	125
Figura 95. Diagrama de secuencia de Buscar productos	126
Figura 96. Diagrama de secuencia de Registrar productos	126
Figura 97. Diagrama de secuencia de Modificar productos	127
Figura 98. Diagrama de secuencia de Dar de baja productos	127
Figura 99. Diagrama de secuencia de Buscar órdenes de compra	128
Figura 100. Diagrama de secuencia de Registrar órdenes de compra	128
Figura 101. Diagrama de paquetes	129
Figura 102. Diagrama de clases	130
Figura 103. Diagrama de componentes	131
Figura 104. Diagrama de despliegue	131
Figura 105. Interfaz gráfica de la Página principal	132
Figura 106. Interfaz gráfica de Ingreso al sistema	132
Figura 107. Interfaz gráfica de Registro en el sistema	133
Figura 108. Interfaz gráfica de Recuperar contraseña	133
Figura 109. Interfaz gráfica de Buscar Productos	134
Figura 110. Interfaz gráfica de Buscar productos por categorías	134
Figura 111. Interfaz gráfica de Ver producto	135
Figura 112. Interfaz gráfica de Perfil de usuario	135
Figura 113. Interfaz gráfica de Listar direcciones del cliente	136
Figura 114. Interfaz gráfica de Nueva dirección	136
Figura 115. Interfaz gráfica de Ver Configuraciones del sistema	137
Figura 116. Interfaz gráfica de Modificar parámetro de configuración	137

Figura 117. Interfaz gráfica de Listar usuarios	138
Figura 118. Interfaz gráfica de Nuevo usuario	138
Figura 119. Interfaz gráfica de Modificar usuario	139
Figura 120. Interfaz gráfica de Dar de baja usuario	139
Figura 121. Interfaz gráfica de Listar proveedores	140
Figura 122. Interfaz gráfica de Nuevo proveedor	140
Figura 123. Interfaz gráfica de Modificar proveedor	141
Figura 124. Interfaz gráfica de Listar productos	141
Figura 125. Interfaz gráfica de Nuevo producto	142
Figura 126. Interfaz gráfica de Modificar producto	142
Figura 127. Interfaz gráfica de Dar de baja producto	143
Figura 128. Interfaz gráfica de Listar categorías	143
Figura 129. Interfaz gráfica de Nueva categoría	144
Figura 130. Interfaz gráfica de Modificar categoría	144
Figura 131. Interfaz gráfica de Listar marcas	145
Figura 132. Interfaz gráfica de Nueva marca	145
Figura 133. Interfaz gráfica de Modificar marca	146
Figura 134. Interfaz gráfica de Filtrar listado de servicios	146
Figura 135. Interfaz gráfica de Ver Inventario	147
Figura 136. Interfaz gráfica de Detalles de un producto del inventario	147
Figura 137. Interfaz gráfica de Listar órdenes de compra	148
Figura 138. Interfaz gráfica de Detalles de una orden de compra	148
Figura 139. Interfaz gráfica de Nueva orden de compra Etapa de seleccionar proveedor .	149
Figura 140. Interfaz gráfica de Nueva orden de compra (Etapa de seleccionar productos)	149
Figura 141. Interfaz gráfica de Nueva orden de compra (Etapa de registro del comprobante de pago)	150
Figura 142. Interfaz gráfica del Carrito de compras	150
Figura 143. Interfaz gráfica de órdenes realizadas por el cliente	151
Figura 144. Interfaz gráfica de Ver detalles de una orden	151
Figura 145. Interfaz gráfica de Registrar constancia del pago de una orden	152
Figura 146. Interfaz gráfica de Listar órdenes de venta	152
Figura 147. Interfaz gráfica de Detalles de una orden de venta	153
Figura 148. Interfaz gráfica de Nueva orden de venta (Etapa de seleccionar productos) ..	153
Figura 149. Interfaz gráfica de Nueva orden de venta (Etapa de seleccionar cliente)	154
Figura 150. Interfaz gráfica de Nueva orden de venta (Etapa de resumen)	154
Figura 151. Interfaz gráfica de Listar impuestos	155
Figura 152. Interfaz gráfica de Nuevo impuesto	155
Figura 153. Interfaz gráfica de Modificar impuesto	156
Figura 154. Pruebas unitarias para la validación de Cédula/RUC	163
Figura 155. Pruebas unitarias para la encriptación y validación de contraseñas	163
Figura 156. Pruebas de integración de Iniciar Sesión	164
Figura 157. Pruebas de integración de listar usuarios	164
Figura 158. Pruebas de integración de Registrar usuarios	164
Figura 159. Prueba de integración de Modificar rol de un usuario	165
Figura 160. Prueba de integración de Dar de baja un usuario	165
Figura 161. Prueba de integración de Listar productos	165
Figura 162. Pruebas de integración de Registrar productos	166
Figura 163. Prueba de integración de Modificar producto	166
Figura 164. Prueba de integración de Dar de baja producto	167
Figura 165. Pruebas de integración de Listar órdenes	167
Figura 166. Pruebas de integración de Realizar órdenes	167

Figura 167. Pruebas de integración de Registrar órdenes de venta	168
Figura 168. Pruebas de integración de Registrar órdenes de compra	168
Figura 169. Pruebas de integración de Registrar pago de una orden.....	169
Figura 170. Encuesta de Satisfacción en <i>Google Forms</i>	171
Figura 171. Resultados: ¿Considera que la Aplicación Web es fácil de usar?	176
Figura 172. Resultados: ¿Considera que la manera de llevar a cabo la adquisición de productos dentro de la Aplicación Web es adecuada?	176
Figura 173. Resultados: ¿Considera que mediante el uso de la Aplicación puede agilizar la adquisición de productos?.....	177
Figura 174. Resultados: ¿Considera útil que la Aplicación se pueda instalar en un dispositivo directamente?	177
Figura 175. Resultados: ¿Cómo calificaría el proceso de realizar un pedido?	178
Figura 176. Resultados: ¿Qué dificultad representó para usted el proceso de buscar productos en la aplicación?	178
Figura 177. Resultados: ¿Qué tan probable es que usted vuelva a utilizar la aplicación? .	179
Figura 178. Resultados: ¿Qué tan probable es que recomiende la Aplicación a otras personas que necesiten adquirir algún producto o servicio en línea?.....	179
Figura 179. Formulario de inicio de sesión.....	191
Figura 180. Formulario para el registro de nueva cuenta de usuario	191
Figura 181. Correo electrónico para la confirmación de una nueva cuenta	192
Figura 182. Formulario para la recuperación de contraseña.....	192
Figura 183. Correo electrónico para el cambio de contraseña.....	192
Figura 184. Formulario para el cambio de contraseña	193
Figura 185. Página de perfil de usuario.....	193
Figura 186. Página de direcciones registradas del usuario.....	194
Figura 187. Formulario para el registro de una nueva dirección	194
Figura 188. Página de ordenes realizadas por el usuario	195
Figura 189. Comprobante de venta en formato PDF	196
Figura 190. Formulario para actualizar contraseña.....	196
Figura 191. Página de usuarios del sistema	197
Figura 192. Formulario para modificar usuario.....	197
Figura 193. Cuadro de diálogo para habilitar/deshabilitar cuenta de usuario	198
Figura 194. Página para el registro de un nuevo usuario.....	198
Figura 195. Página de configuraciones generales del sistema	199
Figura 196. Formulario para modificar el valor de una configuración	199
Figura 197. Página principal.....	200
Figura 198. Buscador de productos o servicios	201
Figura 199. Página de buscar productos por categorías.....	201
Figura 200. Página para ver detalles de un producto.....	202
Figura 201. Menú lateral de carrito de compras	203
Figura 202. Página de carrito de compras	203
Figura 203. Página de ver detalles de una orden.....	204
Figura 204. Formulario para el registro de la referencia de pago	205
Figura 205. Página de Inventario de productos.....	206
Figura 206. Página de detalles del inventario	206
Figura 207. Página de Órdenes de venta.....	207
Figura 208. Página de Ver detalles de la orden de venta	208
Figura 209. Formulario de nueva orden de venta - Paso 1	209
Figura 210. Formulario de nueva orden de venta - Paso 2	210
Figura 211. Formulario de nueva orden de venta - Paso 3 (primera parte)	211
Figura 212. Formulario de nueva orden de venta - Paso 3 (segunda parte).....	211

Figura 213. Confirmar registro de la orden de venta	212
Figura 214. Orden de venta en estado: Por pagar	212
Figura 215. Formulario para actualizar el estado de la orden a Por verificar pago	213
Figura 216. Página de Proveedores	213
Figura 217. Formulario para modificar proveedor	214
Figura 218. Página para el registro de un nuevo proveedor	214
Figura 219. Página de categorías	215
Figura 220. Formulario para modificar categoría	215
Figura 221. Página para el registro de una nueva categoría	216
Figura 222. Página de marcas	216
Figura 223. Formulario para modificar marca	217
Figura 224. Página para el registro de una nueva marca	217
Figura 225. Página de productos	218
Figura 226. Cuadro de dialogo para habilitar/deshabilitar producto	218
Figura 227. Página para el registro de un nuevo producto	219
Figura 228. Página para modificar un producto	219
Figura 229. Página de órdenes de compra	220
Figura 230. Formulario de nueva orden de compra - Paso 1	221
Figura 231. Formulario de nueva orden de compra - Paso 2	221
Figura 232. Cuadro de dialogo para establecer el precio de venta	222
Figura 233. Formulario de nueva orden de compra - Paso 3	222
Figura 234. Confirmar registro de la orden de compra	223
Figura 235. Página de impuestos	223
Figura 236. Formulario para modificar impuesto	224
Figura 237. Página para el registro de un nuevo impuesto	224
Figura 238. Sección de información del pago de la orden	225
Figura 239. Formulario para actualizar el estado de la orden a En proceso	225
Figura 240. Orden de venta en estado: En proceso	226
Figura 241. Formulario para actualizar el estado de la orden a Completada	226

Índice de anexos:

Anexo 1. Entrevista para la obtención de requerimientos	88
Anexo 2. Aprobación de los requerimientos de software	92
Anexo 3. Modelado de arquitectura física y lógica en base a los requerimientos del usuario utilizando el Modelamiento de desarrollo de software dirigido por ICONIX	93
Anexo 4. Interfaces gráficas de la implementación de la solución informática	132
Anexo 5. Manual de instalación de la Aplicación Web Progresiva	157
Anexo 6. Pruebas unitarias y de integración de la Aplicación Web Progresiva	163
Anexo 7. Modelo de Encuesta de Satisfacción	170
Anexo 8. Resultados de la Encuesta de Satisfacción	172
Anexo 9. Acta de validación de Pruebas de la Aplicación Web Progresiva para la gestión y comercialización de productos y servicios para la empresa Compumars	181
Anexo 10. Manual de usuario de la Aplicación Web Progresiva	189
Anexo 11. Certificado de traducción del resumen	227

1. Título

Desarrollo de una aplicación web progresiva para la gestión y comercialización de productos y servicios para la Empresa “Compumars” de la ciudad de Loja

2. Resumen

Actualmente, la falta de difusión en la red en empresas que buscan reconocimiento sobre los productos y servicios que ofrecen puede provocar la pérdida de posibles clientes potenciales, dado a que la sociedad actual dedica mucho tiempo a revisar sitios en internet y a realizar compras a través de ellos. A este contexto se suma Compumars, una empresa que además de buscar este reconocimiento digital, tiene la necesidad de gestionar y comercializar sus productos y servicios por medio de una aplicación informática, ya que hasta el momento han llevado estos procesos de manera manual. Considerando esto surgió la siguiente pregunta de investigación: ¿En qué grado una Aplicación Web ayudará a la empresa Compumars en la gestión y comercialización de sus productos y servicios? Por ello, el presente Trabajo de Titulación (TT) tiene como objetivo principal el desarrollo de una Aplicación Web Progresiva (PWA) para la gestión y comercialización de productos y servicios para la Empresa Compumars, valiéndose de las ventajas y facilidades que ofrece la metodología ICONIX y de las capacidades para la implementación que poseen las tecnologías basadas en el lenguaje de programación JavaScript usadas en el presente TT. Para cumplir con este objetivo, se consideraron 3 fases. En la primera fase, se indagó sobre los trabajos relacionados al objeto de estudio y se aplicó una entrevista al Gerente-Propietario de la empresa para la obtención de los requerimientos de la solución informática, además, se utilizó el modelamiento de software dirigido por la metodología ICONIX, obteniendo así los distintos diagramas UML que representaron la arquitectura física y lógica y que también proporcionaron un panorama más amplio de la solución. Seguidamente, en la segunda fase se implementó la arquitectura obtenida en una PWA siguiendo las indicaciones de la metodología ICONIX y empleando las tecnologías involucradas en el *stack* MERN (MongoDB, Express.js, React, Node.js) de JavaScript, dando como resultado el software funcional que cumplió con los requerimientos y los modelos obtenidos durante la primera fase. Por último, la fase final consistió en comparar los procesos de gestión y comercialización tradicional y web en un ambiente de pruebas simulado que permitió dar respuesta a la pregunta de investigación planteada.

Palabras claves: Gestión y comercialización, ICONIX, PWA, MERN stack.

2.1. Abstract

Currently, the lack of diffusion in the network of companies that seek recognition for the products and services they offer can cause the loss of potential customers since today's society spends much time reviewing Internet sites and making purchases through them. In this context is Compumars, a company that, apart from seeking this digital recognition, needs to manage and market its products and services through a computer application. Up to now, they have carried out these processes manually. Therefore, the following research question arose: To what degree will a Web Application help the company Compumars manage and market its products and services? Moreover, the main objective of this Degree Project (TT) is the development of a Progressive Web Application (PWA) for the management and commercialization of products and services for the Compumars Company, using the advantages and facilities offered by the ICONIX methodology, furthermore, the capacities for implementing technologies based on the programming language JavaScript used in the present TT. In order to achieve the objective, the process consisted of 3 phases. In the first phase, the researcher analyzed the work related to the study object, and an interview was conducted with the company's Manager-Owner to obtain the computer solution's requirements. Furthermore, the researcher used the software modeling directed by the ICONIX methodology; thus obtaining the different UML diagrams that represented the physical and logical architecture and provided a broader overview of the solution. Then, in the second phase, the architecture obtained was implemented in a PWA following the indications of the ICONIX methodology and using the technologies involved in the MERN stack (MongoDB, Express.js, React, Node.js) of JavaScript, resulting in the functional software that met the requirements and the models obtained during the first phase. Finally, the final phase consisted of comparing the traditional and web management and marketing processes in a simulated test environment that allowed us to answer the research question posed.

Keywords: Management, commercialization, ICONIX, PWA, MERN stack.

3. Introducción

La gestión y comercialización de productos y servicios son uno de los espacios más trascendentales dentro de una empresa, debido a que de estas depende la obtención de beneficios y la viabilidad del negocio [1]. Es por ello que, ante un mundo cada vez más exigente y digitalizado, las empresas tienen la tarea de enfrentar estos retos para de esta manera conservar la atención de sus actuales clientes y así mismo conseguir el interés de nuevos clientes potenciales. Debido a estos aspectos es importante que las empresas construyan su presencia en internet y el comercio electrónico (*E-commerce*) es una forma de venta por este medio, ya que facilita la interacción entre cliente-vendedor y por muchos otros detalles que se mencionan en la **sección 4.1.1 Comercio Electrónico**.

Con lo anterior mencionado, el presente TT propone el desarrollo de una Aplicación Web Progresiva (PWA) para la gestión y comercialización de productos y servicios, la misma que permitió a las personas que conforman la empresa Compumars a disminuir los problemas procedentes de la inadecuada gestión que hasta el momento estaban llevando, ya que la administración de sus artículos se daba de forma manual y, a más de eso, dependía de una sola persona. Asimismo, esta solución permitió a Compumars comercializar sus productos y servicios en línea, ya que la carencia de un servicio de venta-compra por internet estaba ocasionando que la empresa se pierda, posiblemente, de la atención de nuevos clientes.

Por tal motivo, el presente TT tuvo como objetivo principal “Desarrollar una Aplicación Web Progresiva (PWA) para la gestión y comercialización de productos y servicios para la Empresa Compumars”. Y para lograrlo se desarrollaron tres objetivos específicos, el primero fue “Construir el modelo de arquitectura física y lógica en base a los requerimientos del usuario utilizando el Modelamiento de desarrollo de software dirigido por ICONIX”. Seguidamente, se dio paso al segundo objetivo específico que fue “Implementar la arquitectura de la solución informática en una Aplicación Web Progresiva (PWA) mediante el uso de tecnologías JavaScript y la metodología ICONIX”. Por último, el tercer objetivo fue “Comparar los procesos de gestión y comercialización tradicional y web en un ambiente de pruebas simulado para determinar los beneficios de la actualización”.

En cuanto al desarrollo de los objetivos específicos planteados, en primer lugar, se empleó la etapa de Modelado: análisis y diseño descrita por la metodología ICONIX y de esta manera se pudo obtener el modelo de arquitectura física y lógica en base a los requerimientos del usuario que se estableció en el primer objetivo específico. Posteriormente para la ejecución del segundo objetivo específico se siguieron las fases de Inicialización y Construcción establecidas por ICONIX y además se hizo uso del *stack* MERN de JavaScript. Por último, en cuanto a comparar los procesos de gestión y comercialización tradicional y web, se determinó

realizar pruebas en un ambiente simulado tomando en cuenta las actividades más relevantes de la metodología propuesta por el autor Pablo González [2]. Todos estos apartados se los puede evidenciar en la sección de **Resultados** del presente TT.

Finalmente, las secciones que componen el presente proyecto son las siguientes: en la **sección Marco Teórico** se encuentran los antecedentes, trabajos relacionados y los distintos conceptos necesarios para la comprensión del tema; la **sección Metodología** habla sobre el área de estudio, el procedimiento y los recursos que fueron utilizados para el desarrollo del presente TT; la **sección Resultados** se compone de las evidencias obtenidas del desarrollo de los objetivos específicos planteados; la **sección Discusión** se analizan e interpretan los resultados obtenidos desde el punto de vista del autor; la **sección Conclusiones** abarca los acontecimientos más sobresalientes del proyecto; finalmente la **sección Recomendaciones** posee las sugerencias que se pueden tomar en cuenta para posibles trabajos futuros.

4. Marco Teórico

En esta sección se abordan conceptos preliminares con el fin de construir una base teórica para el desempeño del presente Trabajo de Titulación (TT). Para ello, en la sección 4.1 se abarca el Comercio electrónico y las Técnicas de manejo de inventario. En la sección 4.2 se detallan conceptos relacionados a la Ingeniería de Software, en donde se incluye la Metodología ICONIX y el desarrollo de las Aplicaciones Web Progresivas (PWA). En la sección 4.3 de Tecnologías, se realiza una contextualización de JavaScript y MongoDB. Por último, en la sección 4.4 se exponen los distintos trabajos relacionados con el objeto de estudio.

4.1. Antecedentes

4.1.1. Comercio electrónico

El comercio electrónico (*e-commerce* por su terminología en inglés) es una forma de venta por medio de Internet que facilita la interacción entre cliente-vendedor y que ha ganado popularidad debido a la pandemia mundial de la COVID-19, este es sin duda una de las soluciones que se han destacado por la facilidad de uso, tiempo ilimitado para navegar entre los productos/servicios y para leer especificaciones a detalle de los mismos [3].

Durante la crisis de la COVID-19, el comportamiento de los consumidores y sus hábitos de compra cambiaron fundamentalmente [4]. Las encuestas más recientes arrojaron que del total de usuarios de plataformas de compra en línea encuestados, el 8% de consumidores afirmó que regresaría a las compras en tiendas físicas. Se reporta también un 48% que decide mantenerse con compras online por seguridad sanitaria, según PRO Ecuador (parte del Viceministerio de Promoción de Exportaciones e Inversiones) [PRO, Ecuador]. Por tanto, las tiendas virtuales son una de las soluciones emergentes que supone una solución viable a largo plazo para los negocios que desean mejorar su alcance e impacto en sus clientes con facilidades para navegar entre sus servicios y comprar mediante Internet [5][6].

El comercio electrónico ha ayudado a las empresas a establecer un mercado más amplio proporcionando canales de distribución más baratos y eficientes para sus productos y servicios. Añadiendo que también puede funcionar y estar disponible las 24 horas al día, 7 días a la semana, y se puede desarrollar entre Empresa-Consumidor o B2C (*Business to Consumer*), Empresa-Empresa o B2B (*Business to Business*), estos modelos tienen una estrategia de precios, una estrategia de marketing y un público objetivo diferente, pero el objetivo no deja de ser el de establecer la presencia en línea y crear confianza [7]. A través de las plataformas de comercio electrónico se puede comprar casi cualquier cosa y es ideal

para que el cliente pueda analizar los detalles del producto o servicio y dejar su elección para más tarde [8][9].

4.1.2. Técnicas de manejo de inventario

Por lo general, dentro de cualquier empresa el inventario es el activo corriente más grande, por lo que es de vital importancia medirlos adecuadamente para asegurar estados financieros precisos. Estos están ahí para garantizar la continuidad en la producción y prestación de productos y servicios, esto significa que tenerlos cuesta dinero. Es por ello que los expertos en el campo han desarrollado métodos para tener la mínima cantidad posible de inventario sin afectar la producción y manteniendo la satisfacción del cliente [10]. De esta manera, es importante hablar de técnicas de manejo y valuación de inventarios, que entre las más populares se tiene [11]:

4.1.2.1. Primeros en entrar, Primeros en salir (PEPS)

Conocido como *first in, first out* (FIFO) por sus siglas en inglés. Esta técnica plantea que los primeros productos que se adquieren son los primeros que se venden, quedando en inventario los artículos más recientes. Contrario a Últimos en entrar, Primeros en salir (UEPS), este método plantea la suposición más real en la práctica. Lo más lógico es que se vendan aquellos artículos más con más tiempo en inventario logrando reducir el riesgo de obsolescencia o vencimiento [12].

Con Primeros en entrar, Primeros en salir (PEPS), el inventario que queda en el almacén se valora a un precio cercano al valor que fija el mercado (considerando un período inflacionario). Es decir que los artículos que llevan más tiempo en el inventario, tienden a tener menor costo que los adquiridos recientemente. Esto último es lo que nos lleva a tener un menor costo de ventas y con él un aumento del beneficio obtenido por la empresa, pero también aumento en el pago de impuestos [10][13].

4.1.2.2. Últimos en entrar, Primeros en salir (UEPS)

Last in, first out (LIFO). Este método de valuación es opuesto al PEPS, y por ende plantea que las últimas unidades que ingresaron, serán las primeras en ser vendidas. En la práctica, no es un método de amplio uso, debido a las desventajas ocasionadas a partir de la naturaleza de los artículos. Si una empresa elige operar bajo la modalidad UEPS, su inventario podría quedarse obsoleto o estar sujeto a daños [13].

Por lo tanto, el método se aleja de la realidad ante variaciones significativas en el precio de los artículos. En un período inflacionario la empresa saldrá inicialmente del inventario más reciente que tiene (las últimas que entraron), lo que la llevará a tener un

inventario infravalorado o subvalorado. Sin embargo, esto también indica que el método UEPS otorga una reducción en la carga tributaria si se opera bajo una economía inflacionaria. Pero, por otra parte, si la economía atraviesa una caída en los precios (deflación) ocurrirá justo lo contrario. Es por ello que la empresa llegará a tener un inventario sobrevalorado, pues el artículo más reciente se adquiere a menores precios de lo que se adquirió el más antiguo [14].

4.1.2.3. Promedio ponderado o Costo promedio constante

Por último, este método de valuación y manejo de inventario consiste en sumar el costo total de las unidades en existencia y dividirlo entre el número de unidades para obtener un precio promedio. Este método trata de estabilizar las probables fluctuaciones de los precios por lo que puede ser recomendable cuando la empresa no tiene muchas variaciones en su inventario [15].

La principal ventaja de usar esta técnica es que resulta de fácil aplicación y presenta una utilidad razonable al promediar los costos antiguos y actuales. Sin embargo, tiene la desventaja de que no permite llevar un control detallado del costo de las mercancías entrantes y salientes [10].

Por otro lado, cada vez que se realice una compra se debe recalcular el promedio, en razón a que esa nueva compra puede tener un precio distinto e incrementa la cantidad de unidades disponibles, que será el nuevo divisor que cambiará el costo promedio. Además, en lo que concierne al tratamiento de las devoluciones por este método, es similar o igual que los otros; la devolución se hace por el valor en que se ingresaron o se le dieron salida, según corresponda. De modo tal que las devoluciones en ventas se les vuelve a dar entrada al inventario por el valor del costo con que se sacaron en el momento de la venta y entran nuevamente a ser parte de la ponderación. En el caso de las devoluciones en compras, es decir, cuando el producto comprado aún no ha sido vendido, estas salen del inventario por el costo en que se incurrió al momento de la compra y se procede nuevamente a realizar la ponderación. Esta manera de funcionar, convierte al Promedio Ponderado en la técnica más utilizada dentro del manejo de inventarios [13][14].

4.2. Ingeniería de Software

4.2.1. Metodología ICONIX

La metodología ICONIX se define como un proceso de desarrollo de software de tipo pesado-ligero, ya que se encuentra entre la complejidad de la metodología RUP (*Rational Unified Processes*) y la simplicidad de la metodología XP (*Extreme Programming*), pero sin dejar de lado las tareas de análisis y de diseño que XP no contempla [16][17].

ICONIX se enfoca en los casos de uso y eso lo convierte en un proceso relativamente pequeño, pero sin descartar las fases de análisis y el diseño como hacen la mayoría de las metodologías ligeras. Además, propone el uso de un subconjunto determinado de lo más importante y útil del Lenguaje Unificado de Modelado (UML), al mismo tiempo que se centra en la trazabilidad de los requisitos del usuario [18]. En la Figura 1 se puede observar cómo es que se lleva a cabo el proceso propuesto por ICONIX.

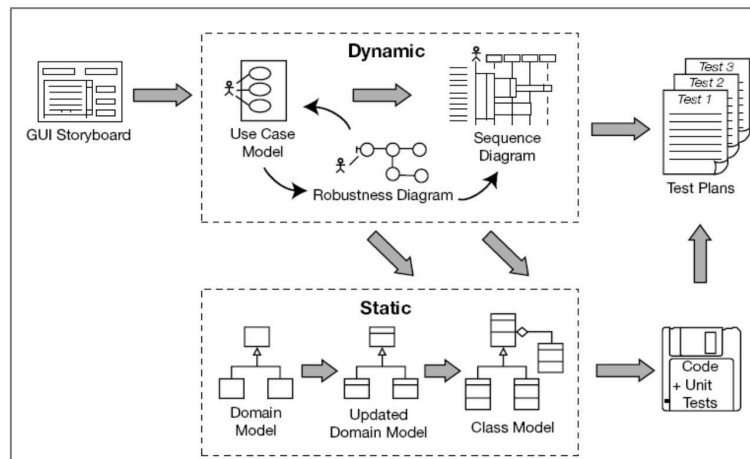


Figura 1. Flujo de trabajo de la Metodología ICONIX

De igual manera, esta metodología ofrece un enfoque simplificado de desarrollo de software que incluye un conjunto mínimo de diagramas y técnicas que permite a pequeños grupos de un proyecto pasar de los casos de uso al código de forma rápida y eficiente, lo que faculta la entrega continua y en plazos breves de software funcional. Según [19] estos diagramas corresponden a:

- Diagramas de clases, tanto en el análisis (dominio del problema) y diseño.
- Casos de uso (diagramas y textos).
- Los diagramas de secuencia.
- Diagramas de robustez.

4.2.1.1. Características principales de la metodología ICONIX

Como lo menciona [16], la planificación del desarrollo del proceso ICONIX tiene 3 características fundamentales:

- Es **iterativo e incremental**: existen varias iteraciones entre el desarrollo del modelo del dominio y la identificación de los casos de uso. Es incremental puesto que cada nueva versión se construye progresivamente sobre la versión anterior.

- Se centra en la **trazabilidad**: es decir que la metodología tiene la capacidad de seguir una relación, ya que cada paso se encuentra referenciado por algún requisito.
- Es **impulsado por modelos**: ICONIX permite el uso dinámico de lo más útil de UML, ya que la planificación de desarrollo se basa en las tareas de ingeniería que resultan en los diagramas de los modelos.

4.2.1.2. Fases de la metodología ICONIX

La metodología ICONIX no impone ninguna regla o restricción sobre cómo organizar el proyecto cuando se habla de fases, excepto que algunos análisis de requisitos y trabajos propios del diseño deben tener lugar antes de pasar a la codificación. Sin embargo, el proceso propone las siguientes fases [20]:

1. Modelado: análisis y diseño.
2. Planificación: inicialización.
3. Planificación: construcción

Modelado: análisis y diseño.

Aquí es donde se encuentra la parte “medular” de esta metodología. De acuerdo a [21] en esta fase se encuentran 3 de los componentes principales de ICONIX, que son:

- **Análisis de requerimientos**: el primer levantamiento de las necesidades a cubrir con el proyecto, es decir, identificar objetos del mundo real con diagramas de modelado de dominio, reglas y casos de uso.
- **Depurar la lista de requerimientos y construir la ficha de casos de uso**, se puede valer de algunas interfaces de usuario para una mejor comprensión tanto para el equipo como para el cliente.
- **Diseño detallado**: ideal para determinar el comportamiento de los objetos mediante modelos de interacción, es decir, diagramas de secuencia, modelos estructurales, con ayuda de diagramas de paquetes y de clases, también se define la arquitectura física de la solución mediante diagramas de componentes y de despliegue.

Planificación: inicialización

Esta es la fase en la que el equipo empieza tanto a explorar los requerimientos, casos de uso, como a familiarizarse con las tecnologías que se utilizarán para el desarrollo del

producto, y en donde se elabora el prototipo del sistema. Esta fase tiene la particularidad de que se ejecuta una sola vez para todo el proyecto [20][22].

Planificación: construcción

La fase de construcción es donde más se evidencia la incidencia de XP en esta metodología, ya que su dinámica consiste en ofrecer pequeñas series de lanzamientos (esto en las metodologías ágiles se denomina “incremento”). El mantenimiento se considera parte de esta fase puesto que desde la primera versión que llega al cliente el producto entra en fase de mantenimiento [21][23].

4.2.2. Aplicaciones Web Progresivas

Las Aplicaciones Web Progresivas (PWA por sus siglas en inglés), nacen en los últimos años como un nuevo concepto para acercar más a las aplicaciones web de las nativas con el fin de ofrecer la misma experiencia de usuario. Para lograr este fin, estas aplicaciones aprovechan todo el potencial que ofrecen las API modernas de los navegadores [24]. Los objetivos principales de las PWA son: lograr el mayor rendimiento posible en cualquier dispositivo móvil, que los tiempos de carga de las funcionalidades sean lo más mínimo posible, que la interfaz gráfica se asemeje en su mayoría a las aplicaciones nativas, que, de ser posible, se pueda trabajar sin conexión a internet, y la capacidad para poder enviar notificaciones a los distintos dispositivos que hagan uso de la aplicación [25].

Asimismo, otra de las características relevantes que ubican a las PWA como una tecnología novedosa, es que estas pueden ser instaladas en cualquier sistema operativo (como si se tratase de una aplicación nativa) directamente desde el navegador [26]. Todas estas características son posibles debido a que aplica una arquitectura que se compone de los siguientes elementos:

- Un *Service Worker*, el cual es considerado como el cerebro de todo el funcionamiento. Esta tecnología no es más que un código JavaScript que se ejecuta en segundo plano en el navegador y de acuerdo a las necesidades que tenga el desarrollador, este puede administrar caches, interceptar solicitudes de red y notificaciones push, permitir el comportamiento offline, entre otras tareas. Es necesario tener en cuenta que un *Service Worker* no depende de una página web, lo que da lugar a ejecutar funciones que no dependen de un sitio web o de la interacción del usuario [27].
- Un archivo conocido como *Manifest*, un fichero de tipo JSON que se encarga de especificar propiedades (tales como nombre, descripción, iconos, colores de tema,

entre otras) para que la aplicación web posea las características de apariencia de una aplicación nativa [28].

4.2.2.1. Ventajas

Según [29], las PWA ofrecen un número de ventajas que las empresas pueden aprovecharlas y obtener un gran beneficio. Algunas de las más principales son:

- Son rentables, el desarrollo de una PWA para las empresas supone una reducción de costes debido a que el desarrollo no está enfocado en una sola plataforma, como IOS, Android o cualquier otro sistema operativo.
- Es *SEO-friendly* (Apta para el SEO), esto quiere decir que una PWA está construida de tal manera que contribuye a la optimización de los motores de búsqueda, lo cual es un aspecto muy relevante para las empresas, ya que esto les ayuda a fijarse en los primeros lugares de búsqueda.
- No dependen de una tienda de aplicaciones. Para poder instalar una PWA en algún dispositivo basta con insertar el enlace web en algún navegador y de esta manera aparecerá un cuadro de dialogo el cual provee la opción para instalar la aplicación de manera rápida. Además, al no depender de una tienda específica, al momento de producirse cambios en cualquier aspecto de la aplicación, esta se actualizará automáticamente [27].
- Consumen la menor cantidad de datos. Las PWA no requieren del uso de grandes cantidades de datos como lo suelen hacer las aplicaciones web normales, esto hace que estas aplicaciones tengan un mejor rendimiento.

4.3. Tecnologías

4.3.1. JavaScript

JavaScript es un lenguaje de programación que permite implementar funciones complejas en las páginas web. Cada vez que un sitio web hace algo más que mostrar información estática, como mostrar actualizaciones de contenido oportunas, mapas interactivos, gráficos 2D/3D animados, video jukeboxes de desplazamiento, etc., puede apostar que JavaScript está probablemente involucrado. Es la tercera capa de la tarta de tecnologías web estándar, dos de las cuales son HTML (HyperText Markup Language - Lenguaje de marcado de hipertexto) y CSS (Cascading Style Sheets - Hojas de estilo en cascada) [30].

JavaScript, el estándar de la web, desde sus inicios en 1995 hasta la actualidad se ha posicionado como uno de los lenguajes de programación con mayor aceptabilidad en el año

2021, tanto que el 68.62% de 58,031 desarrolladores profesionales lo usan, según los datos de Stack Overflow [31], sitio de consultas relacionadas a desarrollo de software, siendo el top 1, mientras que lenguajes de programación como Python y Java se ubican en las posiciones 4 y 7 respectivamente. Además, 13,719 empresas de todo el mundo usan JavaScript en sus desarrollos, según Stackshare [32], entre las empresas que más destacan están Paypal, Walmart, Instagram, LinkedIn, Netflix, Groupon y otros.

Adicionalmente y considerando las nuevas tendencias de los usuarios por el uso de herramientas basadas en la web [33], JavaScript y sus marcos de trabajo se han impuesto en los últimos años como la herramienta favorita para desarrollar software del lado del cliente [34][35]. Pero no solo aplica a *frontend*, sino que también se ha vuelto popular en la construcción de los servicios y APIs REST (*Representational State Transfer* - Transferencia de Estado Representacional) con su entorno de ejecución Node.js.

4.3.1.1. Node.js

Es un entorno de ejecución para JavaScript construido con el motor de JavaScript V8 de Chrome. Ideado como un entorno de ejecución de JavaScript orientado a eventos asíncronos, Node.js está diseñado para crear aplicaciones *network* escalables. Asimismo, pueden atenderse muchas conexiones simultáneamente. Por cada conexión, se activa la devolución de llamada (*callback* por su traducción al inglés), pero si no hay trabajo que hacer, Node.js se entrará en reposo [36].

JavaScript es el lenguaje de programación más popular dentro del desarrollo web, y gracias a Node.js se ha utilizado ampliamente en el lado del servidor, siendo a día de hoy el más popular junto al framework Express.js, el mismo ocupa el puesto 7 de los repositorios de mayor aceptación en la plataforma GitHub. Con 23 millones de descargas semanales, ExpressJS ha llegado al top 5 en favoritos, según StackOverflow, y el puesto 4 entre los más queridos [31]. Además, Node.js ocupa el lugar 8 entre los puestos de trabajo más demandados en USA [37][38].

Express.js

Express es una infraestructura de aplicaciones web de Node.js mínima y flexible que proporciona un conjunto sólido de características para las aplicaciones web y móviles. Con miles de métodos de programa de utilidad, Protocolo de Transferencia de Hipertexto (HTTP) y middleware a su disposición, la creación de una API sólida es rápida y sencilla. Express.js proporciona una delgada capa de características de aplicación web básicas, que no ocultan las características provenientes de Node.js [39].

Express.js se ha convertido en uno de los frameworks más utilizados dentro del desarrollo de aplicaciones web, gracias a que es minimalista y flexible, y al ser de código abierto ha permitido que miles de desarrolladores en el mundo contribuyan para agregar cada vez nuevas funcionalidades, incrementando su poderío. Se encuentra alojado en el entorno de ejecución de Node.js, pero equipando un conjunto versátil de funcionalidades muy útiles para la eficaz construcción de aplicaciones web y móviles, lo que lo convierte en la herramienta más preferida por los programadores [40][41].

4.3.1.2. React

Es una librería de JavaScript, dos de las aplicaciones web más utilizadas en el mundo están desarrolladas con React, estas son Facebook e Instagram, por supuesto entre muchas otras, que utilizan la capacidad interactiva que proporciona esta librería. Si el objetivo es lanzar una web robusta, dinámica y escalable con un rendimiento fiable, la elección debe ser React.js. Otra característica a destacar es la capacidad de traer elementos reutilizables. Esto permite a los desarrolladores reutilizar estos componentes en varias aplicaciones. Por último, destacar que ofrece un conjunto de paquetes dedicados al diseño UI (*User Interfaces*), como por ejemplo Material-UI, el cual se inspira en el diseño de materiales de Google [42][43].

Una de las librerías más jóvenes de JavaScript, React, desde su liberación en el 2013 a la comunidad por parte de Facebook, sus creadores, se ha convertido en una de las opciones favoritas para desarrollar del lado del cliente siendo el top 1, entre los desarrolladores que participan de la comunidad Stack Overflow [31], seguido de Vue.js y Angular, frameworks de JavaScript, mientras que Django, Flask y Laravel ocupan las posiciones 3, 6 y 15 respectivamente. Incluso muchos sitios de aficionados al desarrollo de software mencionan a React entre sus 3 preferidos. Además, según datos de GitHub, React es el segundo marco web con más estrellas en sus repositorios. Según *Job Openings* en los Estados Unidos, los puestos más demandados son para desarrolladores de React [44][45][46].

4.3.2. MongoDB

Este es un sistema gestor de base de datos (SGBD) No Relacional orientado a documentos, el más destacado del grupo NoSQL [47]. Se ha popularizado tanto desde su lanzamiento en 2009, hasta llegar a ser top 5 en las bases de datos más usadas por los desarrolladores que forman parte de Stack Overflow, siendo MySQL, PostgreSQL, SQLite y Microsoft SQL Server los SGBD que lo superan [31].

Tal como lo menciona [48], las principales características que tiene MongoDB son:

- **Flexibilidad:** Ya que los datos se almacenan en formato de documento utilizando JSON, lo convierte en un documento sin esquema y por ende se mapea a tipos de lenguaje de programación nativos.
- **Lenguaje de consulta enriquecido:** Provee la característica de los Sistemas de gestión de bases de datos relacionales (RDBMS) pero con características adicionales propias. Las consultas dinámicas, la ordenación, los índices secundarios, las actualizaciones, la agregación, el *upsert* (actualizar si el documento existe e insertar si no existe) son algunas de las características del RDBMS y la flexibilidad y la escalabilidad son las que ofrece MongoDB.
- **Facilidad de uso:** Es muy fácil de instalar, utilizar, mantener y configurar. Además, tiene una documentación muy bien detallada para todas las funcionalidades existentes.
- **Alto rendimiento:** Proporciona una persistencia de datos de alto rendimiento, debido a que reduce la actividad de E/S en el sistema de base de datos al admitir documentos incrustados. Por otro lado, el uso de la indexación permite realizar consultas más rápidas [49].
- **Alta disponibilidad:** soporta la facilidad de replicación llamada conjunto de réplicas (o *replica set*). El conjunto de réplicas viene a ser un grupo de servidores que mantiene el mismo conjunto de datos, lo que proporciona una mayor disponibilidad de los datos.

4.4. Trabajos relacionados

4.4.1. Desarrollo de una aplicación WEB para pedidos en línea en la casa comercial Calva & Calva.

Con el afán de lograr un trabajo investigativo sobre la influencia del comercio electrónico, y como este puede servir de base para fortalecer el nivel competitivo de las empresas que preocupadas por lograr competitividad en un mercado que cada día demanda de nuevos y mejores servicios, se ha optado por desarrollar una aplicación Web para pedidos en línea. La aplicación de pedidos en línea posibilitará a las empresas ofertar sus productos y servicios, de tal forma que los Clientes, previa a la creación de una cuenta de usuario puedan acceder por medio de Internet a los servicios ofertados, como es la adquisición de una gama de productos. En este caso Calva & Calva comercial, que es la empresa base para el desarrollo de la aplicación, es quién receptorá el pedido del Cliente y posteriormente procederá a despacharlo, entregándolo a domicilio, al momento de la entrega el Cliente deberá efectuar el pago respectivo [50].

4.4.2. Diseño e implementación de una tienda virtual para la venta de equipos informáticos para la empresa "SETCOMPC" de la ciudad de Loja.

El proyecto desarrollado cuenta con un portal Web para ventas de partes de computadores y equipos de computación en general donde los clientes podrán seleccionar las partes que necesiten y realizar sus adquisiciones; suscripciones a boletines y promociones, consultas de productos, cotizaciones Además el sistema permite realizar la administración, gestión y actualización de los productos ofrecidos, el envío de boletines, promociones a los clientes, obtener reportes, administración de ventas y pedidos realizadas, administración de clientes. Se garantizará que la información crítica que viaje desde el cliente al servidor y la residente en las bases de datos estará protegida con técnicas de encriptación especialmente las claves de usuario, números de tarjetas de crédito, fecha de expiración, entre otros que se considere necesario, siempre buscando garantizar la seguridad de la información especialmente en los procesos de criticidad elevada, como por ejemplo los pagos con tarjetas de crédito. Uno de los aspectos más importantes hoy en día en el desarrollo de aplicaciones Web es reducir el tiempo de latencia o de respuesta ante procesos lentos y que demandan requerimientos de las comunicaciones excesivos, por tal razón dentro del proyecto se han implementado técnicas que mejoren el rendimiento como AJAX especialmente en procesos donde intervengan imágenes, múltiples llamadas al servidor para un proceso sencillo y que involucre un segmento pequeño o definido de una misma página. Adicionalmente ejecuciones de procesos del lado del cliente con JavaScript principalmente procesos de validación de elementos de entrada. Una idea que promovimos en el proyecto es desarrollar una base de conocimiento que se alimentará de las experiencias aprendidas a lo largo de la construcción del proyecto, que permita ser el punto de partida en nuevos proyectos. Dentro de un proyecto de software siempre es importante y necesario poder tener planes de validación desde el punto de vista del programador, de un tester auditor, y de los futuros usuarios del sistema. Así también dentro del proyecto se promovió la realización de un plan de validación que permita alcanzar los objetivos trazados y cumpla con los requerimientos especificados en la etapa de análisis. Partiendo de este punto se requirió realizar pruebas unitarias, pruebas funcionales y pruebas de usuarios. Las pruebas unitarias son desarrolladas o van a la par del proceso de creación de los módulos respectivos y son ejecutadas por el programador encargado. Las pruebas de funcionalidades son realizadas por el tester y deben buscar la verificación de los casos de uso y el cumplimiento de los requerimientos definidos por el cliente, finalmente las pruebas de usuarios son desarrolladas por las personas que manejan el sistema a fin de que estos den su aprobación en cada una de las especificaciones emitidas. Finalmente mencionar que el sistema está disponible en internet para todas las personas que deseen realizar sus compras vía online en la empresa SetComPc [51].

4.4.3. Creación de una tienda virtual para compras de abarrotes en línea utilizando metodología de desarrollo SCRUM.

El proyecto está compuesto de tres capítulos, detallados a continuación: el capítulo uno describe los antecedentes, la situación actual y cómo las empresas proveedoras de abarrotes ven una oportunidad ante estas necesidades. El capítulo dos muestra el método y la solución tecnológica implementado para el sitio web “mi tienda de abarrotes”, se habla de las herramientas usadas y el proceso SCRUM. En el capítulo tres se explica los beneficios que brindan a los clientes y empresas la creación de este sitio web [52].

4.4.4. Desarrollo de un software para la creación de tiendas virtuales orientadas a b2c que facilite el ingreso de pequeñas y medianas empresas ecuatorianas al comercio electrónico.

El presente artículo es una síntesis del trabajo realizado durante el desarrollo de un software que permite la introducción al mundo del comercio electrónico a pequeñas y medianas empresas que quieren tener presencia en el mundo de Internet, a través de la creación de una tienda virtual. Sus excelentes características, también lo presentan como una opción atractiva para las personas con conocimientos básicos del manejo del computador, que tienen una buena idea de negocio y poco presupuesto. CreaTe - el software de creación de tiendas virtuales - se presenta como una solución de comercio electrónico, que automatiza muchas tareas relacionadas con la creación y administración de una tienda virtual, desde la realización de las páginas de la tienda hasta el procesamiento de solicitudes de compra [53].

5. Metodología

Esta sección aborda la metodología utilizada para el desarrollo y cumplimiento del presente TT. Para ello, en la sección 5.1 se describe el área de estudio en donde se llevó a cabo el progreso de este proyecto; la sección 5.2 expone el procedimiento seguido para consumir cada uno de los objetivos planteados; por último, en la sección 5.3 se dan a conocer los diferentes recursos que fueron empleados.

5.1. Área de estudio

El presente trabajo de titulación se llevó a cabo en la empresa Compumars, la cual es una entidad dedicada a la comercialización de equipos y partes de sistemas informáticos y que también ofrece asesoría técnica en Tecnologías de la información y la comunicación (TICs), Compumars se encuentra ubicada en la ciudad de Loja en dirección: Sucre 154-43 y Juan de Salinas, y cuyo representante legal es Mario Enrique Cueva Hurtado. El TT se realizó en un ámbito académico y durante el periodo de abril 2022 a septiembre 2022 establecido por la Universidad Nacional de Loja.

5.2. Procedimiento

Con la finalidad de alcanzar el objetivo general planteado para el presente TT se llevó a cabo el siguiente procedimiento:

5.2.1. Construir el modelo de arquitectura física y lógica en base a los requerimientos del usuario utilizando el Modelamiento de desarrollo de software dirigido por ICONIX.

- Se realizó una entrevista al gerente-propietario de la empresa Compumars, el Ing. Mario Enrique Cueva Hurtado con la finalidad de obtener información detallada de las funciones que realiza la empresa, las mismas que estuvieran relacionadas al proceso de gestión y comercialización de sus productos y servicios (véase **Anexo 1. Entrevista para la obtención de requerimientos**).
- Se analizó toda la información recolectada y se procedió a organizar el documento de modelado de requerimientos y diagrama del dominio (véase **sección 6.1.1. Análisis y diseño preliminar**).
- Se realizó el modelado de la arquitectura física y lógica, el mismo que incluye: casos de uso, diagramas de secuencia, diagrama de paquetes, diagrama de clases y el diseño de la arquitectura física de la solución informática (véase **sección 6.1.2. Diseño de la arquitectura física y lógica**).

5.2.2. Implementar la arquitectura de la solución informática en una Aplicación Web Progresiva (PWA) mediante el uso de tecnologías JavaScript y la metodología ICONIX.

- Se realizó la implementación en base al modelo de la arquitectura física y lógica obtenido en la fase de modelamiento de software dirigido por ICONIX, de esta manera se construyeron los módulos de software (usuarios, productos, servicios, inventario, ventas) con la ayuda del lenguaje de programación JavaScript, con el entorno de ejecución Node.js y el *framework* Express.js para la parte del servidor (*Backend*), y con la librería React y el *framework* Next.js para el apartado del cliente (*Frontend*) (véase **sección 6.2.1. Implementación**).
- Se ejecutó pruebas unitarias y de integración de software de la Aplicación Web Progresiva con el fin de validar que la solución informática codificada posee una correcta interacción entre sus módulos y que las distintas piezas de código funcionan apropiadamente de forma individual (véase **sección 6.2.2. Realizar pruebas de funcionamiento del software**).

5.2.3. Comparar los procesos de gestión y comercialización tradicional y web en un ambiente de pruebas simulado para determinar los beneficios de la actualización.

- Se elaboró la planificación para la ejecución de pruebas referentes a la simulación de la gestión y comercialización de productos y servicios tradicional (manual) versus web (véase **sección 6.3.1. Planificación del caso de prueba para la simulación manual vs web**).
- Se realizó la ejecución de la simulación con un grupo de estudiantes de la carrera de Ingeniería en Computación para conocer su grado de satisfacción; y por otro lado con el Gerente-Propietario y el personal de Compumars, estas pruebas permitieron obtener datos relevantes para su posterior análisis (véase **sección 6.3.2. Ejecución de la simulación**).
- Se recopilaron los datos obtenidos durante la ejecución de la simulación planificada para realizar un análisis cuantitativo de la información rescatada (véase **sección 6.3.3. Establecer un análisis cuantitativo de los resultados**).

5.3. Recursos

Con el propósito de dar cumplimiento a los objetivos planteados en el presente TT, se hizo uso de los siguientes recursos:

5.3.1. Recursos Científicos

5.3.1.1. Estudios de caso

Los estudios de caso fueron de gran utilidad para obtener información relevante y relacionada a la gestión y comercialización de productos y servicios, de tal manera que ayudaron a entender más sobre el objeto de estudio.

5.3.1.2. Investigación Bibliográfica

La técnica de Investigación Bibliográfica fue útil para la recolección de información que tenga validez científica y que por ende ayude a sustentar el fundamento teórico del presente TT, por lo que las fuentes bibliográficas se basaron en libros, artículos científicos, tesis, revistas científicas, trabajos relacionados, entre otros.

5.3.2. Recursos de Hardware y Software

5.3.2.1. Hardware

- **Laptop Lenovo Core i7 modelo Ideapad serie 330:** Equipo que fue necesario para las distintas investigaciones que requería este proyecto, así como para cumplir con el desarrollo de solución informática.

5.3.2.2. Software

- **Mendeley Reference Manager:** programa informático que permitió la gestión de las fuentes bibliográficas.
- **Visual Studio Code:** editor de código utilizado para la implementación de la Aplicación Web Progresiva.
- **ClickUp:** software utilizado para la gestión de tareas y actividades del proyecto, permitió el seguimiento de los objetivos y la planificación de horarios.
- **MERN Stack:** conjunto de software de JavaScript compuesto por *MongoDB*, *Express.js*, *React* y *Node.js* que permitió construir la solución informática.

5.3.3. Recursos Técnicos

5.3.3.1. Entrevista

La entrevista permitió la recolección de información relevante sobre la gestión y comercialización de productos y servicios de la empresa Compumars, de esta manera se pudo obtener la especificación de requerimientos necesarios para posteriormente desarrollar la solución informática.

5.3.3.2. Encuesta

Esta técnica permitió conocer el grado de satisfacción de un grupo concreto de personas acerca de la Aplicación Web Progresiva construida, cuya información obtenida fue de gran utilidad para su posterior análisis.

5.3.3.3. Metodología de desarrollo de Software

La metodología empleada para el desarrollo de la Aplicación Web Progresiva (PWA) fue la metodología ICONIX, misma que cuenta con 3 fases detalladas a continuación:

- **Modelado: análisis y diseño:** En esta fase se realizó la recolección de la información relacionada a la gestión y comercialización de productos y servicios de la empresa Compumars, para de esta manera conocer y delimitar las necesidades y posteriormente elaborar el documento de Modelado de Requerimientos y Diagrama de Dominio. Esto se pudo realizar gracias a la entrevista realizada al gerente-propietario de Compumars.

Además, se procedió a depurar la lista de requerimientos efectuando las respectivas validaciones con el cliente para seguidamente producir el diseño de la arquitectura física y lógica, la misma que consta de: modelado de casos de uso, modelado de interacción mediante diagrama de secuencia, modelado estructural mediante diagrama de paquetes y diagrama de clases y por último el diseño de arquitectura física. Todas estas actividades otorgan el cumplimiento del primer objetivo detallado en la sección de Resultados.

- **Planificación: inicialización:** En esta fase se realizó la exploración de todos los requerimientos y casos de uso obtenidos, así como el inicio de la codificación del proyecto, generando la estructura de directorios adecuada para lograr una armonía entre las tecnologías a usar y los requerimientos. Esta fase complementó el cumplimiento del primer objetivo, y marcó el comienzo para el desarrollo del segundo objetivo.
- **Planificación: construcción:** Por último, en esta fase se realizó la codificación de cada uno de los módulos que conforman la solución informática, validando con el cliente los resultados obtenidos con cada uno de los requisitos previamente establecidos. Esta última fase de la metodología ICONIX permitió el cumplimiento del segundo objetivo del presente TT.

6. Resultados

Con el afán de exponer la evidencia de los resultados obtenidos del cumplimiento de los objetivos específicos establecidos para el presente TT se pudo obtener los siguientes resultados:

6.1. Objetivo 1: Construir el modelo de arquitectura física y lógica en base a los requerimientos del usuario utilizando el Modelamiento de desarrollo de software dirigido por ICONIX.

Para dar cumplimiento al primer objetivo del presente TT, se llevaron a cabo dos fases: El análisis y diseño preliminar y el Diseño de la arquitectura física y lógica, detallados en las secciones 6.1.1 y 6.1.2 respectivamente.

6.1.1. Análisis y diseño preliminar.

Para la fase de análisis y diseño preliminar se realizó una entrevista con el cliente, en este caso el gerente-propietario de la empresa Compumars, el Ing. Mario Enrique Cueva Hurtado, con la finalidad de explorar más acerca de la gestión y comercialización de productos y servicios para Compumars, lo que fue de gran ayuda para desarrollar el modelado de requerimientos y diagrama del dominio.

En la Tabla 1 se presentan algunas de las preguntas más relevantes extraídas de la entrevista realizada al gerente-propietario de Compumars. El desarrollo completo de la entrevista se encuentra en el **Anexo 1. Entrevista para la obtención de requerimientos.**

Tabla 1. Preguntas relevantes de la entrevista para la obtención de requerimientos

Pregunta	Detalle
¿Quiénes serán los usuarios del sistema?	Con esta pregunta se obtuvo que los usuarios del sistema son: Administrador, Cajero, Bodeguero, Operador y Cliente.
¿Cuáles son las responsabilidades de cada usuario?	Con esta pregunta se obtuvo que las responsabilidades de cada rol son: Administrador: es el encargado de crear cuentas de usuario, cambiar roles y dar de baja cuentas. Bodeguero: es el encargado de registrar actualizaciones en los productos, ya sea registrar un nuevo producto, modificar un producto existente, asignar el inventario de cada producto o dar de baja un producto existente. Cajero: se encarga de registrar las ventas en el local, registrar el pago y el comprobante del mismo.

	Operador: se encarga de dar seguimiento a las órdenes realizadas en línea por el cliente, verificar el pago y realizar el comprobante de venta asociada a la orden.
¿Qué datos se debe registrar de los productos?	Se obtuvo que de los productos se debe registrar: código, nombre, marca, modelo, color, descripción, tamaño, foto, precio, código auxiliar, categoría, código de barras.
¿Cómo se debe gestionar el inventario?	Se obtuvo para gestionar el inventario se debe registrar la entrada de nuevos productos al almacén mediante las órdenes de compra y, por otro lado, para las salidas de productos las órdenes de venta llevan a cabo este registro. En sí, estas órdenes son las que se encargan de actualizar la disponibilidad de los productos. Si al registrar una entrada el producto no está registrado en la lista, se debe primero registrar el producto y después actualizar la disponibilidad
¿Cómo se debe generar una orden de venta?	Para esta pregunta se obtuvo que existen dos caminos: las órdenes en línea (realizadas por el cliente mediante el sitio web) y las ventas en el local (el cliente compra y el Cajero registra la respectiva orden).
¿Cómo se debe verificar los pagos?	Se obtuvo que existen dos maneras: el cajero registra inmediatamente el pago (para ventas en el local), y el cliente registra la constancia del pago (para órdenes en línea). Para esta última, el rol de operador es quien posteriormente valida que el pago registrado sea correcto.
¿Cómo se debe gestionar los servicios?	Para esta pregunta se obtuvo que los servicios se deben registrar como un tipo de producto más dentro del sistema, pero estos no tienen un registro de existencias dentro del inventario.

6.1.1.1. Modelado de requerimientos.

En este punto, una vez realizada la entrevista, se procedió a desarrollar el modelado de los requerimientos funcionales y no funcionales de la solución informática. Para empezar, se estableció el glosario de términos para dar a conocer los principales conceptos que el sistema debe ser capaz de administrar. En la Tabla 2 se puede evidenciar estos conceptos.

Tabla 2. Glosario de términos

Término	Concepto
Administrador	Rol encargado de la gestión de usuarios con sus respectivos roles (cajero, bodeguero, operador y administrador)

Cliente	Rol que tiene la facultad de navegar por las categorías de productos, realizar órdenes en línea y gestionar su propia cuenta de usuario. Asimismo, puede registrar su propia cuenta de usuario
Cajero	Rol encargado de registrar ventas inmediatas (desde la empresa misma) sin previo pedido desde el servicio en línea.
Bodeguero	Rol encargado de registrar actualizaciones en los productos, ya sea registrar un nuevo producto, modificar un producto existente, asignar el inventario de cada producto o dar de baja un producto existente
Operador	Rol encargado de dar seguimiento a las órdenes en línea, así como validar pagos realizados por clientes y registrar el comprobante de venta asociado a la orden cuando el pago ha sido validado con éxito
Inventario	Es el registro de la información de los productos que existen en bodega, en donde se registran datos como: cantidad disponible en el inventario, cantidad reservada por una orden, cantidad física existente, precio de venta del producto, impuesto del producto y la información del producto asociado.
Producto	Es un artículo específico que posee la empresa, el cual va a ser comercializado
Orden de venta	Es un tipo de orden realizada para los clientes y representa la salida de productos que existen en bodega
Orden de compra	Es un tipo de orden que representa la entrada de productos en bodega
Pago	Es una constancia que representa la cancelación de una orden, este pago puede ser por depósito o transferencia bancaria o efectivo y posee una referencia que indica la constancia del pago, la cual debe ser validada por el rol operador
Servicio	Un servicio para la empresa es un tipo de producto más, pero que no posee un registro en el inventario

A continuación, en la Tabla 3 se muestra algunos de los requerimientos funcionales más relevantes de la solución informática, la lista completa se encuentra en el **Anexo 3. Sección 1.2. Requerimientos Funcionales.**

Tabla 3. Requerimientos funcionales más relevantes

Código	Descripción
RF01	El sistema permitirá a todos los usuarios acceder a su cuenta de usuario mediante sus credenciales como dirección de correo electrónico y clave
RF05	El sistema permitirá al administrador crear nuevas cuentas de usuario con la información de: nombre, apellido, dirección de correo electrónico y clave

RF06	El sistema permitirá al administrador gestionar (agregar, modificar) los siguientes roles (administrador, cajero, bodeguero y operador) a las cuentas de usuario
RF08	El sistema permitirá al administrador dar de baja a cuentas de usuario
RF09	El sistema permitirá al cliente registrar su cuenta de usuario. Su cuenta de usuario se registrará por defecto con el rol de Cliente.
RF10	El sistema permitirá al cliente buscar productos a través de categorías
RF11	El sistema permitirá al cliente agregar productos al carrito de compras
RF14	El sistema permitirá al cliente realizar órdenes de compra en línea (para el sistema será representado a través de órdenes de venta) a través del carrito de compras
RF15	El sistema permitirá al cliente registrar el pago para las órdenes de compra
RF17	El sistema permitirá al cajero registrar órdenes por concepto de venta
RF19	El sistema permitirá al bodeguero registrar productos
RF23	El sistema permitirá al bodeguero modificar detalles de los productos
RF24	El sistema permitirá al bodeguero dar de baja productos
RF25	El sistema permitirá al bodeguero registrar órdenes por concepto de compra
RF27	El sistema permitirá al operador buscar órdenes de compra (para el sistema será representado a través de órdenes de venta) realizadas en línea por el cliente
RF29	El sistema permitirá al operador validar pagos de órdenes de compra realizadas en línea por el cliente
RF30	El sistema permitirá al operador generar el comprobante de venta cuando un pago de la orden de compra ha sido validado.

En la Tabla 4 se expone la lista de requerimientos no funcionales correspondientes a la solución informática, en donde el sistema deberá:

Tabla 4. Requerimientos no funcionales

Código	Descripción	Categoría
RNF01	Proveer tiempos de respuesta accesibles (entre 2 a 5 segundos) por cada petición u operación	Tiempos de respuesta
RNF02	Permitir que cada rol de usuario tenga sus propios privilegios de acceso a las distintas funcionalidades de la aplicación	Autorización
RNF03	Encriptar la clave de cada usuario antes de su almacenamiento en la base de datos	Seguridad
RNF04	Estar disponible las 24 horas de cada día	Disponibilidad
RNF05	Poseer la facultad de ser utilizada en cualquier navegador web. Y gracias a las características de las PWA también podrá ser instalada y usada en sistemas operativos móviles IOS o Android.	Tipo de aplicación

RNF06	Tener una interfaz amigable e intuitiva para facilitar las tareas del usuario	Interfaz gráfica
-------	-------------------------------------------------------------------------------	------------------

6.1.1.2. Modelado del diagrama del dominio.

Para dar cumplimiento a la primera fase del Análisis y Diseño preliminar se realizó el modelado del diagrama del dominio, con esto se puede evidenciar cada uno de los términos utilizados en el proyecto y las relaciones que existen entre ellos. Con el propósito de dar a conocer todos los conceptos de una manera más entendible, se realizó tres diagramas del dominio expuestos a continuación: en la Figura 2 se muestra el diagrama del dominio correspondiente a los usuarios, por otro lado, en la Figura 3 se presentan los principales conceptos que intervienen en las órdenes realizadas por el rol cliente. Finalmente, en la Figura 4 se expone el diagrama del dominio con relación al inventario.

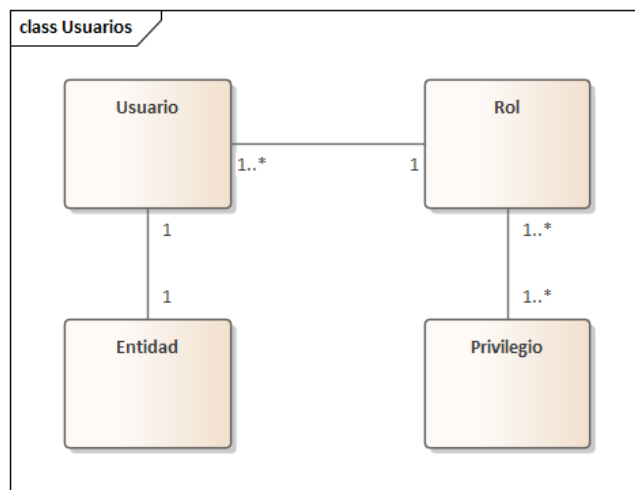


Figura 2. Diagrama del dominio de Usuarios

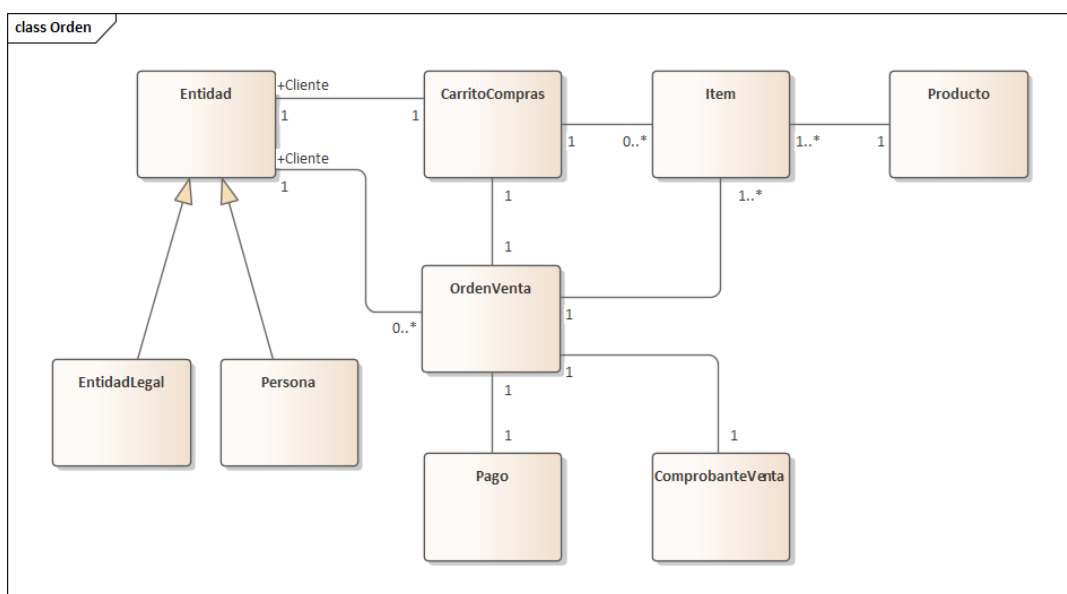


Figura 3. Diagrama del dominio de Órdenes realizadas por el cliente

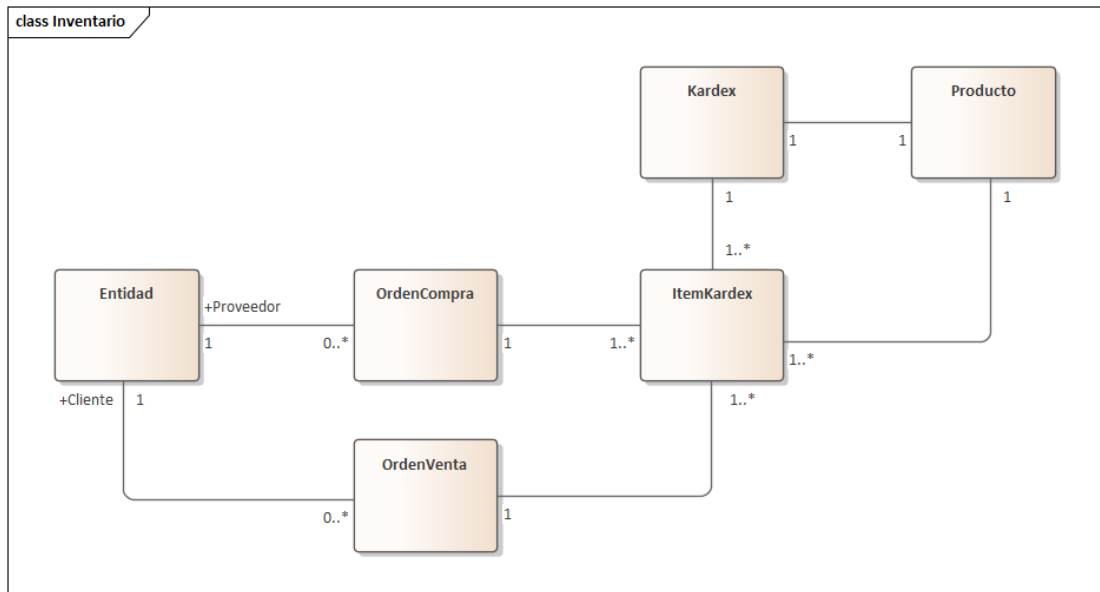


Figura 4. Diagrama del dominio de Inventario

6.1.2. Diseño de la arquitectura física y lógica.

En esta fase se desarrolló todas las tareas necesarias para llevar a cabo el Diseño de la arquitectura física y lógica. Para la ejecución de este apartado, primeramente, se realizó el modelado de casos de uso para seguidamente dar paso al modelado de los diagramas de secuencia por cada caso de uso descubierto. Estos dos modelados permitieron la elaboración del diagrama de paquetes y clases correspondiente. Y, por último, se culmina con el diseño la arquitectura física de la solución informática.

6.1.2.1. Modelado de casos de uso.

Para el modelado de casos de uso se empezó identificando los actores que intervienen en el sistema, los cuales son: Administrador, Cliente, Cajero, Operador y Bodeguero. Cada uno de estos actores poseen distintas funcionalidades, las mismas que pueden ser evidenciadas en la Figura 5, en donde se muestra el diagrama de casos de uso general del sistema para la gestión y comercialización de productos y servicios de Compumars.

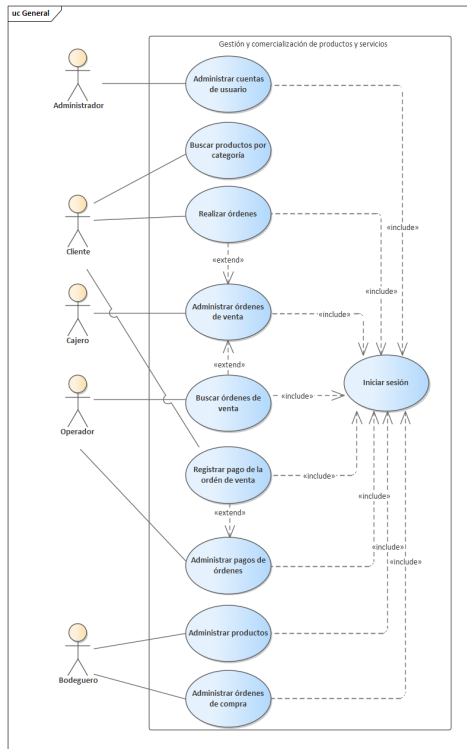


Figura 5. Diagrama de casos de uso General

En el diagrama de casos de uso general de la Figura 5 se puede observar las distintas actividades de administración que posee cada actor del sistema. Por ello, con el fin de mostrar a detalle que otras funcionalidades compone cada uno de estos casos de uso generales, se desarrollaron los diagramas de caso de uso explotados, los mismos que son: Administración de Cuentas de usuario (ver Figura 6), Administración de Órdenes de venta (ver Figura 7), Administración de Pagos de órdenes (ver Figura 8), Administración de Productos (ver Figura 9), y Administración de Órdenes de compra (ver Figura 10).

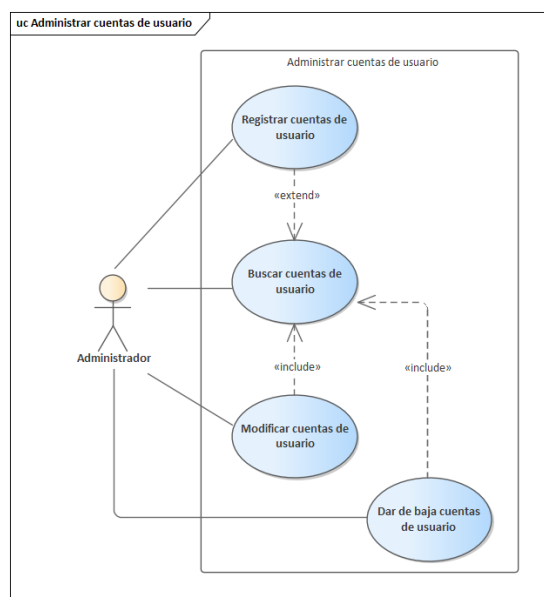


Figura 6. Diagrama de casos de uso de Administrar cuentas de usuario

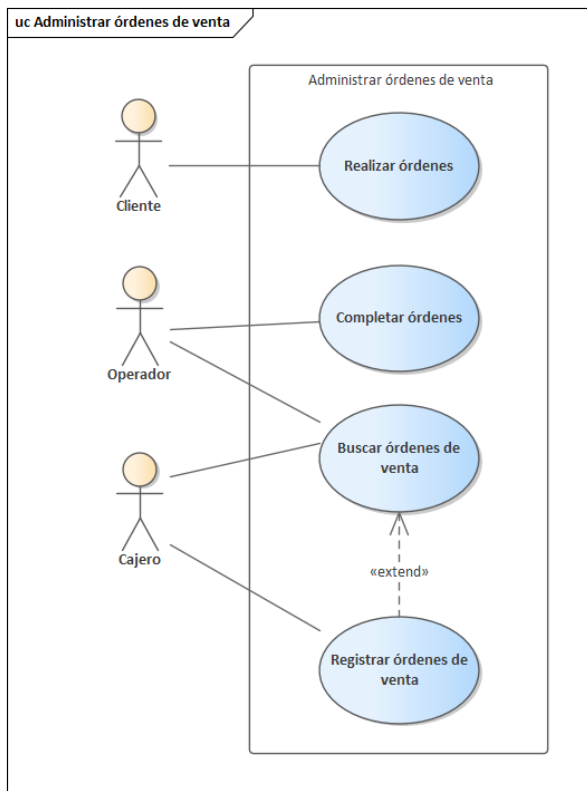


Figura 7. Diagrama de casos de uso de Administrar órdenes de venta

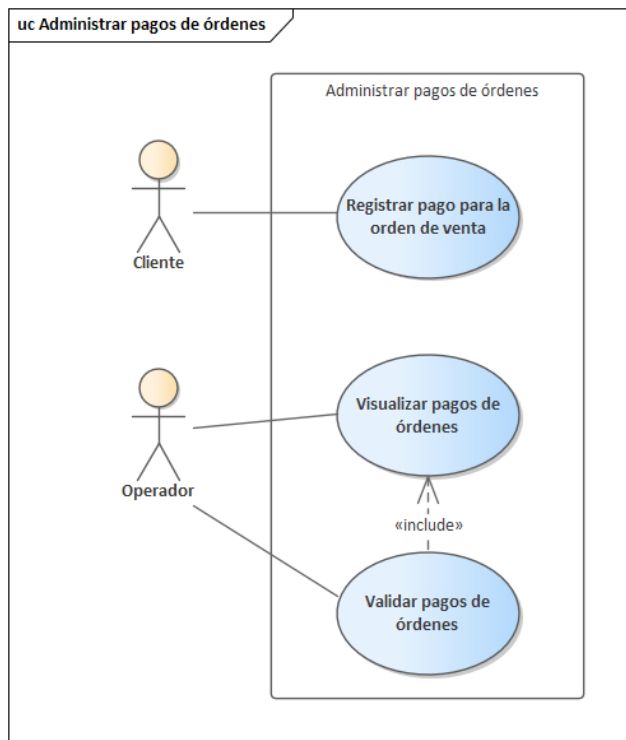


Figura 8. Diagrama de casos de uso de Administrar pagos de órdenes

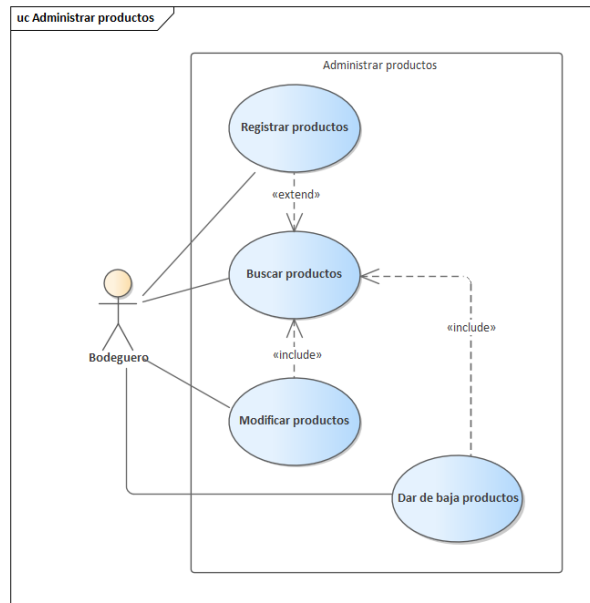


Figura 9. Diagrama de casos de uso de Administrar productos

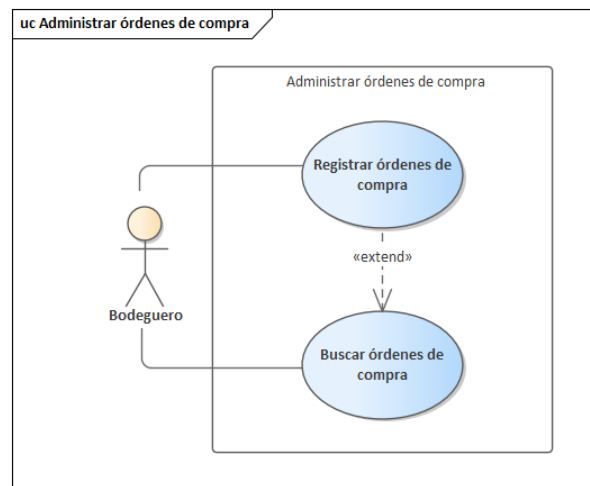


Figura 10. Diagrama de casos de uso de Administrar órdenes de compra

Especificación de casos de uso

Tomando en cuenta los casos de uso explotados presentados anteriormente, se procedió a realizar una descripción de los mismos, para ello se tomó como ejemplo algunos de los casos de uso más relevantes. La especificación de casos de uso completa se puede ver en el **Anexo 3. Sección 3.3. Especificación de casos de uso.**

Antes de nada, cabe recalcar que, para la especificación de casos de uso, la metodología ICONIX recomienda (de ser necesario) el uso de *Wireframes* o prototipos que permitan tener una mejor comprensión y una visión más amplia del flujo que puede tomar cada caso de uso.

En la Tabla 5 se encuentra la especificación del caso de uso para el Inicio de sesión de todos los actores del sistema.

Tabla 5. Especificación del caso de uso Iniciar sesión

Identificador	CU01
Nombre	Iniciar Sesión
Actores	Administrador, Cliente, Cajero, Operador, Bodeguero
Descripción	Autenticación del usuario en el sistema
Precondiciones	<ul style="list-style-type: none"> a. Haberse registrado en el sistema b. Encontrarse en la página "Iniciar Sesión"
Flujo Principal	<ul style="list-style-type: none"> 1. El sistema muestra el formulario de Iniciar sesión 2. El usuario ingresa sus credenciales de acceso (dirección de correo electrónico y clave) 3. El usuario pulsa el botón "Iniciar sesión" del formulario 4. El sistema valida la información ingresada en el formulario 5. El sistema autentica al usuario y carga su información y funcionalidades 6. El caso de uso finaliza
Flujo Alternativo	<ul style="list-style-type: none"> a. Campos obligatorios vacíos 4.1. El sistema debe mostrar un mensaje indicando que existen campos obligatorios vacíos b. Credenciales de acceso no válidas 5.1. El sistema debe mostrar un mensaje que dice: Credenciales de acceso no válidas

En la Tabla 6 se halla la especificación del caso de uso para el registro de cuentas de usuario por parte del Administrador.

Tabla 6. Especificación del caso de uso Registrar cuentas de usuario

Identificador	CU04
Nombre	Registrar cuentas de usuario
Actores	Administrador
Descripción	El administrador registra un nuevo usuario
Precondiciones	<ul style="list-style-type: none"> a. Administrador autenticado b. Encontrarse en la página Cuentas de usuario c. Es una extensión del caso de uso Buscar cuentas de usuario
Flujo Principal	<ul style="list-style-type: none"> 1. El administrador pulsa el botón "Registrar cuenta de usuario" de la página Cuentas de usuario 2. El sistema muestra el formulario de nueva cuenta de usuario 3. El administrador registra los datos de la cuenta de usuario 4. El administrador le asigna un rol a la cuenta de usuario 5. El administrador presiona el botón Guardar del formulario 6. El sistema valida la información ingresada en el formulario 7. El sistema verifica que la dirección de correo electrónico del usuario ingresado no existe 8. El sistema crea el nuevo usuario y muestra un mensaje indicando: Usuario creado correctamente 9. El sistema redirige al administrador a la página "Cuentas de usuario"

	10. El caso de uso finaliza
Flujo Alternativo	<p>a. Campos obligatorios vacíos 6.1. El sistema debe mostrar un mensaje indicando que existen campos obligatorios vacíos</p> <p>b. Cuenta de usuario ya existe 7.1. El sistema debe mostrar un mensaje indicando que la cuenta de usuario ingresada ya se encuentra registrada</p>

En la Figura 11 se presenta el *wireframe* relacionado al caso de uso Realizar orden, que es ejecutado por el Cliente, el mismo que se encuentra especificado en la Tabla 7.

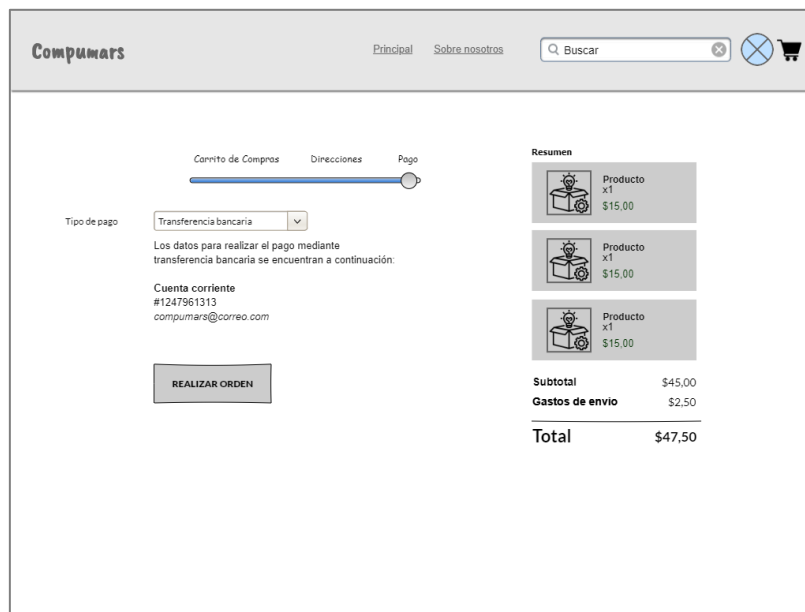


Figura 11. *Wireframe* de Realizar orden

Tabla 7. Especificación de caso de uso Realizar órdenes

Identificador	CU07
Nombre	Realizar órdenes
Actores	Cliente
Descripción	El cliente realiza una nueva orden en base a los productos agregados en su carrito de compras
Precondiciones	a. Cliente autenticado
Flujo Principal	<ol style="list-style-type: none"> 1. El cliente añade productos al carrito de compras 2. El cliente pulsa el botón Ver carrito de compras del menú de navegación 3. El sistema muestra el listado de todos los productos añadidos al carrito de compras 4. El sistema muestra el formulario de agregar una dirección de envío y una dirección para el comprobante de venta. 5. El cliente registra los datos de dirección de envío y dirección de comprobante de venta 6. El sistema muestra información acerca del método de pago 7. El cliente presiona el botón Realizar orden

	8. El sistema valida la información ingresada en el formulario 9. El sistema crea una nueva orden de venta y muestra un mensaje indicando: Orden creada correctamente 10. El sistema envía la confirmación de la orden al correo electrónico del cliente 11. El sistema muestra los detalles de la orden registrada 12. El caso de uso finaliza
Flujo Alternativo	a. Carrito de compras vacío 7.1. El sistema debe deshabilitar el botón Realizar orden ya que no existen productos en el carrito de compras b. Campos obligatorios vacíos 8.1. El sistema debe mostrar un mensaje indicando que existen campos obligatorios vacíos

A continuación, en la Figura 12 se presenta el *wireframe* relacionado al registro de órdenes de venta que realiza el actor Cajero, el caso de uso se encuentra especificado en la Tabla 8.

Compumars

Principal > Nueva Orden de venta

Nueva Orden de venta

Cliente: o

Buscar un producto:

Código	Producto	Precio Unitario	Cantidad	Precio	Acción
###	Producto AAA	\$10.00	12	\$120.00	Quitar
###	Producto AAA	\$10.00	12	\$120.00	Quitar
###	Producto AAA	\$10.00	12	\$120.00	Quitar
				Subtotal	\$380.00
				Costos de envío	\$0.00
				IVA	\$20.00
				Total a pagar	\$400.00

Dirección de envío: o

Dirección de comprobante: o

Tipo de pago: Estado de la orden:

Tipo de comprobante: N° de Comprobante:

Series: Fecha de emisión:

Figura 12. *Wireframe* de Registrar órdenes de venta

Tabla 8. Especificación de caso de uso Registrar órdenes de venta

Identificador	CU09
Nombre	Registrar órdenes de venta
Actores	Cajero
Descripción	El cajero registra una nueva orden de venta
Precondiciones	a. Cajero autenticado b. Encontrarse en la página Órdenes de venta c. Es una extensión del caso de uso Buscar órdenes de venta

Flujo Principal	<ol style="list-style-type: none"> 1. El cajero pulsa el botón “Registrar orden de venta” de la página Órdenes de venta 2. El sistema muestra el formulario de nueva orden de venta 3. El cajero registra los datos de la orden de venta (datos del cliente, dirección de envío y comprobante de venta, lista de productos, información del pago) 4. El cajero presiona el botón Guardar del formulario 5. El sistema valida la información ingresada en el formulario 6. El sistema crea una nueva orden de venta y muestra un mensaje indicando: Orden creada correctamente 7. El sistema envía la confirmación de la orden al correo electrónico del cliente 8. El sistema redirige al cajero a la página “Órdenes de venta” 9. El caso de uso finaliza
Flujo Alterno	<p>a. Campos obligatorios vacíos</p> <ol style="list-style-type: none"> 5.1. El sistema debe mostrar un mensaje indicando que existen campos obligatorios vacíos

Prosiguiendo con el caso de uso correspondiente al registro de la constancia de pago que realiza el Cliente por una orden realizada previamente, la especificación respectiva al caso de uso mencionado se encuentra a continuación en la Tabla 9.

Tabla 9. Especificación de caso de uso Registrar pago para la orden de venta

Identificador	CU11
Nombre	Registrar pago para la orden de venta
Actores	Cliente
Descripción	El cliente registra el comprobante de pago para una orden realizada previamente
Precondiciones	<ol style="list-style-type: none"> a. Cliente autenticado b. Haber realizado una orden previamente
Flujo Principal	<ol style="list-style-type: none"> 1. El cliente pulsa el botón Ver órdenes del menú de navegación 2. El sistema muestra el listado de todas las órdenes realizadas por el cliente y marcará aquellas que requieran registrar el pago 3. El cliente selecciona una orden del listado 4. El sistema muestra información acerca de la orden y el formulario de registrar la referencia del pago de la orden 5. El cliente registra los datos del pago de la orden 6. El cliente presiona el botón Guardar del formulario 7. El sistema valida la información ingresada en el formulario 8. El sistema registra el pago de la orden de venta y muestra un mensaje indicando: Pago registrado correctamente 9. El caso de uso finaliza
Flujo Alterno	<p>a. Pago ya registrado</p> <ol style="list-style-type: none"> 5.1. El sistema no mostrará el formulario para registrar la referencia del pago de la orden <p>b. Campos obligatorios vacíos</p> <ol style="list-style-type: none"> 7.1. El sistema debe mostrar un mensaje indicando que existen campos obligatorios vacíos

En la Figura 13 se muestra el *wireframe* relacionado a la validación que realiza el actor Operador sobre la constancia del pago que ha registrado un cliente por una orden. En la Tabla 10 se puede observar la especificación relacionada a este caso de uso.

Figura 13. *Wireframe* de Validar pagos de órdenes

Tabla 10. Especificación de caso de uso Validar pagos de órdenes

Identificador	CU13
Nombre	Validar pagos de órdenes
Actores	Operador
Descripción	El operador valida el pago registrado de una orden
Precondiciones	<ul style="list-style-type: none"> a. Operador autenticado b. Encontrarse en la página Ver detalles de una orden c. Incluye el caso de uso Visualizar pagos de órdenes
Flujo Principal	<ol style="list-style-type: none"> 1. El operador verifica que el pago registrado por el cliente es correcto 2. El operador actualiza el estado de la orden a Procesando 3. El sistema muestra el formulario de registro del comprobante de venta asociado a la orden 4. El operador registra los datos del comprobante de venta (Tipo de documento, número de comprobante, fecha de emisión, número de serie de los productos de la orden) 5. El operador presiona el botón Guardar del formulario 6. El sistema valida la información ingresada en el formulario 7. El sistema establece el estado de la orden en Procesando y muestra un mensaje indicando: Orden modificada correctamente 8. El sistema crea el comprobante de venta de la orden y muestra un mensaje indicando: Comprobante de venta creado correctamente 9. El caso de uso finaliza
Flujo Alternativo	<ol style="list-style-type: none"> a. Pago no válido

	2.1. El operador actualiza el estado de la orden a Cancelada 2.2. El sistema establece el estado de la orden en Cancelada y muestra un mensaje indicando: Orden cancelada b. Campos obligatorios vacíos 6.1. El sistema debe mostrar un mensaje indicando que existen campos obligatorios vacíos
--	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Siguiendo con la especificación de casos de uso, en la Tabla 11 se sitúa la especificación del caso de uso para el registro de productos, mismo que lo efectúa el actor Bodeguero.

Tabla 11. Especificación de caso de uso Registrar productos

Identificador	CU15
Nombre	Registrar productos
Actores	Bodeguero
Descripción	El bodeguero registra un nuevo producto
Precondiciones	a. Bodeguero autenticado b. Encontrarse en la página Productos c. Es una extensión del caso de uso Buscar productos
Flujo Principal	1. El bodeguero pulsa el botón “Registrar producto” de la página Productos 2. El sistema muestra el formulario de nuevo producto 3. El bodeguero registra los datos del producto 4. El bodeguero le asigna una categoría al producto 5. El bodeguero presiona el botón Guardar del formulario 6. El sistema valida la información ingresada en el formulario 7. El sistema verifica que el código del producto ingresado no existe 8. El sistema crea el nuevo producto y muestra un mensaje indicando: Producto creado correctamente 9. El sistema redirige al bodeguero a la página Productos 10. El caso de uso finaliza
Flujo Alternativo	a. Campos obligatorios vacíos 6.1. El sistema debe mostrar un mensaje indicando que existen campos obligatorios vacíos b. Producto ya existe 7.1. El sistema debe mostrar un mensaje indicando que el producto ya se encuentra registrado

Culminando con la especificación de los casos de uso más relevantes, en la Figura 14 se encuentra el *wireframe* asociado al caso de uso de registrar órdenes de compra por parte del actor Bodeguero, el mismo que está especificado en la Tabla 12.

Figura 14. Wireframe de Registrar órdenes de compra

Tabla 12. Especificación de caso de uso Registrar órdenes de compra

Identificador	CU19
Nombre	Registrar órdenes de compra
Actores	Bodeguero
Descripción	El bodeguero registra una nueva orden de compra
Precondiciones	<ol style="list-style-type: none"> Bodeguero autenticado Encontrarse en la página Órdenes de compra Es una extensión del caso de uso Buscar órdenes de compra
Flujo Principal	<ol style="list-style-type: none"> El bodeguero pulsa el botón “Registrar orden de compra” de la página Órdenes de compra El sistema muestra el formulario de nueva orden de compra El cajero registra los datos de la orden de compra (datos del proveedor, datos del comprobante de venta, lista de productos) El cajero presiona el botón Guardar del formulario El sistema valida la información ingresada en el formulario El sistema crea una nueva orden de compra y muestra un mensaje indicando: Orden creada correctamente El sistema actualiza el stock de los productos asociados a la orden de compra El sistema redirige al cajero a la página “Órdenes de compra” El caso de uso finaliza
Flujo Alternativo	<ol style="list-style-type: none"> Campos obligatorios vacíos <ol style="list-style-type: none"> El sistema debe mostrar un mensaje indicando que existen campos obligatorios vacíos

6.1.2.2. Modelado de interacción mediante diagrama de secuencia.

En este apartado se desarrolló el modelo dinámico de la solución informática, es decir se realizó diagramas de secuencia con el fin de descubrir el comportamiento que tendrán cada una de las clases que resulten del modelado del diagrama del dominio, y también para tener una visión más amplia de como interactúan cada una de las entidades del sistema. Para ello, igualmente teniendo en cuenta que se presentó la especificación de casos de uso más relevantes en la sección anterior, se muestra a continuación los diagramas de secuencia relacionados a estos casos de uso especificados. La lista completa de diagramas de secuencia se puede observar en el **Anexo 3. Sección 4. Diagramas de Secuencia**.

A continuación, se exponen los diagramas de secuencia más relevantes correspondientes a: Inicio de sesión (ver Figura 15), Registrar cuentas de usuario (ver Figura 16), Realizar orden (ver Figura 17), Registrar órdenes de venta (ver Figura 18), Registrar pago para la orden de venta (ver Figura 19), Validar pagos de órdenes (ver Figura 20), Registrar Productos (ver Figura 21) y Registrar órdenes de compra (ver Figura 22).

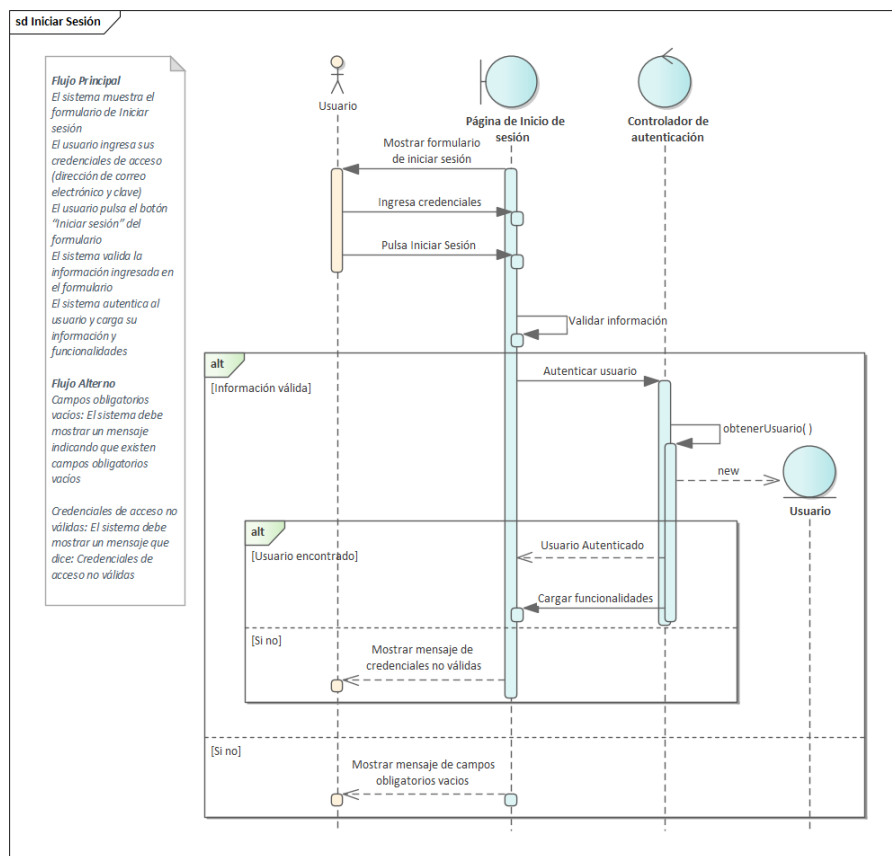


Figura 15. Diagrama de secuencia de Iniciar sesión

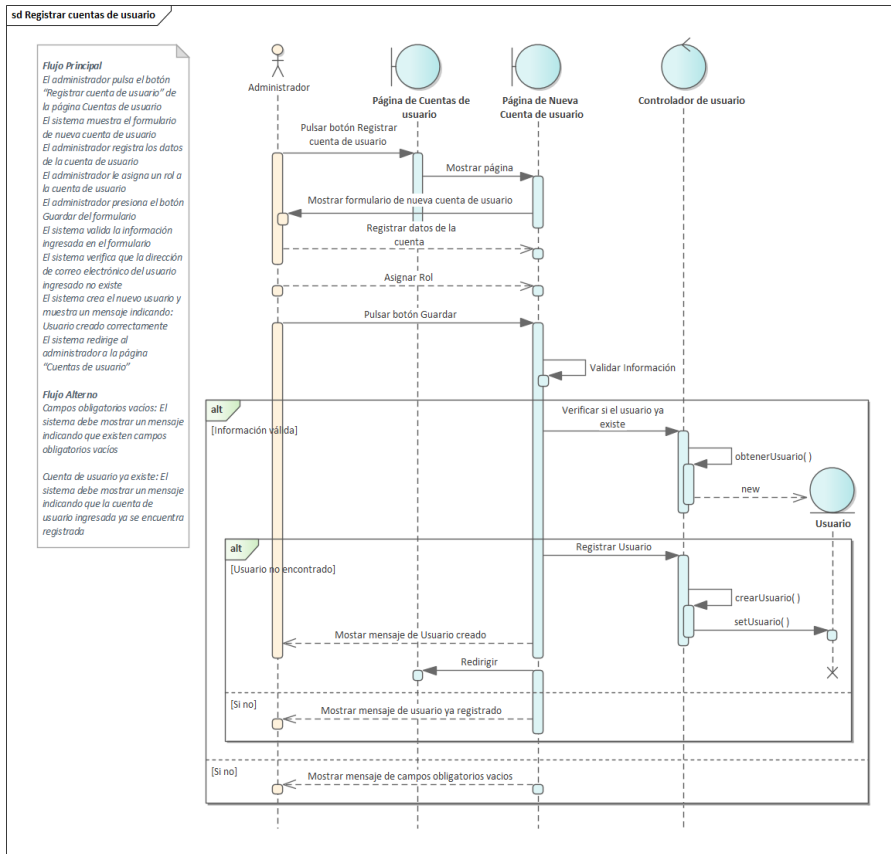


Figura 16. Diagrama de secuencia de Registrar cuentas de usuario

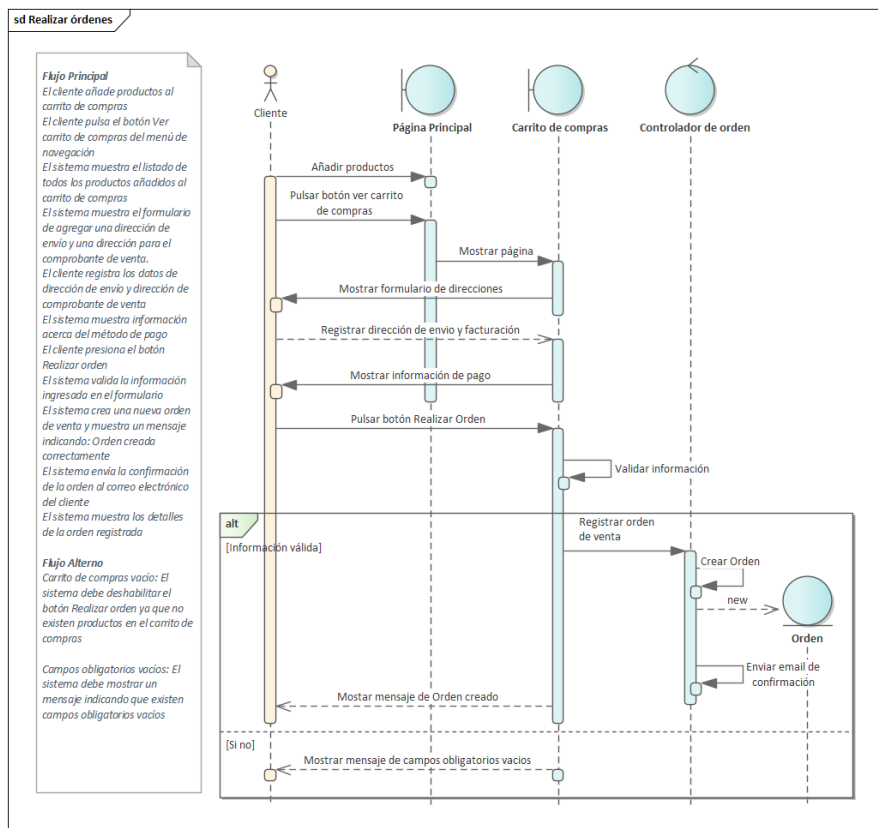


Figura 17. Realizar orden

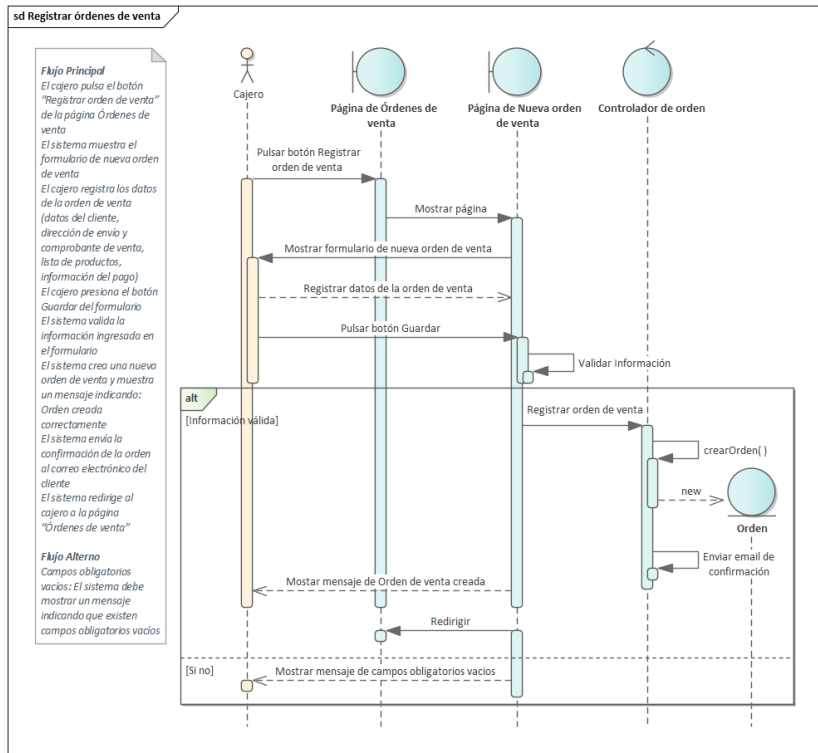


Figura 18. Registrar órdenes de venta

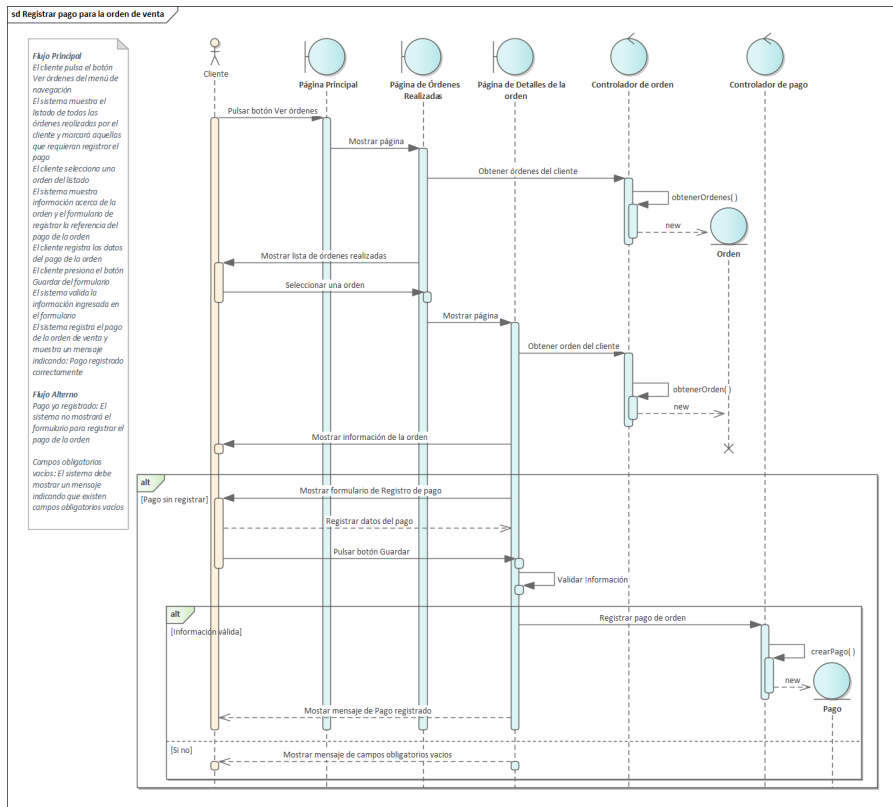


Figura 19. Registrar pago para la orden de venta

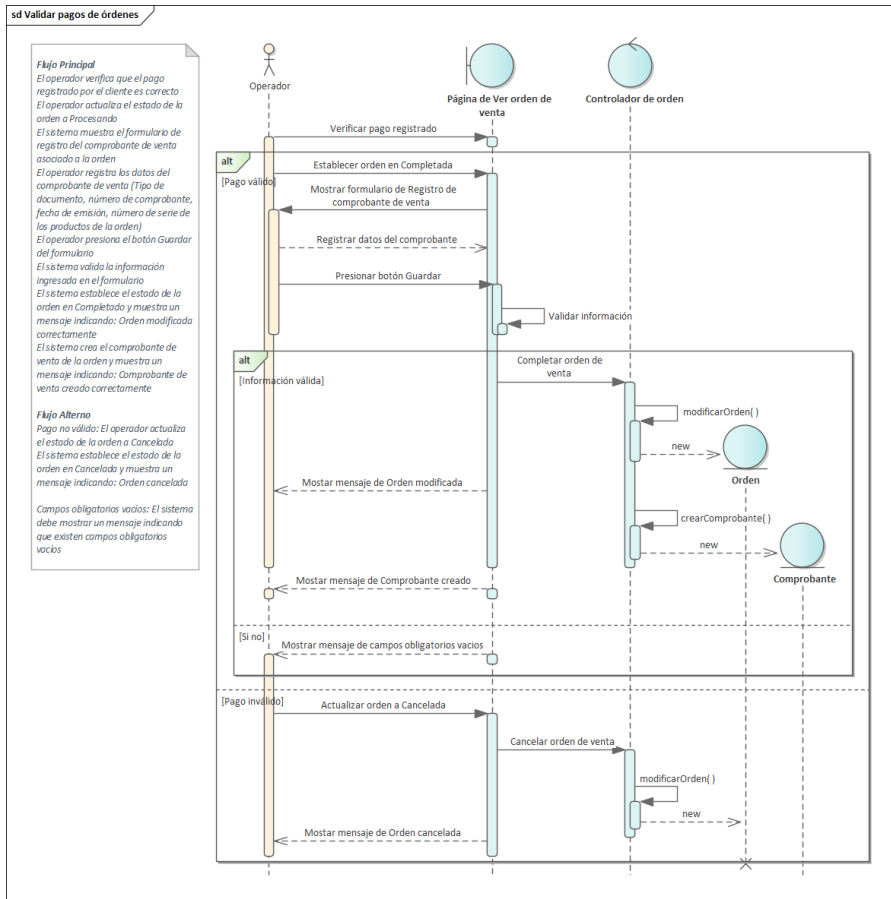


Figura 20. Validar pagos de órdenes

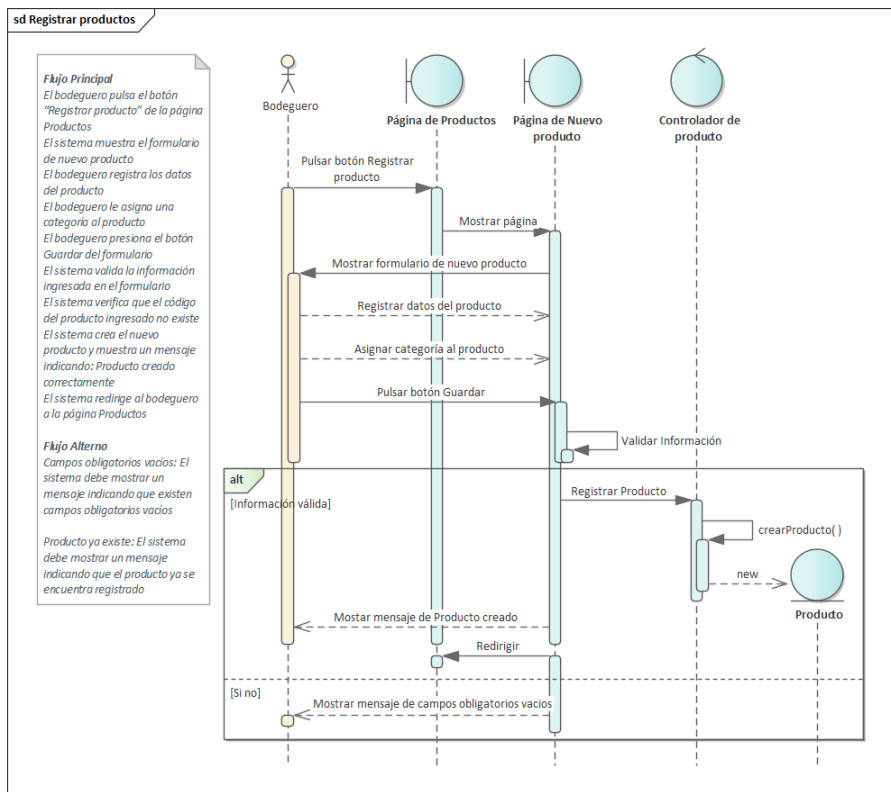


Figura 21. Registrar Productos

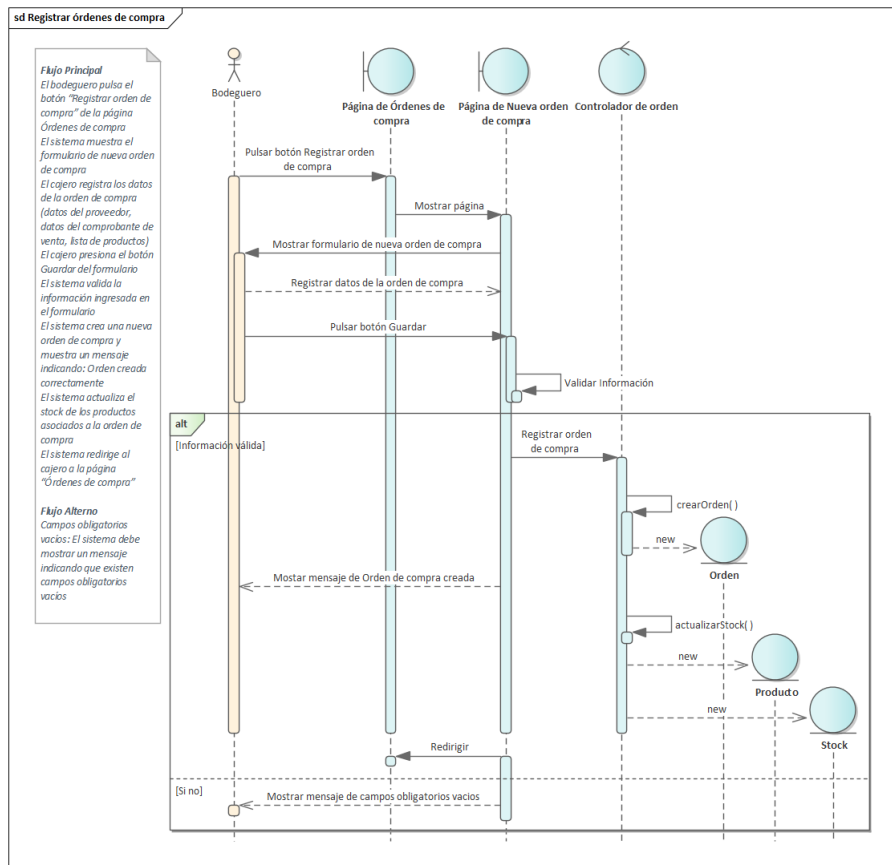


Figura 22. Registrar órdenes de compra

6.1.2.3. Modelado estructural mediante diagrama de paquetes y diagrama de clases.

En este apartado se construyó el diagrama de paquetes y clases correspondiente a la solución informática para la gestión y comercialización de productos y servicios de la empresa Compumars, los cuales son de demasiada importancia en todo este proceso puesto a que en esta sección se abarcan aspectos lógicos y de estructura del proyecto.

Primeramente, se realizó el diagrama de paquetes con el objetivo de proporcionar una organización visual de la arquitectura MVC (Modelo-Vista-Controlador) empleada. En la Figura 23 se puede observar el diagrama de paquetes elaborado para este propósito.

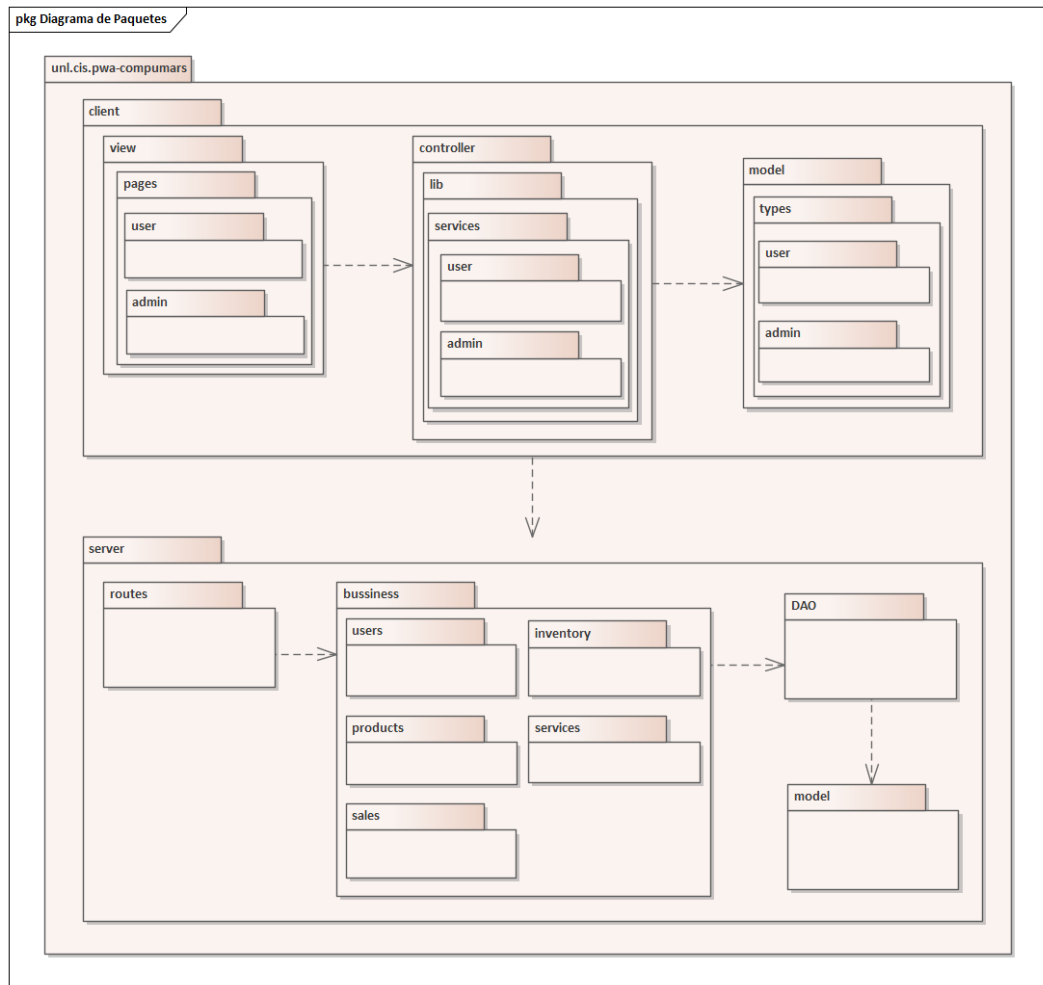


Figura 23. Diagrama de paquetes

La solución informática se organiza en dos paquetes contenedores: Cliente (*client*) y Servidor (*server*). El paquete Cliente se compone de tres conceptos: la vista (*view*) representada por el conjunto de páginas de usuario y administración, el controlador (*controller*) representada por los servicios que comunican al cliente con el servidor, y finalmente con el modelo (*model*) representada por las unidades de datos (*types*). Por otro lado, el paquete Servidor contiene: las rutas (*routes*) que son la puerta de salida y comunicación con el cliente, la Lógica de negocio (*bussiness*) que está representada por cada uno de los módulos de gestión (Usuarios, Inventario, Productos, Servicios y Ventas), la Capa de Acceso a Datos (DAO por sus siglas en inglés) que representa la fachada sobre la base de datos y que contiene las funciones para no entregar responsabilidades a los modelos del sistema, por último, los modelos (*model*) que son las clases extraídas del modelo del dominio.

Asimismo, se construyó el diagrama de clases, el mismo que se encuentra en la Figura 24 y que fue elaborándose con sus atributos y operaciones a medida que se desarrollaron los casos de uso y diagramas de secuencia correspondientes. De este diagrama se puede destacar la clase de Usuario (*User*) que representa a todos los usuarios con acceso al sistema,

la clase Orden (*Order*) que representa las ventas y compras de la empresa, la clase Comprobante de venta (*Receipt*) que representa el comprobante de venta asociado a la orden, la clase Pago (*Payment*) que representa la constancia de pago registrada para la orden, la clase Producto (*Product*) que representa todos los productos y servicios de Compumars, la clase Stock que representa las existencias de cada producto, y la clase Kardex que representa el precio de venta de cada producto registrado.

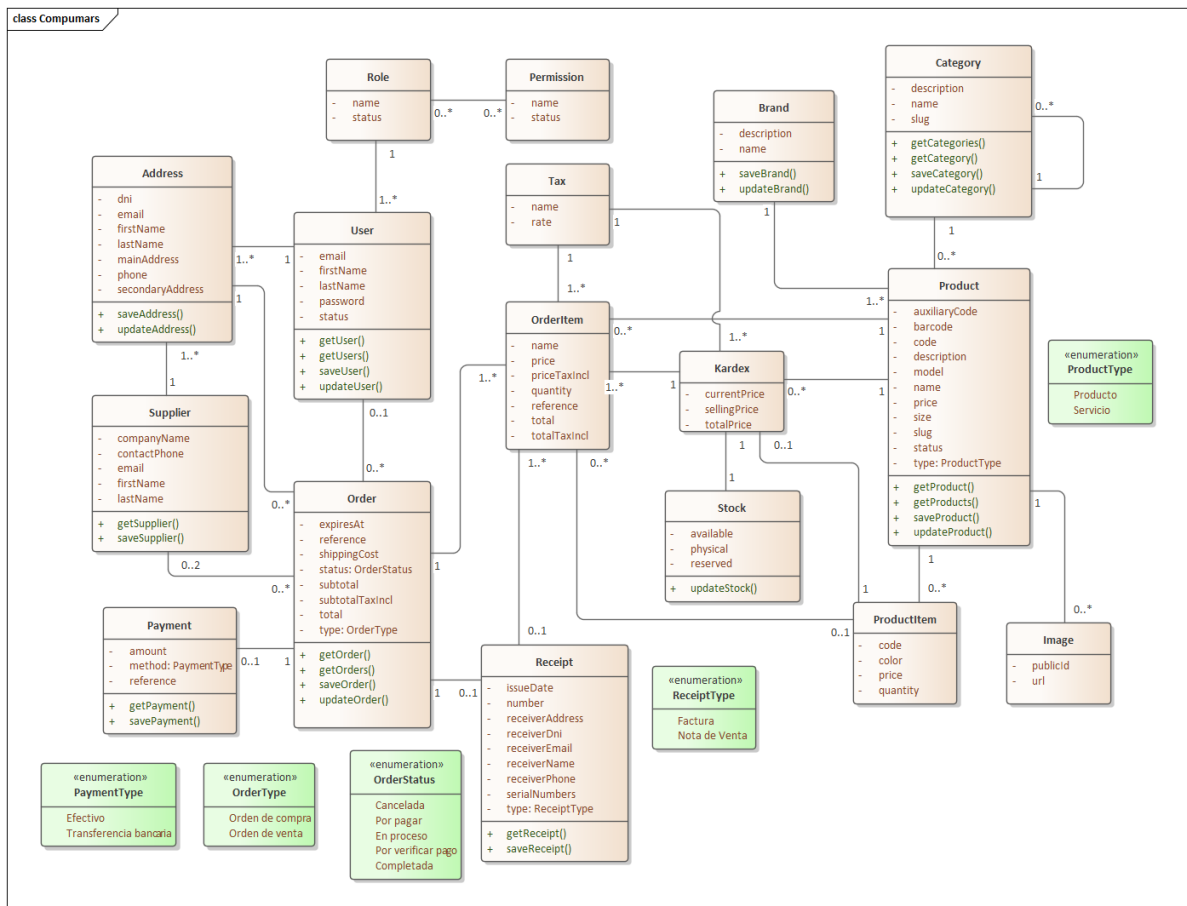


Figura 24. Diagrama de Clases

6.1.2.4. Diseño de Arquitectura física.

Para completar el diseño de la arquitectura física y lógica, en este apartado primeramente se desarrolló el diagrama de componentes (ver Figura 25) para ilustrar las piezas de software que conforman la solución informática.

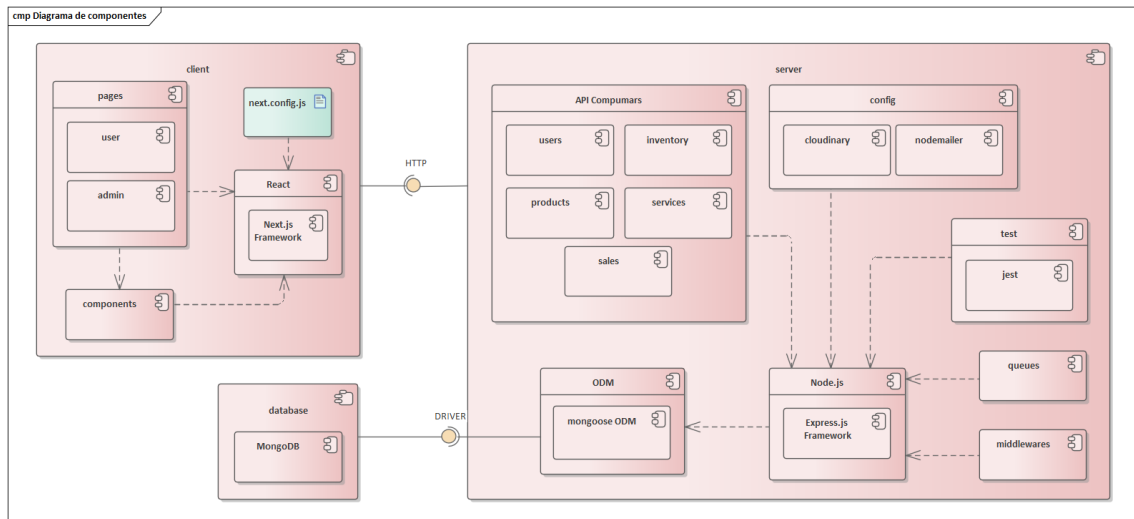


Figura 25. Diagrama de componentes

Estas piezas de software o componentes ilustradas en la Figura 25 se encuentran organizadas de la siguiente manera:

- Cliente (*client*):** este componente es la parte visual de la aplicación web progresiva desarrollada con React y la ayuda del *framework* Next.js. El cliente es el responsable de proporcionar al usuario final las funcionalidades que posee el sistema, tanto para usuarios clientes como para usuarios pertenecientes a la administración. Además, permite la interacción con cada una de estas funcionalidades ayudándose de un conjunto de componentes (*components*) reutilizables. De la misma manera, el componente cliente realiza las distintas peticiones al servidor (API Compumars) mediante el protocolo HTTP cuando es necesario y posee un artefacto de configuración denominado *next.config.js* que comprende los parámetros necesarios para establecer el comportamiento de la aplicación web como una PWA.
- Servidor (*server*):** el componente de servidor es el que comprende toda la lógica de negocio de la solución informática. Este componente se encuentra desarrollado en el entorno Node.js con la ayuda del *framework* Express.js y se encarga de dar respuesta a todas las peticiones elaboradas y enviadas a través del componente Cliente. Por ello, el servidor contiene la estructura de rutas y funcionalidades correspondientes a los módulos de gestión que componen el API de Compumars. Asimismo, posee piezas de software que hacen referencia a los *middlewares* (conjunto de funciones invocadas antes de resolver la petición final), *config* (configuración de funcionalidades para el envío de correos mediante *nodemailer* y carga imágenes a *cloudinary*), *queues* (establecimiento de tareas programadas), y *test* (pruebas unitarias y de integración).

- **Base de datos (*database*):** este componente representa la persistencia de los datos de la solución informática. Para ello se usa el sistema de base de datos NoSQL MongoDB el cual puede realizar comunicación con el componente Servidor mediante una librería de Modelado de Datos de Objetos (ODM por sus siglas en inglés) denominada *mongoose*.

Para culminar con el diseño de la arquitectura física, en la Figura 26 se puede evidenciar el Diagrama de Despliegue, en el cual se presentan los nodos físicos que conforman la arquitectura para la ejecución del sistema y proveen detalles de cómo y dónde se desplegará la solución informática.

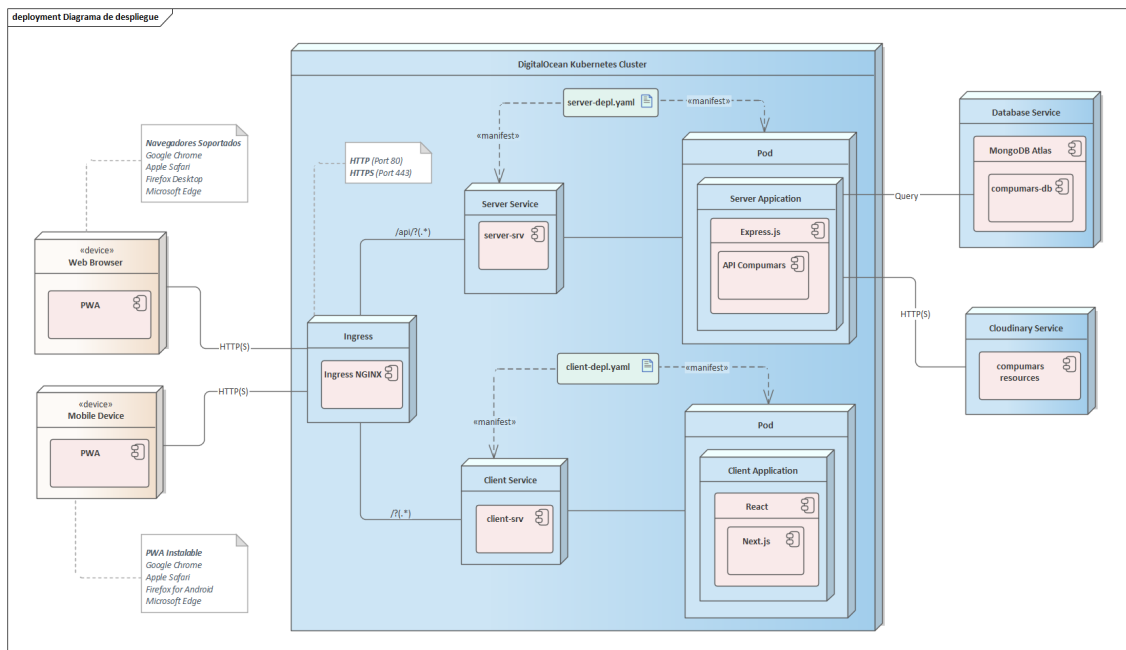


Figura 26. Diagrama de despliegue

De este diagrama de despliegue es importante resaltar los siguientes elementos:

- **Dispositivos (*devices*):** nodos que representan los dispositivos físicos mediante los cuales se puede tener acceso a la PWA.
- **Kubernetes Cluster:** permite la ejecución de aplicaciones (cliente y servidor) envueltas en contenedores de Docker, permitiendo que estas ocupen sus propias dependencias y servicios necesarios. Este clúster permite el desarrollo, la ejecución y la gestión de estas aplicaciones de un modo mucho más fácil.
- **Ingress:** este objeto (denominado elemento API) permite la gestión del acceso externo a los servicios del clúster mediante HTTP y HTTPS a través de los puertos 80 y 443 respectivamente. Además, permite redirigir las peticiones externas al servicio correcto (ya sea cliente o servidor).

- **Service:** tanto el *Server Service* como el *Client Service*, son elementos de Kubernetes que permiten exponer las aplicaciones (cliente y servidor) que se encuentran en su respectivo Pod para el acceso dentro del cluster.
- **Pod:** posee las aplicaciones (cliente y servidor) que se encuentran en contenedores de Docker. Estos elementos son accesibles desde los dispositivos gracias al trabajo conjunto que realiza el objeto Ingress y los Servicios enlazados a estos pods.
- **Manifiestos:** los archivos de manifiesto *server-depl* y *client-depl* son ficheros en formato *yaml* que proporcionan al Cluster las especificaciones para la creación de los elementos service y pod de las aplicaciones servidor y cliente.
- **Database Service:** representa el servicio externo al cual la aplicación *server* requiere acceso para la persistencia de los datos de la PWA. Para este caso se usa el servicio de base de datos MongoDB Atlas.
- **Cloudinary Service:** servicio externo que permite la gestión de imágenes que son utilizadas en la PWA. Este servicio es requerido por la aplicación *server*.

6.2. Objetivo 2: Implementar la arquitectura de la solución informática en una Aplicación Web Progresiva (PWA) mediante el uso de tecnologías JavaScript y la metodología ICONIX.

6.2.1. Implementación.

En esta fase se codificó las funcionalidades del sistema con el fin de responder a las necesidades del cliente especificadas en la arquitectura presentada del objetivo 1. La finalidad fue construir cada uno de los módulos de acuerdo a su prioridad para funcionamiento e integración especificados en la Tabla 13 y de esa manera poder abarcar cada uno de los escenarios que se va a desempeñar en el presente proyecto. Para lograrlo se utilizó un *stack* de tecnologías basadas en el lenguaje de programación JavaScript, denominado *Stack MERN* (MongoDB, Express.js, React, Node.js).

Tabla 13. Módulos de software a implementar

Módulo	Casos de uso	Tiempo empleado (N° de semanas)
Gestión de usuarios	CU01, CU03, CU04, CU05, CU06	1
Gestión de productos	CU02, CU14, CU15, CU16, CU17	2
Gestión de servicios	CU14, CU15, CU16, CU17	1
Gestión de inventario	CU18, CU19	3
Gestión de ventas	CU07, CU08, CU09, CU10, CU11, CU12, CU13	3

En líneas generales, el entorno de desarrollo se construyó mediante la arquitectura cliente-servidor, y tal como se puede visualizar en la Figura 27 donde se encuentra la estructura general de carpetas del proyecto, se creó dos carpetas principales: *client* y *server*, para representar estos dos conceptos respectivamente. Además, se desarrolló la solución informática bajo el patrón de diseño Modelo Vista Controlador (MVC), separando la parte de los modelos y controladores en el Servidor y la Vista en el Cliente. Todo el código fuente que se muestra de manera general de aquí en adelante puede ser visto completamente en el repositorio del proyecto Compumars¹ alojado en GitHub.

¹ Repositorio del proyecto Compumars: <https://github.com/victand98/compumars>

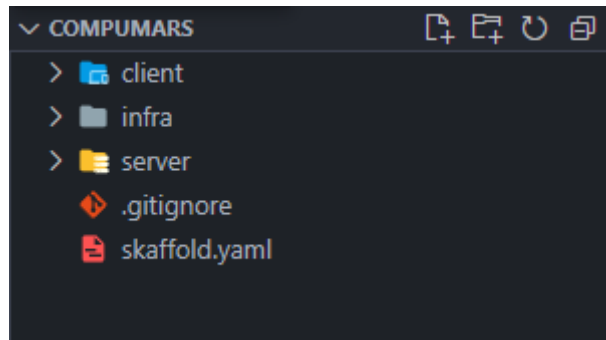


Figura 27. Estructura general de carpetas del proyecto

De forma detallada, la estructura general del proyecto está compuesta por:

- **client:** o denominado Cliente, es el directorio que posee todos los archivos necesarios para la construcción de la interfaz gráfica de la aplicación, así como la funcionalidad requerida para hacer peticiones al servidor.
- **infra:** o denominado Infraestructura, posee los archivos de configuración para la automatización del despliegue del sistema y el manejo de las aplicaciones (cliente y servidor) en contenedores mediante Docker.
- **server:** o denominado Servidor, este directorio posee los archivos de modelos, controladores y rutas necesarias para la construcción del API REST.
- **skaffold.yaml:** Archivo de manifiesto que se encarga de construir e implementar la aplicación, además lee los cambios de las carpetas *client*, *server* e *infra* y actualiza cualquier modificación realizada en estos directorios en tiempo real.

El entorno de desarrollo dentro de la carpeta *client* posee la estructura denotada en la Figura 28, y contiene todos los elementos visuales envueltos en la carpeta *components*, las funciones de lógica y de comunicación con el servidor dentro del directorio *lib*, las páginas que finalmente serán mostradas al usuario final y que reutilizan componentes previamente construidos, en el interior de la carpeta *pages*, los archivos de especificación del tema visual de la aplicación envueltos en la carpeta *styles*, y finalmente los archivos encargados de establecer el tipo de dato de los diferentes objetos del sistema dentro de la carpeta *types*.

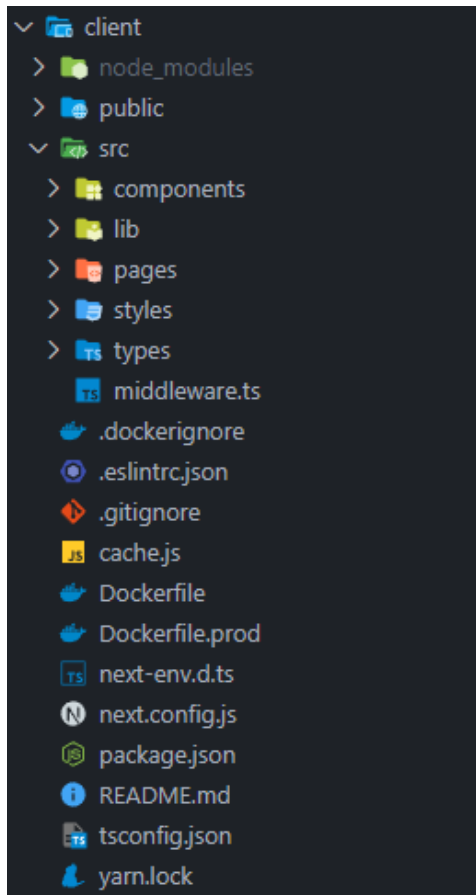


Figura 28. Estructura de directorios del Cliente

Cabe también mencionar al archivo *middleware.ts* (ver en la Figura 29) ya que juega un papel demasiado importante puesto que se encarga de manejar la autenticación de un usuario en el sistema y de proteger las distintas páginas que requieran de un inicio de sesión previo.

```
compumars - middleware.ts
1 import { NextURL } from "next/dist/server/web/next-url";
2 import type { NextRequest } from "next/server";
3 import { NextResponse } from "next/server";
4
5 export function middleware(req: NextRequest) {
6   const session = req.cookies.get("session");
7   const url: NextURL = req.nextUrl.clone();
8
9   if (!session) {
10    const searchParams = url.search;
11    url.search = "";
12    url.searchParams.set("returnUrl", `${url.pathname}${searchParams}`);
13    url.pathname = `/ingresar`;
14    return NextResponse.redirect(url);
15  }
16
17  return NextResponse.next();
18 }
19
20 export const config = {
21   matcher: ["/perfil/:path*", "/panel/:path*", "/carrito"],
22 };
23
```

Figura 29. Middleware de autenticación en el cliente

De igual modo, cabe mencionar al archivo *next.config.js* (ver Figura 30) el cual es de suma importancia ya que posee la configuración esencial para la ejecución de la aplicación cliente con React y el *framework* Next.js. Así como también posee las líneas de configuración para convertir la aplicación convencional en una PWA.

```
MERN (Workspace) - next.config.js
1  /** @type {import('next').NextConfig} */
2
3  const intercept = require("intercept-stdout");
4  const runtimeCaching = require("./cache");
5  const withPWA = require("next-pwa")({
6    dest: "public",
7    disable: process.env.NODE_ENV === "development",
8    runtimeCaching,
9  });
10
11 // safely ignore recoil stdout warning messages
12 function interceptStdout(text) {
13   if (text.includes("Duplicate atom key")) {
14     return "";
15   }
16   return text;
17 }
18
19 // Intercept in dev and prod
20 intercept(interceptStdout);
21
22 const nextConfig = withPWA({ output: "standalone" });
23
24 module.exports = nextConfig;
25
```

Figura 30. Código de configuración general del cliente y la PWA

El espacio de trabajo del servidor, que se encuentra en la carpeta *server*, la misma que tiene una estructura tal cual se muestra en la Figura 31, contiene dentro del directorio *models* todos los modelos detallados en el diagrama de clases realizado, así como todas las funciones necesarias y las reglas del negocio de Compumars en la carpeta *routes*. Asimismo, el servidor comprende: los archivos de configuración para el envío de correos electrónicos y subida de imágenes en la carpeta *config*; funciones para el control de acceso basado en roles (RBAC por sus siglas en inglés), autenticación y manejo de errores en la carpeta *middlewares*; operaciones para ejecutar tareas programadas y generar un *hash* de contraseñas en los directorios *queues* y *helpers* respectivamente.

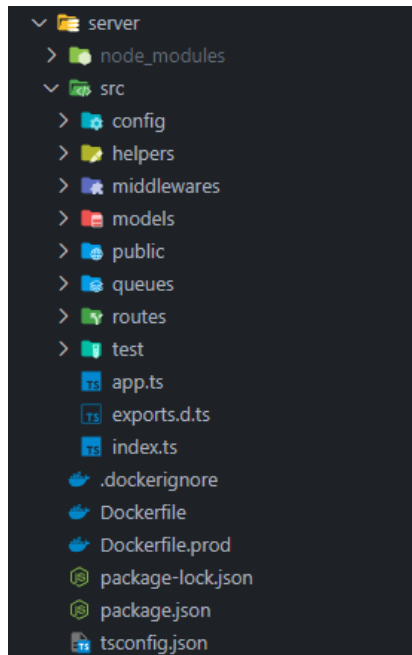


Figura 31. Estructura de directorios del Servidor

Por destacar algunas de las funciones más relevantes en el servidor, en la Figura 32 se puede evidenciar el código utilizado para la conexión y envío de correos electrónicos en Node.js.

```
compumars - emails
1 import { Settings } from "@compumars/common";
2 import nodemailer from "nodemailer";
3 import { MailOptions } from "nodemailer/lib/json-transport";
4 import { getSettingValue } from "../helpers";
5 import { SettingModel } from "../models";
6
7 export const transporter = async () => {
8   const settings = await SettingModel.findSettings([
9     Settings.host,
10    Settings.port,
11    Settings.secure,
12    Settings.authUser,
13    Settings.authPassword,
14  ]);
15
16  const host = getSettingValue(Settings.host, settings);
17  const port = getSettingValue(Settings.port, settings);
18  const secure = getSettingValue(Settings.secure, settings);
19  const authUser = getSettingValue(Settings.authUser, settings);
20  const authPassword = getSettingValue(Settings.authPassword, settings);
21
22  return nodemailer.createTransport({
23    host,
24    port: parseFloat(port),
25    secure: secure === "true",
26    auth: { user: authUser, pass: authPassword },
27  });
28 };
29
30 /**
31  * Function that allows send an email with default mail options
32  * @param mailOptions
33  * @returns
34  */
35 export const sendMail = async (mailOptions: MailOptions) => {
36   const transport = await transporter();
37   const defaultMailOptions: MailOptions = {
38     from: "Compumars" <compumars@noreply.com>, // sender address
39     subject: "COMPUMARS", // Subject line
40   };
41   return transport.sendMail({ ...defaultMailOptions, ...mailOptions });
42 };
43
44
```

Figura 32. Función para el envío de correos electrónicos desde el servidor

Por otro lado, en la Figura 33 se puede observar el contenido del archivo *password.ts*, el cual posee las funciones que han sido elaboradas con la finalidad de generar un hash de una contraseña y para comparar una cadena de texto entrante con una contraseña previamente codificada. Esto fue relevante para la aplicación ya que permite guardar estos datos sensibles en un formato no legible.

```
compumars - password.ts
1 import { scrypt, randomBytes } from "node:crypto";
2 import { promisify } from "util";
3
4 const scryptAsync = promisify(scrypt);
5
6 export class Password {
7   static async toHash(password: string) {
8     const salt = randomBytes(8).toString("hex");
9     const buf = (await scryptAsync(password, salt, 64)) as Buffer;
10
11     return `${buf.toString("hex")}.${salt}`;
12   }
13
14   static async compare(storedPassword: string, suppliedPassword: string) {
15     const [hashedPassword, salt] = storedPassword.split(".");
16     const buf = (await scryptAsync(suppliedPassword, salt, 64)) as Buffer;
17
18     return buf.toString("hex") === hashedPassword;
19   }
20 }
21
```

Figura 33. Funcionalidad para generar y comparar un hash de una contraseña

A continuación, la carpeta *infra* que posee los archivos de manifiesto *yaml* (ver Figura 34) que representan la infraestructura de la solución informática. Estos archivos no son más que la representación de objetos de *Kubernetes*, y básicamente han sido utilizados para ejecutar las aplicaciones (cliente, servidor) dentro de contenedores y además para aplicar las políticas que permitan la comunicación entre estos elementos. *Kubernetes* aquí actúa como un elemento de vital importancia ya que automatiza la creación de estos objetos y permite el despliegue de la aplicación de una manera simple y estructurada tanto para entornos de desarrollo como de producción.

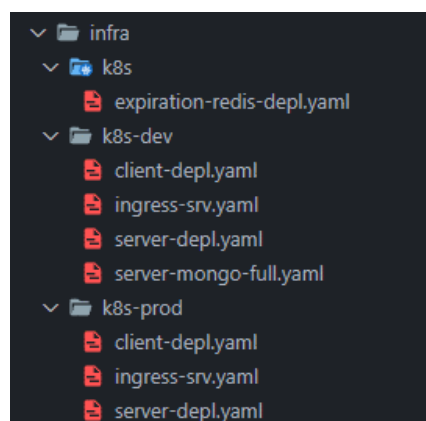


Figura 34. Estructura de archivos de manifiesto para la infraestructura de la aplicación

En la Figura 35 se puede observar el archivo de manifiesto *ingress-srv.yaml* el cual genera un objeto de *Kubernetes* que se encarga de gestionar el acceso externo a los servicios en el clúster mediante el protocolo HTTP. Aquí se definieron reglas basadas en la dirección de acceso para controlar el enrutamiento del tráfico entrante: “/api/” para el servidor y “/” para la aplicación cliente.

```
compumars - ingress-srv.yaml
1  apiVersion: networking.k8s.io/v1
2  kind: Ingress
3  metadata:
4    name: ingress-service
5    annotations:
6      kubernetes.io/ingress.class: nginx
7      nginx.ingress.kubernetes.io/use-regex: "true"
8      nginx.ingress.kubernetes.io/proxy-body-size: "0"
9      nginx.ingress.kubernetes.io/proxy-read-timeout: "600"
10     nginx.ingress.kubernetes.io/proxy-send-timeout: "600"
11  spec:
12    rules:
13      - host: compumars.dev
14        http:
15          paths:
16            - path: /api/(?.*
17              pathType: Prefix
18              backend:
19                service:
20                  name: server-srv
21                  port:
22                    number: 3000
23            - path: /?(.*)
24              pathType: Prefix
25              backend:
26                service:
27                  name: client-srv
28                  port:
29                    number: 3000
30
```

Figura 35. Configuración del acceso externo a los servicios de la aplicación

De igual forma, se configuraron dos archivos de manifiesto, uno para el cliente y el segundo para el servidor. En la Figura 36 se puede observar un ejemplar de estos archivos, en donde cada uno de estos está compuesto por:

- **Deployment:** encargado de la creación de un objeto de *Kubernetes* denominado *pod*, el mismo que comprende un contenedor de Docker donde se ejecuta cada aplicación.
- **Service:** objeto de *Kubernetes* encargado de especificar la forma en la cual se puede acceder a este *pod*.

```

compumars - client-depl.yaml
1  apiVersion: apps/v1
2  kind: Deployment
3  metadata:
4    name: client-depl
5  spec:
6    replicas: 1
7    selector:
8      matchLabels:
9        app: client
10   template:
11     metadata:
12       labels:
13         app: client
14     spec:
15       containers:
16         - name: client
17           image: victorandresrojas/compumars-client
18   ---
19   apiVersion: v1
20   kind: Service
21   metadata:
22     name: client-srv
23   spec:
24     selector:
25       app: client
26     ports:
27       - name: client
28         protocol: TCP
29         port: 3000
30         targetPort: 3000

```

Figura 36. Manifiesto con la configuración de un *deployment*

Un archivo de manifiesto de vital importancia es *expiration-redis-depl.yaml* (ver Figura 37) el cual posee la configuración necesaria para la creación de una base de datos en memoria en *Kubernetes*, en este caso *redis*. Este objeto fue útil para poder guardar el registro del tiempo de expiración de las órdenes que se generan en la aplicación.

```

compumars - expiration-redis-
1  apiVersion: apps/v1
2  kind: Deployment
3  metadata:
4    name: expiration-redis-depl
5  spec:
6    replicas: 1
7    selector:
8      matchLabels:
9        app: expiration-redis
10   template:
11     metadata:
12       labels:
13         app: expiration-redis
14     spec:
15       containers:
16         - name: expiration-redis
17           image: redis
18   ---
19   apiVersion: v1
20   kind: Service
21   metadata:
22     name: expiration-redis-srv
23   spec:
24     selector:
25       app: expiration-redis
26     ports:
27       - name: db
28         protocol: TCP
29         port: 6379
30         targetPort: 6379
31

```

Figura 37. Manifiesto con la configuración de *redis*

Una vez se ha indicado de manera general los archivos más relevantes de la implementación del proyecto. A continuación, se presentará el resultado final de las interfaces gráficas obtenidas por cada uno de los módulos desarrollados, de igual modo, el resultado completo de estas interfaces se puede observar en el **Anexo 4. Interfaces gráficas de la implementación de la solución informática.**

6.2.1.1. Desarrollar el módulo de gestión de usuarios.

En la Figura 38 se puede observar la lista de usuarios registrados en el sistema, en esta página se puede pulsar el botón Registrar Usuario para realizar el ingreso de los datos de un nuevo usuario al sistema. Asimismo, cada registro mostrado tiene la opción para Modificar el rol y para dar de baja un usuario, o, por ende, para volver a activar a un usuario que previamente fue inactivado. Además, se puede filtrar la visualización de los usuarios en dos tipos: Activos o Inactivos o se puede hacer uso del buscador de usuarios por medio del correo, nombres o apellidos.

The screenshot displays the 'Usuarios' management interface. On the left is a dark sidebar with a menu including 'Panel', 'Usuarios', 'Proveedores', 'Categorías', 'Marcas', 'Productos', 'Inventario', 'Órdenes de compra', 'Órdenes de venta', 'Impuestos', and 'Configuración'. The main content area is titled 'Usuarios' and includes a '+ Agregar Usuario' button. Below this is a search bar labeled 'Buscar usuarios' with the instruction 'Buscar por correo, nombres o apellidos'. A 'Lista de Registros' section features a table with the following data:

DETALLES	ESTADO	ROL	HISTORIAL	ACCIONES
Admin Test admin@localhost.com	ACTIVO	Administrador	Creado el 14 de julio de 2022 a las 10:12 Última Modificación 14 de julio de 2022 a las 10:12	[Edit] [Delete]
Mario Cueva mecueva@unl.edu.ec	ACTIVO	Operador	Creado el 14 de julio de 2022 a las 10:14 Última Modificación 9 de agosto de 2022 a las 22:26	[Edit] [Delete]
Maria Encalada maria.a.encalada@unl.edu.ec	ACTIVO	Cajero	Creado el 14 de julio de 2022 a las 23:18 Última Modificación 9 de agosto de 2022 a las 22:26	[Edit] [Delete]

Figura 38. Página de Listar Usuarios de Compumars

Dentro del módulo de gestión de usuarios también se encuentran los proveedores, en la Figura 39 se puede ver que se desarrolló un listado de estos, donde se puede añadir un nuevo registro o modificar uno ya existente. De igual manera se puede buscar mediante razón social, correo electrónico, nombres o apellidos.

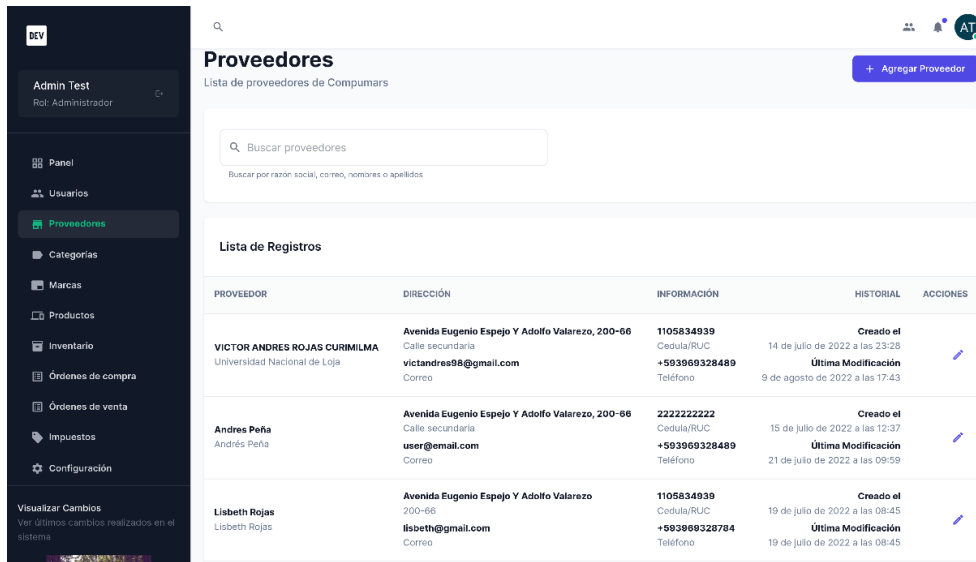


Figura 39. Página de Listar Proveedores

Por otro lado, en la Figura 40 se exhibe el listado de las direcciones registradas por cada usuario de Compumars. En donde también existe la opción para registrar o modificar un registro ya existente.

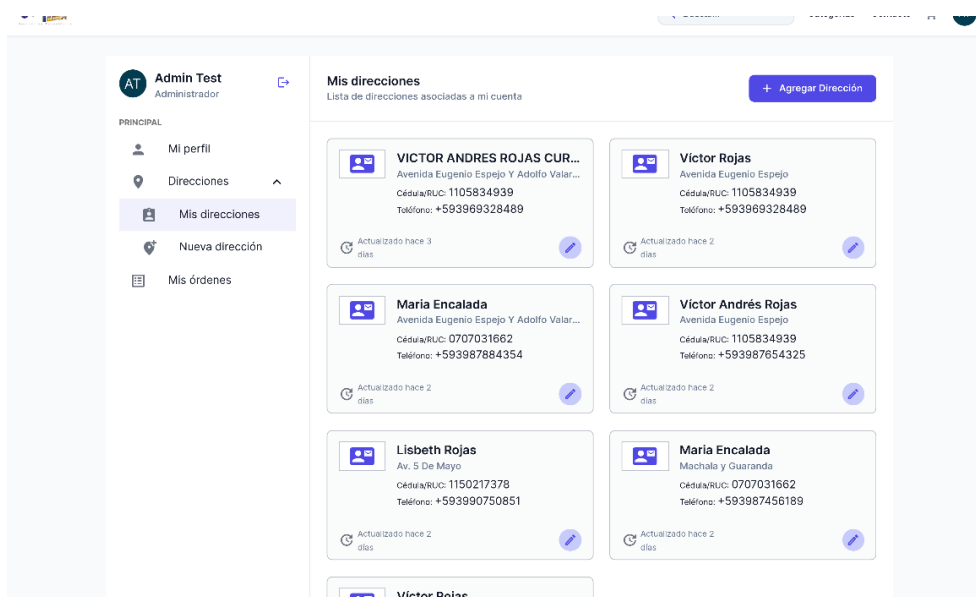


Figura 40. Página de Listar Direcciones del Usuario

6.2.1.2. Desarrollar el módulo de gestión de productos.

La gestión de productos dentro de Compumars se realiza tal como se muestra en la Figura 41. Donde se cuenta con las funciones de: buscar productos ya registrados mediante el código, código auxiliar y nombre; registrar nuevo producto, modificar detalles y dar de baja. Esta última opción es la que condiciona a cada producto para aparecer o no en la página principal de la aplicación.

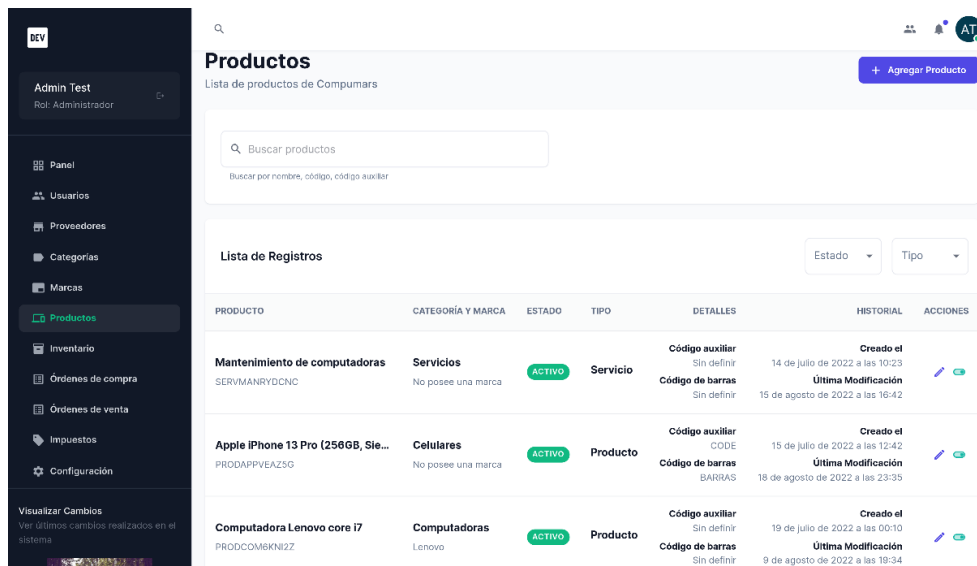


Figura 41. Página de Listar Productos de Compumars

Este módulo también comprende la gestión de categorías y como se evidencia en la Figura 42, se puede ver el registro de todas las categorías del sistema. Es posible registrar una nueva categoría, buscar mediante nombre o enlace de ruta amigable (*slug*), y modificar los datos de un registro existente.

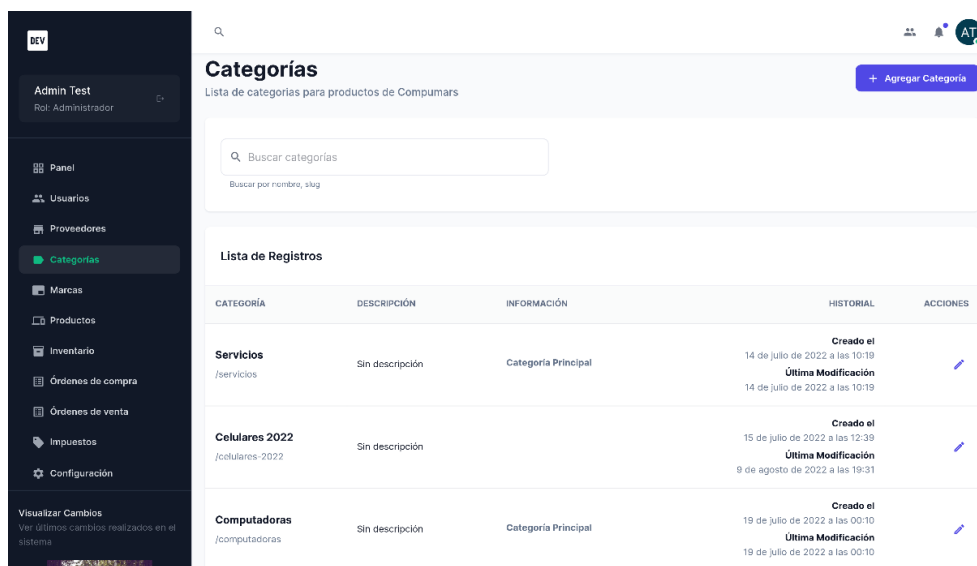


Figura 42. Página de Listar Categorías

La gestión de marcas también forma parte de este módulo. En la Figura 43 se expone los registros existentes, que de igual manera poseen la opción para registrar, modificar y buscar mediante el nombre.

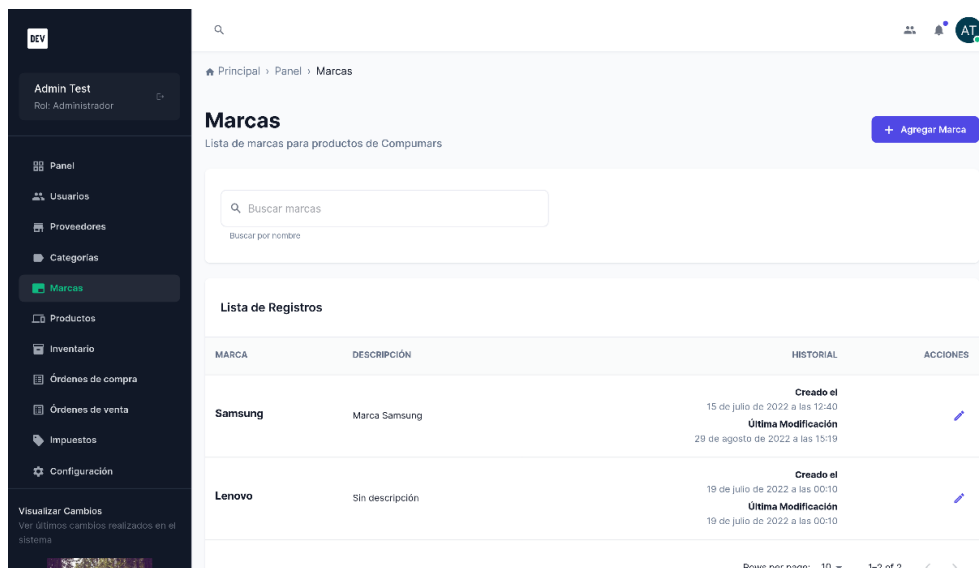


Figura 43. Página de Listar Marcas

6.2.1.3. Desarrollar el módulo de gestión de servicios.

Según los requerimientos del usuario, los servicios son un tipo de producto más, por consiguiente, el proceso de añadir un nuevo servicio se lo realizó en el formulario de registro de un nuevo producto. Esto fue posible ya que existen dos tipos de productos: Producto y Servicio. Este campo permitió distinguir a estos dos conceptos. Y para ello, como se muestra en la Figura 44, en la página de listar los productos registrados dentro de Compumars, existe una opción que actúa como filtro para distinguir ya sea los productos o los servicios que se encuentren guardados dentro del sistema.



Figura 44. Página de Listar Servicios

6.2.1.4. Desarrollar el módulo de gestión de inventario.

Como parte de la gestión de inventario consta el detalle de productos disponibles. Se lo realizó con una variación entre las cantidades disponibles que se presentan de forma pública en el sitio web y de lo que existe en forma física, esto considerando que las órdenes realizadas por los clientes tienen un tiempo límite de reserva de los productos hasta que se confirme el pago o se cancele la orden. A continuación, en la Figura 45 se puede observar la lista de productos que se encuentran en el inventario de Compumars.

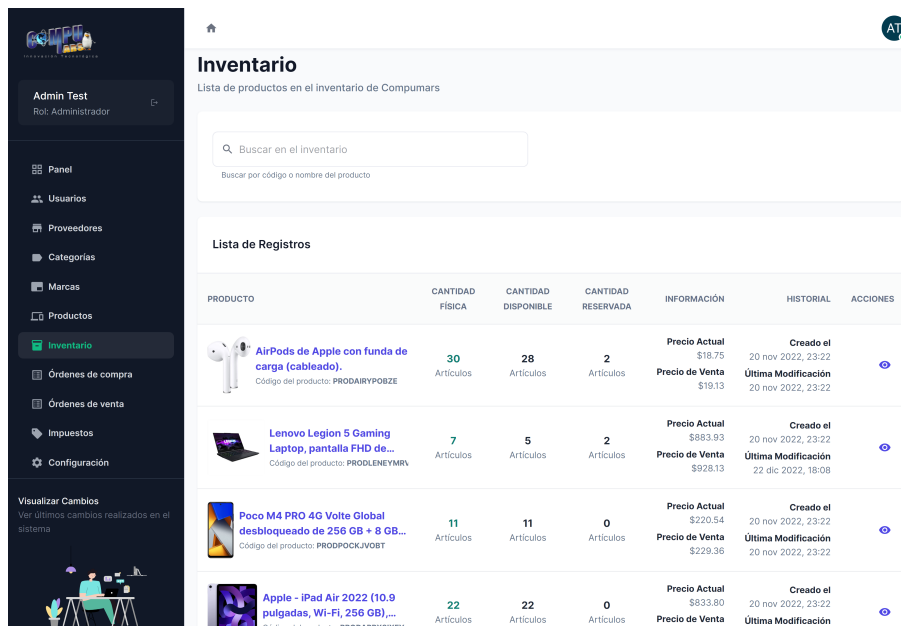


Figura 45. Página del Inventario de Compumars

Cada registro del inventario posee un botón para acceder a los detalles o Kardex del producto, aquí se puede evidenciar información tal como: el precio del producto y la lista de movimientos que históricamente ha tenido ese producto por medio de las órdenes de compra o de venta registradas, todos estos datos se los puede observar en la Figura 46.

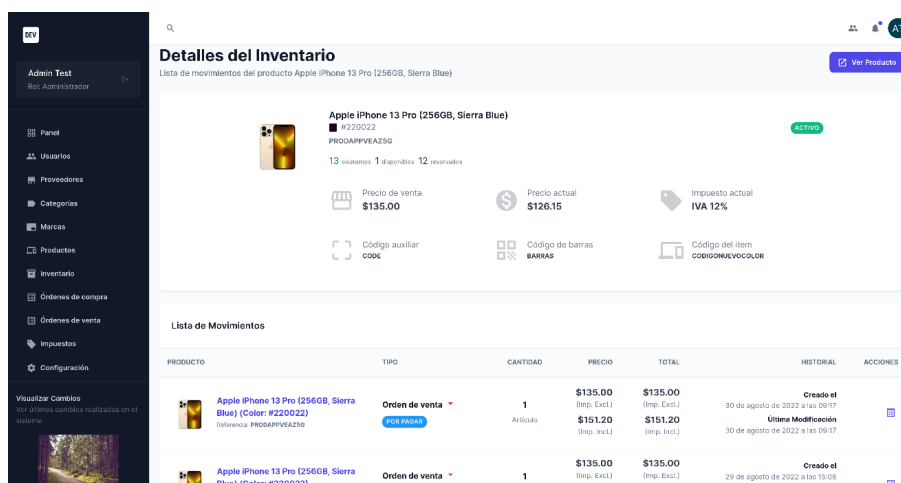


Figura 46. Página de Kardex de un producto

Las órdenes de compra también cumplieron un papel fundamental dentro del inventario, ya que estas representan la entrada de artículos al almacén de Compumars y así mismo, actualizan dicho inventario. En la Figura 47 se observa el listado de órdenes de compra registradas, del mismo modo existe la opción para buscar mediante código de referencia y para registrar una nueva orden de compra.

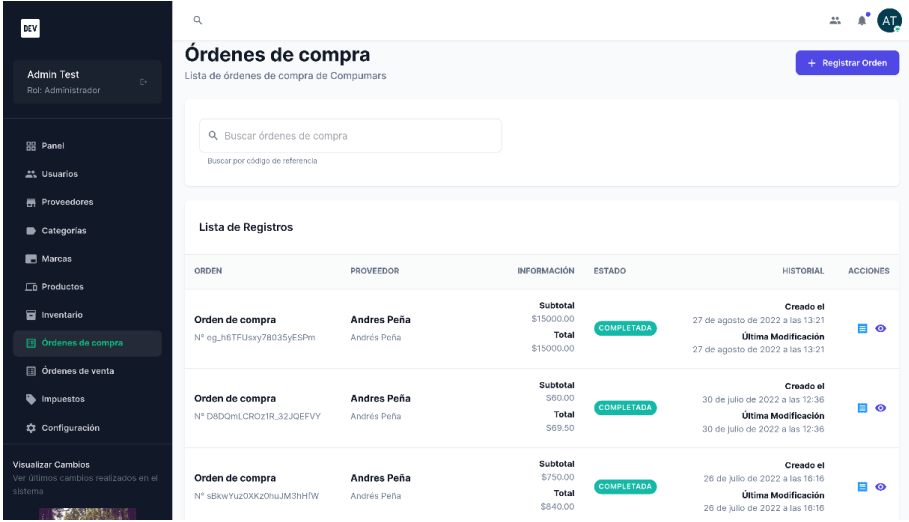


Figura 47. Página de Listar Órdenes de Compra

6.2.1.5. Desarrollar el módulo de gestión de ventas.

La gestión de ventas se refiere al proceso por el cuál Compumars vende productos a sus clientes, sirve como registro de la salida de bienes y representa (al igual que las órdenes de compra) una actualización de inventario. Por ello, en la Figura 48 se expone la lista de órdenes de venta guardadas en el sistema. Las mismas tienen la opción para visualizar los detalles que componen cada orden. Por otro lado, se puede buscar estas órdenes por el código de referencia y también se puede registrar una nueva orden de venta.

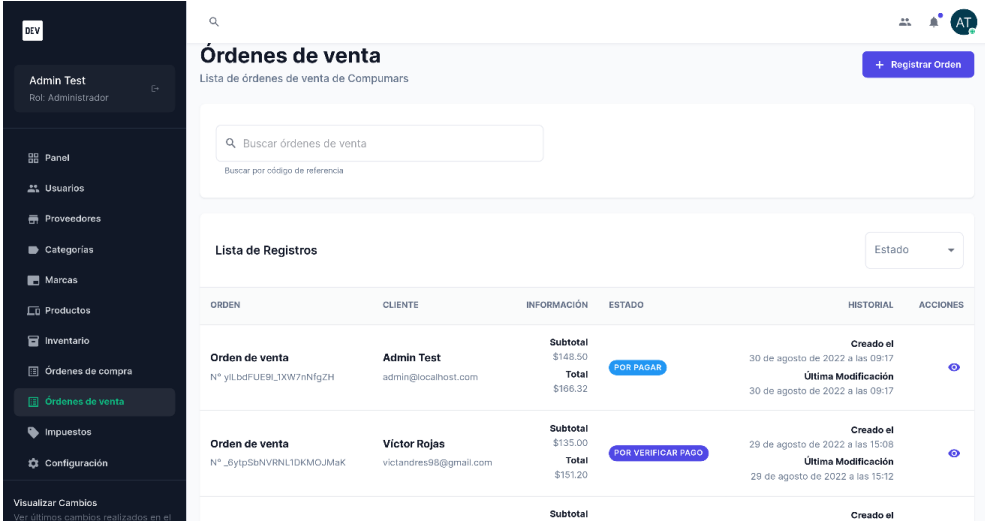


Figura 48. Página de Listar Órdenes de Venta

En caso de que un cliente que se haya registrado dentro de Compumars y que desee realizar una orden, este agrega los productos que va a comprar y en la página de Carrito de compras de la Figura 49 debe completar los campos requeridos que son: dirección de envío, dirección de comprobante de venta y el método de pago. Y al realizar el pedido, se procede a crear una nueva orden de venta en el sistema.

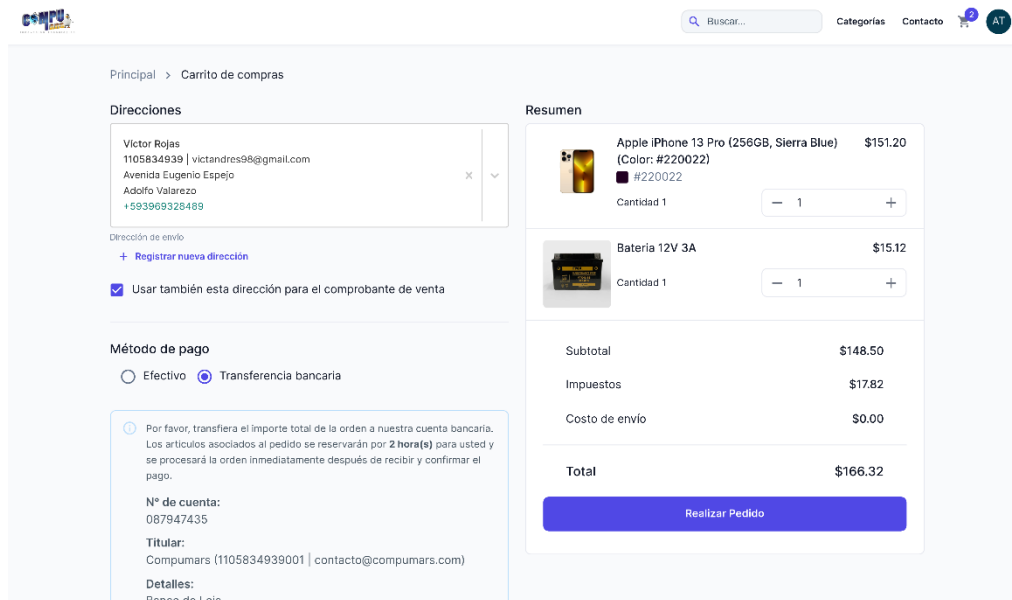


Figura 49. Página de Carrito de Compras

Todas estas órdenes que realiza cada cliente de Compumars aparecen en la página Mis órdenes, ilustrada en la Figura 50, donde se puede buscar por el código de referencia o filtrar de acuerdo a los distintos estados por los que puede atravesar una orden.

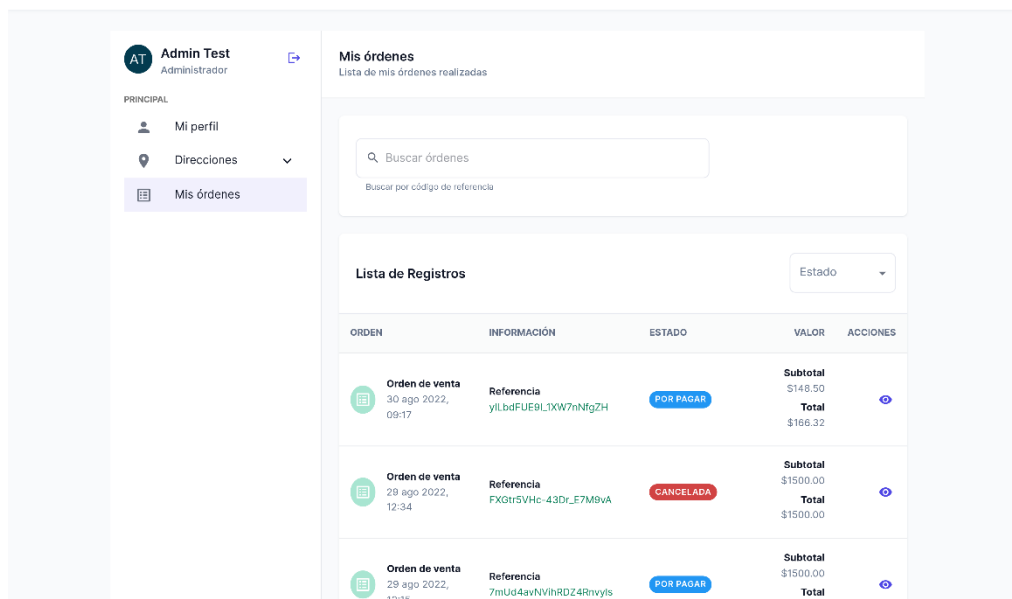


Figura 50. Página de Listar órdenes realizadas por un usuario

Para finalizar, se construyó el apartado para registrar la constancia de pago de una orden registrada, esta sección se encuentra disponible por un tiempo determinado mediante un parámetro establecido en la configuración del sistema, el mismo que indica el número de horas máximo en las que una orden realizada por un cliente reserva los productos asociados. Para esto, en la Figura 51 se puede ver el formulario que solicita el ingreso de la referencia del pago realizado por el cliente por la compra de los artículos de la orden.

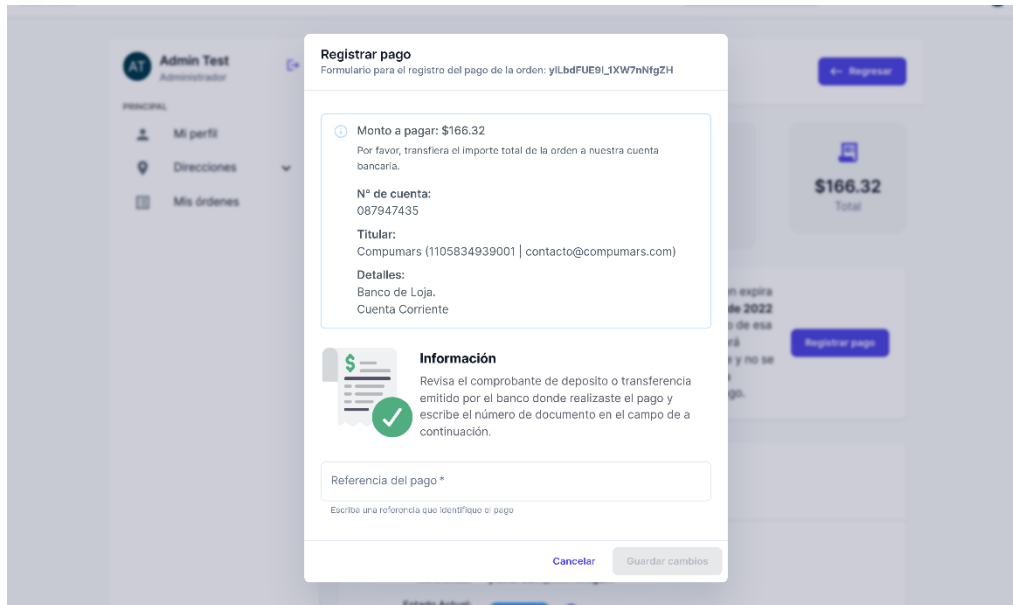


Figura 51. Página de registrar constancia del pago de una orden de venta

6.2.2. Realizar pruebas de funcionamiento del software.

En esta fase que marca el término al segundo objetivo del presente TT, se realizaron pruebas unitarias y de integración sobre las funcionalidades más relevantes construidas durante la pasada fase de Implementación. Estas pruebas se efectuaron con la ayuda de la librería *Jest*, en compañía de la librería *SuperTest*, ambas de JavaScript. Para empezar, en las Figura 52 y Figura 53 se puede visualizar la estructura de los directorios de estas pruebas.

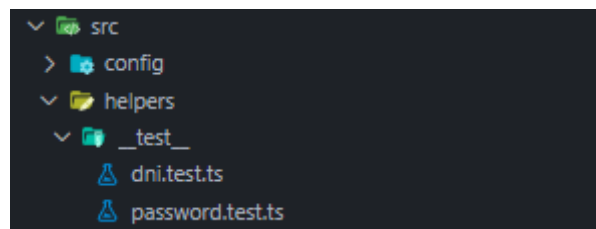


Figura 52. Estructura del directorio de pruebas unitarias

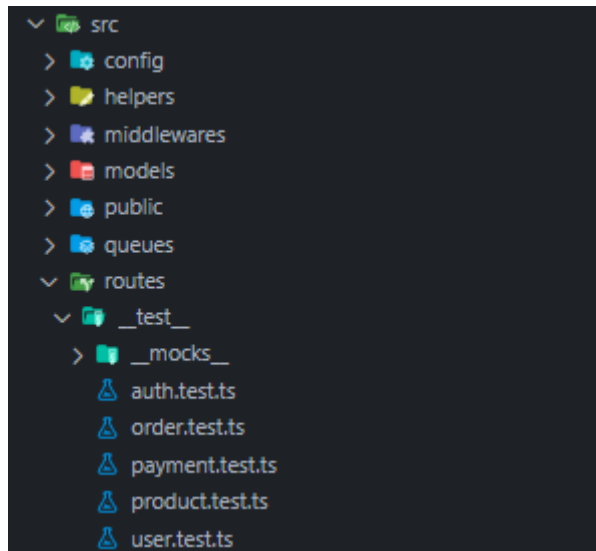


Figura 53. Estructura del directorio de pruebas de integración

Además, cabe resaltar que se estableció un directorio para la configuración general e inicial de la librería *Jest*, esta estructura se encuentra en la Figura 54. Básicamente estos archivos contienen los ajustes necesarios para la conexión a la base de datos MongoDB en un entorno de pruebas, también poseen la carga de datos iniciales, declaración de variables de entorno necesarias para el funcionamiento y funciones globales que fueron útiles en varias ocasiones.

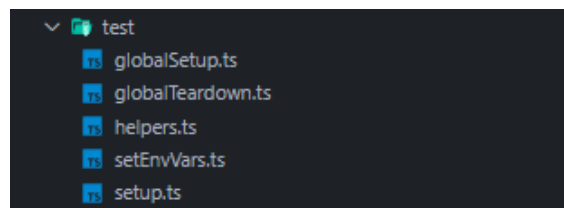


Figura 54. Estructura del directorio para la configuración de pruebas con *Jest*

Abriendo paso a la ejecución de estas pruebas, en la Figura 55 se encuentra el resultado obtenido, en donde se puede evidenciar un total de 37 pruebas individuales envueltas en 7 grupos, de las cuales efectivamente el 100% de estas se ejecutaron satisfactoriamente en aproximadamente 15 segundos de un tiempo estimado de 16 segundos.

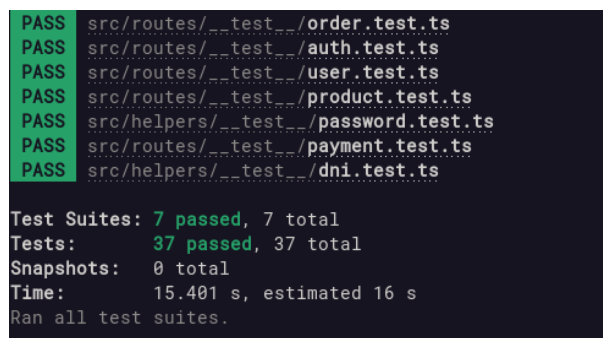


Figura 55. Resultado de las pruebas unitarias y de integración de la Aplicación Web Progresiva

Es importante resaltar que el tiempo de ejecución también incluye el tiempo de conexión a la base de datos MongoDB, puesto que esta se debe preparar en memoria al momento de poner en funcionamiento todas las pruebas. Por último, en el **Anexo 6. Pruebas unitarias y de integración de la Aplicación Web Progresiva** se puede ver más a detalle el contenido de los ficheros utilizados en el presente proyecto para la ejecución de estas pruebas de funcionamiento.

6.3. Objetivo 3: Comparar los procesos de gestión y comercialización tradicional y web en un ambiente de pruebas simulado para determinar los beneficios de la actualización.

Para el presente objetivo se realizaron pruebas a la solución desarrollada con el fin de comparar los resultados obtenidos frente a los procesos que hasta ahora se han llevado a cabo en la empresa Compumars. Es importante mencionar que los apartados que se describirán a continuación se elaboraron basándose en las etapas descritas por la metodología “Implantación de un Sistema ERP en una PyME” del autor Pablo González [2].

6.3.1. Planificación del caso de prueba para la simulación manual versus web.

6.3.1.1. Etapa 1. Organización y preparación

Actividad 1. Consolidación del comité del proyecto

La conformación del comité del proyecto se hizo considerando a quienes participaron de forma directa o indirecta en el proceso de evaluar las funcionalidades de la solución desarrollada. Los actores involucrados fueron:

Tabla 14. Comité del proyecto

Autor	Cargo	Rol
Director de TT	Individual	Supervisar el trabajo técnico realizado
Autor del TT	Individual	Desplegar y configurar la herramienta desarrollada
Gerente-Propietario	Individual	Comprobar el correcto funcionamiento de la solución desarrollada y emitir comentarios
Personal de Compumars	Grupal	Comprobar el correcto funcionamiento de la solución desarrollada y emitir comentarios
Clientes (estudiantes)	Grupal	Comprobar el correcto funcionamiento de la solución desarrollada y emitir comentarios

Actividad 2. Establecimiento del cronograma

Se establecieron tiempos límites para cada una de las actividades principales desarrolladas en este objetivo. Estas se documentan en el cronograma de la Tabla 15.

Tabla 15. Cronograma de actividades

Actividades	Actores	Semana 1					Semana 2					Semana 3					Semana 4					
		L	M	M	J	V	L	M	M	J	V	L	M	M	J	V	L	M	M	J	V	
E1. AC1. Identificación y mitigación de posibles riesgos.	Director TT y Autor de TT	■																				
E2. AC1. Documentación de manual de instalación de la aplicación web.	Director TT y Autor de TT		■	■																		
E2. AC2. Documentación de manual de usuario de la aplicación web.	Director TT y Autor de TT			■	■	■																
E3. AC1. Identificación de los escenarios de evaluación.	Director TT y Autor de TT						■	■														
E3. AC2. Creación de los escenarios de prueba.	Director TT y Autor de TT								■	■	■											
E4. AC1. Pruebas de los escenarios en la forma tradicional (manual).	Autor de TT, Gerente-Propietario, Personal de Compumars											■	■	■	■	■						
E4. AC2. Pruebas de los escenarios en la forma automatizada.	Autor de TT, Gerente-Propietario, Personal de Compumars											■	■	■	■	■						
E4. AC3. Elaboración y aplicación de una encuesta de satisfacción.	Autor de TT y Estudiantes													■	■	■						
E4. AC4. Aplicación de Acta de Validación de la aplicación web.	Autor de TT y Gerente-Propietario																■	■	■	■		

Actividad 3. Mitigación de riesgos

Se hizo una breve recopilación de los posibles problemas que se pueden presentar durante el despliegue de la solución desarrollada. Estos fueron clasificados en dos categorías: Técnico (en referencia a fallas en equipos computacionales) o Funcional (en referencia a fallas de usuarios). Además, se considera el impacto en donde: Alto significa que el sistema informático no se puede ejecutar, Medio significa que la solución se ejecuta, pero pueden aparecer errores que impidan su funcionamiento, Bajo significa que el sistema informático se ejecuta correctamente, pueden aparecer advertencias leves pero que no afectan a su correcto funcionamiento. Asimismo, se considera la Probabilidad en la cual este riesgo puede llegar a suceder. A continuación, en la Tabla 16 se presenta la recopilación realizada.

Tabla 16. Mitigación de riesgos

Descripción	Tipo	Impacto	Probabilidad	Mitigación
Despliegue de la aplicación web progresiva en un entorno que no sea compatible con las tecnologías usadas en el desarrollo del presente TT.	Técnico	Alto	Baja	Analizar previamente las características de los servidores para seleccionar una plataforma de servicios en la nube.
Error en la instalación de la Aplicación Web Progresiva debido a versiones incompatibles de las tecnologías.	Técnico	Alto	Media	Actualizar a versiones más recientes de todas las tecnologías usadas en el servidor.
Incompatibilidad de los navegadores web desde donde se accede a la Aplicación Web Progresiva.	Técnico	Alto	Baja	Analizar la lista de navegadores compatibles con las características que requiere una Aplicación Web Progresiva tanto para dispositivos móviles como para ordenadores.
Falta de información sobre el uso adecuado de la Aplicación Web Progresiva	Funcional	Bajo	Baja	Elaborar manual de usuario.

6.3.1.2. Etapa 2. Análisis Funcional

Actividad 1. Resumen de requerimientos funcionales

En esta actividad el propósito fue realizar un resumen de los requerimientos funcionales obtenidos durante la fase de análisis y diseño preliminar. No obstante, esta tarea se realizó durante el desarrollo del primer objetivo (véase **Sección 6.1.1.1. Modelado de requerimientos**) y la debida aprobación en el **Anexo 2. Aprobación de los requerimientos de software**. De esta manera, se puede decir que esta etapa queda finalizada.

6.3.1.3. Etapa 3. Diseño Conceptual

Actividad 1. Modelo Funcional

En lo que concierne el Modelo Funcional, se construyeron los distintos diagramas que definen el diseño de la arquitectura física y lógica de la solución informática, los mismos que se basaron en los resultados arrojados por el modelado de requerimientos. Este modelado se puede observar a detalle en la **Sección 6.1.2. Diseño de la arquitectura física y lógica**, por lo que esta actividad queda cubierta.

Actividad 2. Diseño de interfaces

Para realizar esta actividad, la metodología sugiere realizar una representación visual de cada una de las pantallas del sistema, esto con el objetivo de tener una visión mucho más amplia de las funcionalidades de la solución informática. Esta actividad queda cubierta ya que se realizaron algunos *Wireframes* durante el desarrollo de los casos de uso (véase **Anexo 3. Modelado de la arquitectura física y lógica. Apartado 3.3. Especificación de casos de uso**).

6.3.2. Ejecución de la simulación.

6.3.2.1. Etapa 4. Construcción

Actividad 1. Carga de Datos y Configuraciones

Continuando con la etapa de Construcción, en esta actividad se realizaron las distintas configuraciones de las tecnologías y herramientas a implementar tanto en el entorno de desarrollo como de producción. Con esta finalidad, se construyó un manual de instalación y configuración, el mismo que se puede ver a detalle en el **Anexo 5. Manual de instalación de la Aplicación Web Progresiva**.

Actividad 2. Pruebas en entorno de desarrollo

Una vez realizada la carga de datos y las respectivas configuraciones, se procedió a realizar un conjunto de pruebas dentro de un ambiente de desarrollo, esto con el fin de evaluar las diferentes funcionalidades de la solución informática previo a su ejecución en un ambiente de producción. Es así que se elaboraron pruebas unitarias y de integración, las mismas que al ponerlas en marcha arrojaron los resultados que se pueden evidenciar en la Figura 56, en donde se puede constatar un total de 37 pruebas individuales envueltas en 7 grupos, de las cuales efectivamente el 100% de estas se ejecutaron satisfactoriamente en aproximadamente 15 segundos de un tiempo estimado de 16 segundos. Se puede observar detalladamente el resultado de estas pruebas en el **Anexo 6. Pruebas unitarias y de integración de la Aplicación Web Progresiva.**

```
PASS src/routes/__test__/order.test.ts
PASS src/routes/__test__/auth.test.ts
PASS src/routes/__test__/user.test.ts
PASS src/routes/__test__/product.test.ts
PASS src/helpers/__test__/password.test.ts
PASS src/routes/__test__/payment.test.ts
PASS src/helpers/__test__/dni.test.ts

Test Suites: 7 passed, 7 total
Tests:       37 passed, 37 total
Snapshots:  0 total
Time:        15.401 s, estimated 16 s
Ran all test suites.
```

Figura 56. Resultado de las pruebas unitarias y de integración de la Aplicación Web Progresiva

6.3.2.2. Etapa 5. Preparación Final y Arranque del Sistema

Actividad 1. Carga de datos en entorno real

Una vez concluida la Etapa 4, se procedió a trasladar la Aplicación Web Progresiva desde un entorno local a una plataforma de servicios en la nube, logrando así que los distintos involucrados en el sistema puedan posteriormente experimentar las funcionalidades directamente desde sus dispositivos electrónicos. Asimismo, se construyó un manual para la instalación y configuración de la solución informática tanto para el entorno de desarrollo como de producción, el mismo que se puede ver a detalle en el **Anexo 5. Manual de instalación de la Aplicación Web Progresiva.**

Es importante destacar aquí que, al desplegar la aplicación en un entorno de producción, se habilitó la opción para instalar la Aplicación Web Progresiva en el dispositivo que el usuario esté haciendo uso. Por ejemplo, tal como se puede observar en la Figura 57, cuando se accede desde un ordenador mediante un navegador compatible aparecerá una alerta que permite al usuario la opción de instalar esta aplicación como si se tratase de una aplicación nativa del sistema operativo, cumpliendo así con una característica importante que tienen las PWA (ver **sección 4.2.2. Aplicaciones Web Progresivas**).



Figura 57. Alerta para la instalación de la PWA en ordenadores

Del mismo modo, en dispositivos móviles al acceder desde un navegador compatible, se mostrará una alerta que da la opción al usuario de instalar la aplicación en su dispositivo, esto se puede contemplar en la Figura 58.



Figura 58. Alerta para la instalación de la PWA en dispositivos móviles

Actividad 2. Pruebas

En esta instancia se acordó una reunión con un grupo de estudiantes de primer ciclo paralelos A y B de la carrera de Ingeniería en Computación con el fin conocer el grado de aceptación que tuvo la solución informática desarrollada. Con este propósito se elaboró una encuesta en la que por cada pregunta se implementó la escala de Likert [54], la misma que mediante un rango del 1 al 5 permitió medir el nivel de acuerdo y desacuerdo de los encuestados. Seguidamente, en la Tabla 17 se encuentran la lista de preguntas realizadas:

Tabla 17. Lista de preguntas de la encuesta de satisfacción

Pregunta	Respuesta				
	1	2	3	4	5
¿Considera que la Aplicación Web es fácil de usar?	Totalmente en desacuerdo	En desacuerdo	Neutral	De acuerdo	Totalmente de acuerdo
¿Considera que la manera de llevar a cabo la adquisición de productos dentro de la Aplicación Web es adecuada?	Totalmente en desacuerdo	En desacuerdo	Neutral	De acuerdo	Totalmente de acuerdo
¿Considera que mediante el uso de la Aplicación puede agilizar la adquisición de productos?	Totalmente en desacuerdo	En desacuerdo	Neutral	De acuerdo	Totalmente de acuerdo
¿Considera útil que la Aplicación se pueda instalar en un dispositivo directamente?	No es útil	Poco útil	Neutral	Útil	Muy útil
¿Cómo calificaría el proceso de realizar un pedido?	Muy difícil	Difícil	Neutral	Fácil	Muy Fácil
¿Qué dificultad representó para usted el proceso de buscar productos en la aplicación?	Muy difícil	Difícil	Neutral	Fácil	Muy Fácil
¿Qué tan probable es que usted vuelva a utilizar la aplicación?	Nada probable	Poco probable	Neutral	Probable	Muy probable
¿Qué tan probable es que recomiende la Aplicación a otras personas que necesiten adquirir algún producto o servicio en línea?	Nada probable	Poco probable	Neutral	Probable	Muy probable
¿En qué podría mejorar la presente aplicación? Por favor, Indique su recomendación.	Recomendaciones				

El modelo de encuesta final se puede ver en el **Anexo 7. Modelo de Encuesta de Satisfacción** y la recopilación de los resultados obtenidos en el **Anexo 8. Resultados de la Encuesta de Satisfacción**.

Por otro lado, se establecieron escenarios de evaluación para comparar los procesos de gestión y comercialización tradicional y web, de igual manera estos escenarios permitieron verificar el cumplimiento de los requerimientos planteados en etapas anteriores del presente TT. En la Tabla 18 y Tabla 19 se puede observar los escenarios que se identificaron para evaluar los procesos tradicional y web respectivamente.

Tabla 18. Escenarios de prueba para el proceso tradicional

Número	Escenario de prueba
1	Registrar productos en el inventario
2	Realizar una nueva venta
3	Registrar y validar pago de una venta

Tabla 19. Escenarios de prueba para el proceso web

Número	Escenario de prueba	Caso de uso	Referencia al escenario de prueba tradicional
1	Registrar órdenes de venta	CU09	2
2	Completar órdenes	CU10	3
3	Visualizar pagos de órdenes	CU12	3
4	Validar pagos de órdenes	CU13	3
5	Registrar productos	CU15	1
6	Registrar órdenes de compra	CU19	1

Para los escenarios de evaluación identificados en la Tabla 18 y Tabla 19 se tomaron en cuenta los casos más relevantes para dar resolución a la pregunta de investigación planteada para el presente TT. Asimismo, para el proceso de evaluación se utilizaron los modelos que se pueden evidenciar en la Tabla 20 y Tabla 21 tanto para el proceso tradicional como para el proceso web. En ambos casos lo importante fue reflejar las actividades realizadas y obtener el tiempo de ejecución de cada escenario de prueba para su posterior comparación.

Tabla 20. Modelo de caso de prueba para el proceso tradicional

Caso de Prueba		Nro: 1
Título		
Área de prueba		
Objetivos		
Acciones Realizadas	Tiempo empleado	Encargado
Tiempo total empleado		

Tabla 21. Modelo de caso de prueba para el proceso web

Caso de Prueba de Aceptación		Nro: 1	
Título			
Pantalla			
Área de prueba			
Objetivos			
Acciones a Realizar			
Resultado(s) Esperado(s)			
Validado	Si	No	Parcialmente
Observación(es)			
Nombre tester			
Tiempo empleado			
Referencia al caso de prueba manual			

Tras proceder con la respectiva ejecución de los escenarios de evaluación, se procedió a recopilar los resultados en el **Anexo 9. Acta de validación de Pruebas de la Aplicación Web Progresiva para la gestión y comercialización de productos y servicios para la empresa Compumars** en donde se encuentra la constancia correspondiente al acta de aprobación de pruebas de la Aplicación Web Progresiva elaborada en conjunto con el Gerente-Propietario y el personal de Compumars.

Actividad 3. Documentación

Para finalizar, en esta actividad se procedió a realizar el manual de usuario del sistema con el fin de describir el uso de la Aplicación Web Progresiva desde la perspectiva de cada uno de los roles que conforman la solución informática, el resultado completo de este manual se lo puede evidenciar en el **Anexo 10. Manual de usuario de la Aplicación Web Progresiva**.

6.3.3. Establecer un análisis cuantitativo de los resultados.

Para dar cumplimiento a este apartado se consideraron los resultados obtenidos en la sección previa. Para empezar, en la Tabla 22 y Figura 59 se puede ver el resumen de resultados obtenidos de la encuesta de satisfacción aplicada a los estudiantes de la quienes actuaron con el rol de clientes de la aplicación web. Para el promedio de aceptación de cada pregunta realizada se tuvo presente los niveles de medición 3, 4 y 5 de la escala de Likert aplicada, dado a que estos valores representan un resultado positivo o de aceptación por parte de los encuestados.

Tabla 22. Resumen de resultados de la Encuesta de Satisfacción

Pregunta	1	2	3	4	5	Promedio de Aceptación
¿Considera que la Aplicación Web es fácil de usar?	0.00%	2.13%	0.00%	19.15%	78.72%	97.87%
¿Considera que la manera de llevar a cabo la adquisición de productos dentro de la Aplicación Web es adecuada?	0.00%	0.00%	6.38%	29.79%	63.83%	100.00%
¿Considera que mediante el uso de la Aplicación puede agilizar la adquisición de productos?	0.00%	0.00%	4.26%	27.66%	68.09%	100.00%
¿Considera útil que la Aplicación se pueda instalar en un dispositivo directamente?	0.00%	4.26%	14.89%	25.53%	55.32%	95.74%
¿Cómo calificaría el proceso de realizar un pedido?	0.00%	4.26%	2.13%	36.17%	57.45%	95.74%
¿Qué dificultad representó para usted el proceso de buscar productos en la aplicación?	0.00%	2.13%	2.13%	27.66%	68.09%	97.87%
¿Qué tan probable es que usted vuelva a utilizar la aplicación?	4.26%	2.13%	19.15%	36.17%	38.30%	93.62%
¿Qué tan probable es que recomiende la Aplicación a otras personas que necesiten adquirir algún producto o servicio en línea?	2.13%	2.13%	12.77%	46.81%	36.17%	95.74%
Promedio Final						97.07%

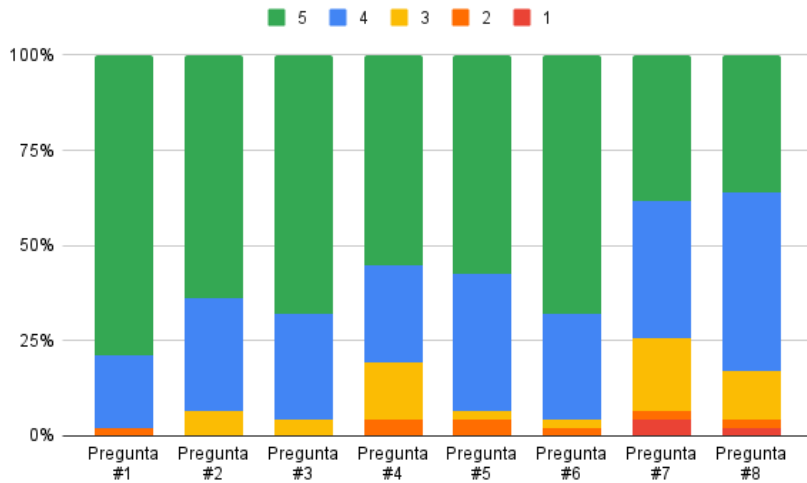


Figura 59. Resumen de resultados de la Encuesta de Satisfacción

La aplicación de la Encuesta de Satisfacción permite evidenciar que, bajo el criterio de los alumnos encuestados (47 en total) con un promedio de aceptación final del 97.07%, la solución informática desarrollada cumple en su mayoría con los requerimientos necesarios para llevar a cabo la correcta gestión y comercialización de productos y servicios para la empresa Compumars. De esta manera, los alumnos correspondientes al periodo académico abril 2022 – octubre 2022 dan la aceptación de la Aplicación Web Progresiva realizada.

Del mismo modo, para los resultados obtenidos tras la ejecución de los escenarios de evaluación de los procesos tradicional y web, se recopilieron los datos en la Figura 60, Figura 61 y Figura 62, en donde se puede evidenciar la diferencia en términos de tiempo total empleado tanto para el proceso tradicional llevado a cabo por Compumars como del proceso web de la solución informática desarrollada en el presente TT.

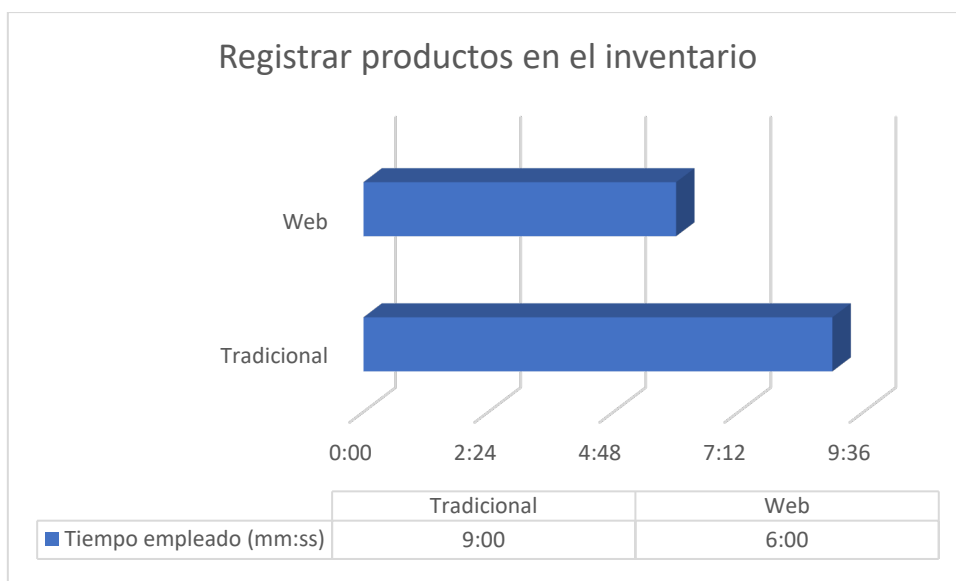


Figura 60. Tiempo empleado al Registrar productos en el inventario (Tradicional vs Web)

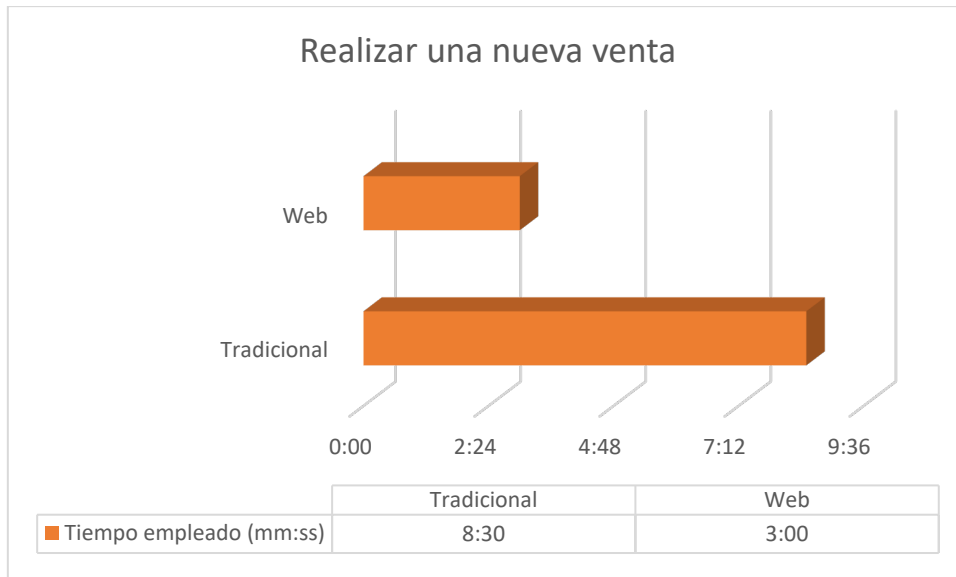


Figura 61. Tiempo empleado al Realizar una nueva venta (Tradicional vs Web)

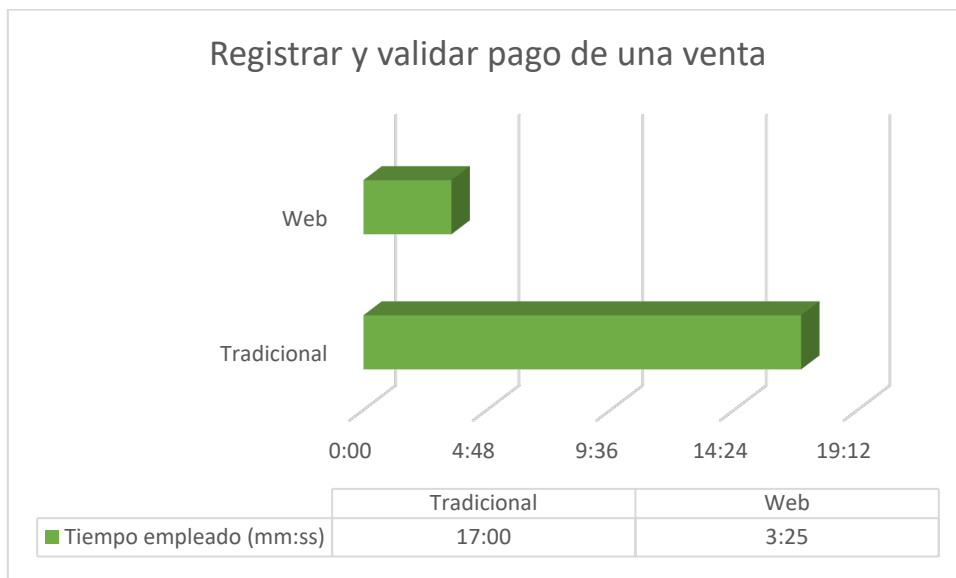


Figura 62. Tiempo empleado al Registrar y validar pago de una venta (Tradicional vs Web)

Como se puede observar en los tres escenarios presentados, es evidente la diferencia de tiempo que existe entre ambos procesos, por lo que la Aplicación Web Progresiva para la gestión y comercialización de productos y servicios ayuda a la empresa Compumars en la ejecución de sus actividades hablando en términos de tiempo y organización, ya que además de ahorrarle al personal de esta empresa segundos, minutos e incluso horas a la larga, les permite organizar cada una de las tareas de acuerdo a cada rol dentro de una sola aplicación.

7. Discusión

La realización de la Aplicación Web Progresiva para la gestión de procesos dentro de Compumars se realizó trabajando de la mano con su Gerente-Propietario, quien ha mostrado apertura y apoyo con el presente Trabajo de Titulación. Se hizo uso de la metodología ICONIX para guiar la construcción del mismo. El desarrollo de este proyecto constó de 3 fases, las cuales corresponden a cada uno de los objetivos planteados en la presentación del anteproyecto. De dichas fases se habla a continuación:

7.1. **Objetivo 1: Construir el modelo de arquitectura física y lógica en base a los requerimientos del usuario utilizando el Modelamiento de desarrollo de software dirigido por ICONIX.**

Existen varias técnicas de recolección de datos las cuales pudieron haber ayudado a la obtención de requerimientos conformes a la gestión y comercialización de productos y servicios para la empresa Compumars, sin embargo, para el cumplimiento de esta tarea se utilizó la entrevista y la observación, cuyos resultados fueron suficientes para de esta manera organizar el documento de modelado de requerimientos y diagrama del dominio. Asimismo, la revisión de los trabajos relacionados con el objeto de estudio (ver **Sección 4.4. Trabajos relacionados**) ayudó tener una noción general sobre los requerimientos que posee una empresa cuando se trata de construir este tipo de soluciones informáticas, de tal modo que se pudo corroborar la relación con los requerimientos obtenidos a través de las técnicas de recolección de datos empleadas en el presente proyecto, eso sí, diferenciándose en ciertos aspectos que se solicitaron especialmente por parte del Gerente-Propietario de Compumars, como son las órdenes de venta y compra para la gestión del inventario, el tratamiento que se le da a los productos involucrados en una orden de venta en donde no se llegó a completar el pago por parte del cliente, y además, los diferentes roles involucrados en el sistema. Es así como se consiguió un total de 30 requerimientos funcionales y 6 requerimientos no funcionales (ver **Sección 6.1.1. Análisis y diseño preliminar**).

Además, las indicaciones descritas dentro de las etapas de Análisis y Diseño correspondientes a la fase de Modelado de la Metodología ICONIX (ver **Sección 4.2.1.2. Fases de la metodología ICONIX**) permitieron ejecutar de manera ordenada las actividades para el cumplimiento de este primer objetivo, donde efectivamente se corrobora que ICONIX es una metodología iterativa e incremental por la razón que posibilitó la mejora de los modelos en cada iteración, ayudando a generar la documentación necesaria para implementar el proyecto, incluso el apartado de Diseño Detallado propone los distintos componentes UML que fueron necesarios para realizar el diseño de la arquitectura física y lógica.

7.2. **Objetivo 2: Implementar la arquitectura de la solución informática en una Aplicación Web Progresiva (PWA) mediante el uso de tecnologías JavaScript y la metodología ICONIX.**

Las etapas de Inicialización y Construcción concernientes a la fase de Planificación que describe la Metodología ICONIX (ver **Sección 4.2.1.2. Fases de la metodología ICONIX. Secciones 2 y 3**) permitieron llevar de manera ideal el desarrollo del segundo objetivo, ya que como lo menciona la etapa de Inicialización, en esta instancia se empezó a explorar los requerimientos obtenidos, los casos de uso y a familiarizarse con las tecnologías que se usaron para la implementación del proyecto. A partir de aquí se procedió a codificar cada uno de los módulos que componen la solución informática (ver **Sección 6.2.1. Implementación**), dando como resultado pequeñas series de lanzamientos similar a como lo describe la etapa de Construcción de esta metodología. No obstante, cabe resaltar que existieron funcionalidades en el transcurso de la implementación que generaron retrasos y ampliaron en cierto modo el alcance del proyecto debido a la complejidad de las reglas del negocio al momento de ponerlas en marcha, como por ejemplo el manejo de las órdenes de compra y venta, el envío de correos electrónicos, la operación de *rollbacks* para revertir el estado de la base de datos en caso de que ocurriera un error inesperado en el transcurso de una operación en el servidor, entre otras.

Por su lado, el hecho de utilizar tecnologías basadas en JavaScript permitió la construcción de las distintas funcionalidades requeridas del proyecto, esto gracias a que este lenguaje de programación tiene la capacidad de implementar programación orientada a objetos y reactiva tanto para el cliente (*frontend*), como para la construcción de servicios y APIs REST (ver **Sección 4.3.1. JavaScript**). A pesar de ello, se utilizó TypeScript, esto con el fin de ofrecerle a JavaScript la ventaja de ser un lenguaje fuertemente tipado y de esa manera facilitar el proceso de desarrollo y robustecer la codificación. Cabe destacar el uso de la librería React para la construcción de la parte cliente, ya que sus facultades posibilitaron la construcción dinámica y escalable de la aplicación web gracias a su capacidad de generar elementos reutilizables, lo cual resultó ser una ventaja en términos de tiempo (ver **Sección 4.3.1.2. React**). Sin embargo, para ofrecerle todavía más potencial a esta librería, se usó el *framework* Next.js, esto con el propósito de agilizar la construcción de la Aplicación Web Progresiva, permitir el levantamiento de páginas sin preocuparse por el enrutamiento debido a su característica de generar rutas automáticamente, y de ejecutar funciones en el lado del servidor, tales como la autenticación de usuarios y el control del acceso a páginas protegidas. Por último, es importante enfatizar el empleo de Kubernetes para el proceso de construcción, despliegue y comunicación de las aplicaciones cliente y servidor como si se tratase de una sola aplicación, ya que permitió realizar estos procesos de manera sencilla a como se lo haría

de modo convencional, es decir, cliente y servidor de forma individual. Adicionalmente, Kubernetes permite facilitar el despliegue de la aplicación en una plataforma de servicios en la nube.

7.3. Objetivo 3: Comparar los procesos de gestión y comercialización tradicional y web en un ambiente de pruebas simulado para determinar los beneficios de la actualización.

Para dar cumplimiento a este tercer objetivo se tomaron en consideración las actividades más relevantes de la metodología descrita en “Implantación de un Sistema ERP en una PyME” [2]. De este modo primeramente se procedió a realizar una planificación tomando en cuenta las actividades y los principales actores involucrados en la ejecución de las mismas (ver **Sección 6.3.1.1. Organización y preparación**). Seguidamente se hizo una recopilación de los posibles problemas que podrían darse durante la construcción final y despliegue de la aplicación web, en donde para cada problema se estableció una posible solución para mitigar el riesgo. Por lo tanto, se corrobora que estas actividades fueron de demasiada utilidad para la posterior ejecución de la planificación realizada.

Seguidamente se realizaron tareas como: la carga de datos y configuraciones, las pruebas unitarias y de integración (ver **Sección 6.3.2.1. Etapa 4. Construcción**), y el despliegue de la Aplicación en una plataforma de servicios en la nube, que para este caso fue *DigitalOcean* (ver **Sección 6.3.2.2. Etapa 5. Preparación Final y Arranque del Sistema**). Esta última tarea fue la que demandó más trabajo debido a las distintas configuraciones que se tuvieron que realizar dentro de este proveedor de servicios y en el proyecto. Además, la aplicación necesitaba ser desplegada bajo el protocolo https (*Hypertext Transfer Protocol Secure* por sus siglas en inglés) ya que para evidenciar una de las características más importantes de las Aplicaciones Web Progresivas, es decir que sea instalable para cualquier dispositivo desde el navegador sin la dependencia de una tienda de aplicaciones (ver **Sección 4.2.2. Aplicaciones Web Progresivas**), se necesitaba que el sitio web desplegado se encuentre bajo este protocolo y esto conllevó también a la adquisición de un nombre de dominio web para la correcta configuración. Sin embargo, todas estas actividades se llevaron a cabo con éxito y finalmente se pudo desplegar la PWA para proceder con las pruebas planificadas con los usuarios participantes.

Para la evaluación de la aplicación se realizó una encuesta en donde se tomó en consideración el criterio de un grupo de estudiantes quienes actuaron bajo el rol de clientes del sistema, los resultados de esta encuesta evidenciaron que la solución informática es aceptada por los usuarios con un porcentaje del 97.07%, no obstante, existieron algunas recomendaciones las cuales debido al alcance del presente TT se las trasladó a la sección de

Trabajos Futuros de este proyecto. Asimismo, se ejecutaron pruebas en conjunto con el Gerente-Propietario y el personal de Compumars, esta evaluación tuvo el propósito de comparar los procesos de gestión y comercialización tradicional y web, para ello se cuantificó el tiempo empleado en las actividades involucradas de estos procesos. Para ambos contextos se tomaron en cuenta los escenarios de evaluación más relevantes en relación con la pregunta de investigación. Dentro de los resultados obtenidos se pudo constatar que para registrar una lista de productos entrantes al almacén y actualizar el inventario, a la aplicación web realizada le tomó 3 minutos menos cumplir con este objetivo a diferencia del procedimiento tradicional. Para registrar una nueva venta y actualizar el inventario de los productos salientes, a la aplicación web le tomó 5 minutos y 30 segundos menos en comparación con el proceso tradicional. Y, por último, para registrar el pago de una venta y generar el comprobante correspondiente, a la aplicación web le tomó 13 minutos y 35 segundos menos a diferencia del proceso tradicional (ver **Sección 6.3.3. Establecer un análisis cuantitativo de los resultados**).

En base a este análisis se refleja que la Aplicación Web Progresiva desarrollada en el presente TT ayuda a la empresa Compumars, a tal grado, de reducir los tiempos empleados y mejorar la organización de las actividades involucradas en la gestión y comercialización de sus productos y servicios. Por lo tanto, es importante mencionar que la pregunta de investigación planteada para este TT queda respondida satisfactoriamente.

8. Conclusiones

Una vez culminado el presente Trabajo de Titulación, se puede concluir lo siguiente:

- La Aplicación Web Progresiva desarrollada en el presente TT ayuda a la empresa Compumars de la ciudad de Loja a tal grado de reducir los tiempos empleados y mejorar la organización de las actividades relacionadas a la gestión y comercialización de sus productos y servicios, conforme a los resultados obtenidos de los escenarios de evaluación realizados, mediante los cuales se comparó estos procesos de forma tradicional y web en un trabajo conjunto con el Gerente-Propietario y el personal de esta empresa.
- El uso de la entrevista y la observación como técnicas de recolección de datos son herramientas útiles cuando se dispone del apoyo de los involucrados en tiempo y espacio.
- Las etapas de Análisis y Diseño correspondientes a la fase de Modelado de la Metodología ICONIX tuvieron un gran impacto en el desarrollo de la arquitectura física y lógica de la solución informática, ya que, desde la construcción de los casos de uso hasta la elaboración del diagrama de despliegue, esta metodología ofreció las pautas necesarias para pasar de la fase de modelado a la codificación de una forma rápida y eficiente.
- La implementación de la arquitectura de la solución informática en una Aplicación Web Progresiva (PWA) con el uso de tecnologías JavaScript por medio del *stack* MERN y la ayuda ofrecida por las etapas de Inicialización y Construcción concernientes a la fase de Planificación de la metodología ICONIX permitieron la codificación ágil y la obtención de software funcional que cumple con los requerimientos establecidos y los modelos obtenidos en el proceso de desarrollo de ICONIX.
- Las evaluaciones realizadas sobre la solución informática desarrollada reflejan que la misma tiene un funcionamiento óptimo, es fácil de usar, posee tiempos de respuesta accesibles y es muy útil en lo que respecta a ser una aplicación instalable en sistemas operativos de ordenadores y móviles. Igualmente, algunas sugerencias proporcionadas por los estudiantes participantes de las encuestas de satisfacción permitieron descubrir algunas nuevas funcionalidades que pueden ser implementadas en trabajos futuros.

9. Recomendaciones

Una vez culminado el presente Trabajo de Titulación, se recomienda lo siguiente:

- Hacer uso de la entrevista y observación directa como técnicas de recolección de datos ya que son herramientas de utilidad para tener una visión clara de los requerimientos que pueden o deben tener este tipo de soluciones informáticas.
- Indagar sobre los trabajos relacionados puesto que ayudan a comprender muchos términos asociados al campo de estudio sobre el cual se va a investigar y amplían la visión del mismo.
- Utilizar la Metodología ICONIX para llevar a cabo proyectos que demanden de software con funcionalidades ya sea simples o complejas, puesto que esta metodología ofrece la ventaja de refinar y actualizar la arquitectura física y lógica obtenida a lo largo del proyecto, permitiendo así conseguir resultados a corto plazo y con la documentación necesaria para su posterior mantenimiento.
- Usar la librería React con el *framework* Next.js para la construcción de Aplicaciones Web Progresivas que requieran de múltiples funcionalidades y en las que sea necesario que la solución informática se ejecute en cualquier sistema operativo como si se tratase de aplicaciones nativas.
- Es recomendable implementar Aplicaciones Web Progresivas en lugar de las Aplicaciones Web Tradicionales, ya que las PWA poseen un mejor rendimiento, se acoplan a cualquier dispositivo, pueden operar sin conexión a la red, y además para los equipos de desarrollo significa la reducción de costes y tiempo en la implementación.

9.1. Trabajos Futuros

- Implementar el servicio de pasarela de pagos para permitir a los clientes de Compumars la adquisición de productos y servicios mediante tarjetas de crédito/débito o pagos directos, y de esta manera facilitar aún más el proceso de compra en la aplicación web progresiva.
- Incorporar el servicio de Facturación Electrónica a la aplicación web progresiva desarrollada para de esta manera tener de forma centralizada los comprobantes de venta electrónicos emitidos por Compumars hacia sus clientes y de esta manera reducir aún más el tiempo ocupado en este tipo de proceso.

- Incluir a la solución informática desarrollada un módulo para la gestión de promociones, descuentos y ofertas de productos y servicios brindados por la empresa Compumars hacia su público objetivo.
- Integrar la funcionalidad de notificaciones push (mensajes que se envían desde un servidor remoto hasta los dispositivos que tienen instalada la PWA) para de este modo mantener a los clientes de Compumars al tanto de las novedades que ofrezca la empresa hacia el público.

10. Bibliografía

- [1] Q. Guagua, C. Cabeza, S. Nelson, B. Jaime, C. Ernesto, y O. Ibarra, "Gestión y comercialización: pequeñas y medianas empresas de servicios en Ecuador", *Revista de ciencias sociales, ISSN-e 1315-9518, Vol. 26, N.º. 3, 2020, págs. 194-206*, vol. 26, núm. 3, pp. 194–206, 2020, Consultado: el 11 de septiembre de 2022. [En línea]. Disponible en: <https://dialnet.unirioja.es/servlet/articulo?codigo=7565476&info=resumen&idioma=ENG>
- [2] P. González Fernández, "Implantación de un SISTEMA ERP en una PyME", 2015.
- [3] K. G. Rodríguez, O. J. Ortiz, A. I. Quiroz, y M. L. Parrales, "El e-commerce y las Mipymes en tiempos de Covid-19", 2020, doi: 10.48082/espacios-a20v41n42p09.
- [4] P. Jílková y P. Králová, "Digital Consumer Behaviour and eCommerce Trends during the COVID-19 Crisis", *International Advances in Economic Research 2021 27:1*, vol. 27, núm. 1, pp. 83–85, mar. 2021, doi: 10.1007/S11294-021-09817-4.
- [5] B. Z. Velasco, E. B. C. Espinoza, y M. A. M. Guatumillo, "El E-Commerce en las empresas ecuatorianas: Un análisis de los informes de la Cámara Ecuatoriana de Comercio Electrónico (CECE) en el marco de la pandemia covid-19", *Revista Publicando*, vol. 8, núm. 29, pp. 13–20, mar. 2021, doi: 10.51528/rp.vol8.id2176.
- [6] A. Bhatti, H. Akram, M. Basit, y A. U. Khan, "E-commerce trends during COVID-19 Pandemic Image Processing View project Non-coding RNAs in cancer diagnosis and therapy View project", 2020, Consultado: el 11 de mayo de 2022. [En línea]. Disponible en: <https://www.researchgate.net/publication/342736799>
- [7] P. P. Kharat y M. Nagare, "Business Development-B2B and B2C Ecommerce", *Journal homepage: www.ijrpr.com ISSN*, vol. 2582, pp. 0–7421, 2017, Consultado: el 11 de mayo de 2022. [En línea]. Disponible en: <https://www.ijrpr.com/uploads/V2ISSUE8/IJRPR1089.pdf>
- [8] PwC Asesores Empresariales Cia. Ltda., "DC0-Información pública E-Commerce en Ecuador", 2020, Consultado: el 11 de mayo de 2022. [En línea]. Disponible en: <https://www.pwc.ec/es/publicaciones/assets/e-commerce-ecuador.pdf>
- [9] M. Galardi Boza y Á. Ruiz Gándara, "Comercio Electrónico Evolución Covid-19", Universidad de Sevilla, Sevilla, 2020.
- [10] M. E. Sepúlveda, "Unidad 4: Conceptos básicos de los inventarios", en *Contabilidad General*, Nueva Granada, 2015.
- [11] R. A. Lancioni y K. Howard, "Inventory Management Techniques", *International Journal of Physical Distribution & Materials Management*, vol. 8, núm. 8, pp. 385–428, may 1978, doi: 10.1108/EB014432.
- [12] I. Journal, M. H. M, y S. Appaiah, "Stabilization of FIFO system and Inventory Management", *International Research Journal of Engineering and Technology*, 2017, Consultado: el 11 de mayo de 2022. [En línea]. Disponible en: www.irjet.net
- [13] J. A. Fuertes, "Métodos, técnicas y sistemas de valuación de inventarios. Un enfoque global", *Revista de la Agrupación Joven Iberoamericana de Contabilidad y Administración de Empresas (AJOICA)*, 2015.
- [14] A. A. Maldonado Cervantes, "MÉTODOS DE VALUACIÓN DE INVENTARIOS", 2018.
- [15] G. Welagedara y M. Victoire, "Inventory Management Techniques and Its Contribution on Better Management of Manufacturing Companies in RWANDA C... Inventory Management Techniques and Its Contribution on Better Management of Manufacturing Companies in RWANDA Case Study: SULFO RWANDA Ltd", *European Journal of Academic Essays*, vol. 2, núm. 6, pp. 49–58, 2015, Consultado: el 11 de mayo de 2022. [En línea]. Disponible en: www.euroessays.org

- [16] L. Olivia *et al.*, "Aplicación de la metodología semi-ágil ICONIX para el desarrollo de software: implementación y publicación de un sitio WEB para una empresa SPIN-OFF en el Sur de Sonora, México", 2014.
- [17] R. S. Casa Toapanta y M. A. Pilatasig Chiluisa, "DESARROLLO E IMPLEMENTACION DE UN SISTEMA PARA EL CONTROL DE INVENTARIOS EN LA EMPRESA SAVREH S.A. DE LA CIUDAD DE LATACUNGA, MEDIANTE EL ANALISIS DE LA METODOLOGIA ICONIX.", 2015.
- [18] A. El-Abbassy y M. El-Zeweidy, "ICONIX Approach to MVC: Applying Robustness Analysis on the Model-View-Controller Architecture", *Journal of the ACS*, vol. 7, 2013.
- [19] T. Rozman, R. V. Horvat, y I. Rozman, "Using ICONIX unified object modelling approach for software tools development", 2000, Consultado: el 10 de mayo de 2022. [En línea]. Disponible en: <https://www.researchgate.net/publication/280521255>
- [20] D. Rosenberg, M. Stephens, y M. Collins-Cope, *Agile Development with ICONIX Process*. A-Press, 2005. doi: 10.1007/978-1-4302-0009-3.
- [21] D. Rosenberg y M. Stephens, "Use case driven object modeling with UML: Theory and Practice", *Use Case Driven Object Modeling with UML: Theory and Practice*, pp. 1–438, 2007, doi: 10.1007/978-1-4302-0369-8_1.
- [22] M. B. Burgos Romero y D. A. T. Huaman Saavedra, "IMPLEMENTACIÓN DE UN CHATBOT, UTILIZANDO LA METODOLOGÍA ICONIX PARA MEJORAR EL PROCESO DE VENTAS EN LA EMPRESA EAC STEEL E.I.R.L.", 2019.
- [23] Q. Ma, "System requirements analysis with ICONIX process case study: group project peer-assessment tool Teaching case for System Analysis View project Technology Acceptance Model View project", 2019, Consultado: el 11 de mayo de 2022. [En línea]. Disponible en: <http://www.aabri.com/copyright.html>
- [24] R. A. Rodríguez, P. M. Vera, M. Roxana Martínez, F. A. Parra Beltrán, A. Trigueros, y M. G. Dogliotti, "Aplicaciones Web Progresivas Impulsadas por el Avance de los Estándares Web", 2019, Consultado: el 28 de abril de 2022. [En línea]. Disponible en: <http://sedici.unlp.edu.ar/handle/10915/77181>
- [25] B. Tello y P. Cristhian, "EVALUACIÓN DE RENDIMIENTO DE UN SISTEMA WEB DESARROLLADA MEDIANTE LA TECNOLOGÍA DE APLICACIONES WEB PROGRESIVAS", 2021, Consultado: el 28 de abril de 2022. [En línea]. Disponible en: <https://repositorio.uss.edu.pe/handle/20.500.12802/8387>
- [26] P. Thakur, "Evaluation and Implementation of Progressive Web Application", 2018, Consultado: el 9 de mayo de 2022. [En línea]. Disponible en: <https://www.theseus.fi/handle/10024/142997>
- [27] A. Gambhir y G. Raj, "Analysis of Cache in Service Worker and Performance Scoring of Progressive Web Application", *Proceedings on 2018 International Conference on Advances in Computing and Communication Engineering, ICACCE 2018*, pp. 294–299, ago. 2018, doi: 10.1109/ICACCE.2018.8441715.
- [28] D. Hume, *Progressive Web Apps*. 2017.
- [29] S. Qu, Z. Zhang, B. Ma, y V. S. Magomadov, "Exploring the role of progressive web applications in modern web development Optimization Method of Web Fuzzy Test Cases Based on Genetic Algorithm Exploring the role of progressive web applications in modern web development", *J. Phys*, p. 22043, 2020, doi: 10.1088/1742-6596/1679/2/022043.
- [30] J. Eguíluz Pérez, *Introducción a JavaScript*. 2019. Consultado: el 12 de mayo de 2022. [En línea]. Disponible en: www.librosweb.es

- [31] Stack Overflow, “Stack Overflow Developer Survey 2021”, 2021. <https://insights.stackoverflow.com/survey/2021#most-popular-technologies-language-prof> (consultado el 12 de mayo de 2022).
- [32] M. Vorobyov, “JavaScript - Reviews, Pros & Cons | Companies using JavaScript”, 2021. <https://stackshare.io/javascript> (consultado el 12 de mayo de 2022).
- [33] K. Sun y S. Ryu, “Analysis of JavaScript programs: Challenges and research trends”, *ACM Comput Surv*, vol. 50, núm. 4, ago. 2017, doi: 10.1145/3106741.
- [34] Anthony. Nandaa, *Beginning API Development with Node.js : Build Highly Scalable, Developer-Friendly APIs for the Modern Web with JavaScript and Node.js*. Packt Publishing Ltd, 2018.
- [35] M. R. Valarezo Pardo, J. A. Honores Tapia, A. S. Gómez Moreno, y L. F. Vincés Sánchez, “Comparación de tendencias tecnológicas en aplicaciones web”, 2018.
- [36] H. Sun, C. Humer, D. Bonetta, y W. Binder, “Efficient dynamic analysis for node.js”, *CC 2018 - Proceedings of the 27th International Conference on Compiler Construction, Co-located with CGO 2018*, vol. 2018-February, pp. 196–206, feb. 2018, doi: 10.1145/3178372.3179527.
- [37] M. Evelin y B. Santos, “JAVASCRIPT: EL PASADO, PRESENTE Y FUTURO DEL DESARROLLO WEB”, 2021.
- [38] M. Mehra, M. Kumar, A. Maurya, C. Sharma, y Shanu, “MERN Stack Web Development”, *Annals of R.S.C.B.*, 2021. <https://www.annalsofrscb.ro/index.php/journal/article/view/7719/5721> (consultado el 12 de mayo de 2022).
- [39] Express, “Express - Node.js web application framework”, 2022. <https://expressjs.com/> (consultado el 12 de mayo de 2022).
- [40] N. Nguyen, “Development & deployment of a web server as an executable with Node.js, Express.js and Vercel/pkg”, Metropolia University of Applied Sciences, Metropolia, 2022.
- [41] O. Karlsson, “A Performance comparison Between ASP.NET Core and Express.js for creating Web APIs”, Jonkoping University, Jonkoping, 2021.
- [42] Meta Platforms Inc., “React – A JavaScript library for building user interfaces”, *Facebook Open Source*, 2022. <https://reactjs.org/> (consultado el 12 de mayo de 2022).
- [43] P. S. Maratkar y P. Adkar, “React JS-An Emerging Frontend Javascript Library”, 2021, Consultado: el 12 de mayo de 2022. [En línea]. Disponible en: <https://nodejs.org/en/download/>
- [44] Dublin Business School, “Benchmarking React Library: A Developer Perspective”, 2021.
- [45] S. Aggarwal, “Modern Web-Development using ReactJS”, *International Journal of Recent Research Aspects*, vol. 5, pp. 133–137, 2018.
- [46] L. Sujay Vailshery, “Most used web frameworks among developers 2021”, 2022. <https://www.statista.com/statistics/1124699/worldwide-developer-survey-most-used-frameworks-web/> (consultado el 12 de mayo de 2022).
- [47] K. Banker, P. Bakkum, T. Hawkins, S. Verch, y D. Garrett, *MongoDB in Action*. Manning Publications Company, 2016.
- [48] K. Bansal, D. Chauhan, y K. L. Bansal, “Using the Advantages of NOSQL: A Case Study on MongoDB International Journal on Recent and Innovation Trends in Computing and Communication Using the Advantages of NOSQL: A Case Study on MongoDB”, *Article in International Journal on Recent and Innovation Trends in Computing and Communication*, 2017, Consultado: el 12 de mayo de 2022. [En línea]. Disponible en: <http://www.ijritcc.org>
- [49] M. M. Patil, A. Hanni, C. H. Tejeshwar, y P. Patil, “A qualitative analysis of the performance of MongoDB vs MySQL database based on insertion and retrieval operations using a web/android

application to explore load balancing-Sharding in MongoDB and its advantages”, *Proceedings of the International Conference on IoT in Social, Mobile, Analytics and Cloud, I-SMAC 2017*, pp. 325–330, oct. 2017, doi: 10.1109/I-SMAC.2017.8058365.

- [50] F. H. Salcedo López y D. D. Cuenca, “Desarrollo de una aplicación WEB para pedidos en línea en la casa comercial Calva & Calva.”, Universidad Nacional de Loja, Loja, 2009.
- [51] G. P. Villamarín Coronel, G. A. Hidalgo Méndez, y L. B. Samaniego Ramón, “Diseño e implementación de una tienda virtual para la venta de equipos informáticos para la empresa ‘SETCOMPC’ de la ciudad de Loja”, Universidad Nacional de Loja, Loja, 2009.
- [52] C. E. Robledo Arteaga y S. M. Vicuña Espinoza, “Creación de una tienda virtual para compras de abarrotes en línea utilizando metodología de desarrollo SCRUM”, Escuela Superior Politécnica del Litoral, Guayaquil, 2017.
- [53] M. Montes Chunga y Á. Plaza Vargas, “Desarrollo de un software para la creación de tiendas virtuales orientadas a B2C que facilite el ingreso de pequeñas y medianas empresas ecuatorianas al comercio electrónico”, Escuela Superior Politécnica del Litoral, Guayaquil, 2018.
- [54] A. Joshi, S. Kale, S. Chandel, y D. K. Pal, “Likert Scale: Explored and Explained”, *Br J Appl Sci Technol*, vol. BJASt, núm. 4, p. 157, 2015, doi: 10.9734/BJAST/2015/14975.

11. Anexos

Anexo 1. Entrevista para la obtención de requerimientos



Universidad
Nacional
de Loja



Área de la Energía, las Industrias y los Recursos Naturales No Renovables

DATOS GENERALES

Acta de Entrevista					
Código:	E-PR-01		Versión:	1	
Fecha:	Mayo 02, 2022	Hora Inicio:	15h00	Hora Fin:	17h00
Lugar:	Ciudad de Loja				
Entrevistador:	Víctor Andrés Rojas Curimilma				
Entrevistada/o	Ing. Mario Enrique Cueva Hurtado				

DESCRIPCIÓN

Introducción

La presente entrevista tiene el objetivo de levantar requerimientos sobre las necesidades actuales de Compumars en relación a sus procesos de gestión y venta de productos informáticos y la automatización del mismo. Para ello se cita al ing. Mario Enrique Cueva Hurtado, gerente propietario de la empresa mencionada, experto en el área.

Desarrollo

- ¿Quiénes serán los usuarios del sistema?**
Los usuarios son: Administrador, Cajero, Bodeguero, Operador y Cliente.
- ¿Qué datos deben registrarse de los usuarios?**
Los datos que son importantes son: nombres, apellidos, dirección de correo electrónico y clave, esto para los colaboradores de Compumars, para los clientes, se necesita, adicional, la cédula, dirección física y número de celular.
- ¿Quién debe registrar los usuarios?**
Las cuentas de administrador, cajero, bodeguero y operador las debería registrar un administrador. Los clientes, a diferencia, deberían poder crear sus propias cuentas.
- ¿Cuáles son las responsabilidades de cada usuario?**
Las responsabilidades son:



unl

Universidad
Nacional
de Loja



Centro de Ingeniería en
Sistemas / Computación

Área de la Energía, las Industrias y los Recursos Naturales No Renovables

Administrador: es el encargado de crear cuentas de usuario, asignar roles y dar de baja cuentas.

Bodeguero: es el encargado de registrar actualizaciones en los productos, ya sea registrar un nuevo producto, modificar un producto existente, asignar el inventario de cada producto o dar de baja un producto existente.

Cajero: se encarga de registrar las ventas en el local, registrar el pago y el comprobante del mismo.

Operador: es quien da seguimiento a los pedidos que realizan los clientes por medio de marketplace (función de ventas a través de Facebook).

5. ¿Qué datos se debe registrar de los productos?

De los productos se debe registrar: código, nombre, marca, modelo, color, descripción, tamaño, foto, precio, código auxiliar, categoría, código de barras.

6. ¿Cómo se debe gestionar el inventario?

Para gestionar el inventario se debe registrar la entrada de nuevos productos al almacén mediante las órdenes de compra y, por otro lado, para las salidas de productos las órdenes de venta llevan a cabo este registro. En sí, estás órdenes son las que se encargan de actualizar la disponibilidad de los productos. Si al registrar una entrada el producto no está registrado en la lista, se debe primero registrar el producto y después actualizar la disponibilidad.

7. ¿Cómo deben gestionarse las cuentas de los clientes?

Las cuentas de los clientes deben ser registradas por ellos mismos, luego deben acceder al sistema y tener la posibilidad de actualizar su información.

8. ¿Cómo se debe generar una orden de venta?

Para las órdenes en línea: el cliente puede navegar por los productos y categorías que haya disponibles o vigentes, ver detalles y precios, seleccionar los que quiera comprar, agregarlos a su lista de compras y finalmente realizar la orden. Una vez que se haya concretado la orden se continúa al pago, entonces aquí debe ingresar una referencia que identifique el comprobante de la transferencia bancaria. Luego el operador verifica el pago registrado, genera el comprobante de venta de la orden y acuerdan la entrega.

Para las ventas en el local físico de Compumars: el cliente revisa los productos que quiere comprar, decide los que llevará, realiza el pago al cajero y este se encarga de registrar en el sistema: la orden, el pago y el comprobante de venta del mismo.

9. ¿De qué datos constan en las órdenes de venta?

Las órdenes de venta deben tener información de los productos que está adquiriendo el cliente, así como la información de este último. Además, debe poseer: un número de referencia de la orden, el estado actual, costo de envío, subtotal, subtotal con impuestos incluidos y el total a pagar.



unl

Universidad
Nacional
de Loja



Centro de Ingeniería en
Sistemas / Computación

Área de la Energía, las Industrias y los Recursos Naturales No Renovables

Una vez concretada la orden, se procede a asociar el pago y el comprobante de venta a la misma.

10. ¿De qué datos constan en las órdenes de compra?

Por su lado, estas órdenes deben tener información de los productos que está entregando el proveedor, así como la información de este último. Asimismo, esta orden debe poseer los mismos datos adicionales que posee la orden de venta.

11. ¿En qué estados puede encontrarse una orden de venta?

La orden de venta puede estar en 5 estados:

Por pagar: A la espera de que el cliente registre una referencia de la constancia del pago realizado.

Cancelada: La orden no llegó a concretarse, ya sea porque el cliente no registró la referencia del pago en tiempo establecido o porque se comprobó que el pago no es válido.

Por verificar pago: Se ha registrado el pago de la orden, y se encuentra a la espera de que se valide por parte de un miembro de la empresa.

En proceso: La orden se está preparando para ser entregada y se está generando el comprobante de venta.

Completada: La orden se ha concretado correctamente.

12. ¿Cómo se debe verificar los pagos?

En el caso de las compras en el local, el cajero registra inmediatamente el pago. Para los pedidos en línea, el pago debe ser una transferencia bancaria verificada por el operador mediante la referencia de la constancia del pago registrada por el cliente.

13. ¿Qué se debe registrar del pago?

Para el pago debe solicitar: el método de pago: efectivo o transferencia bancaria, el monto a pagar, fecha de registro y una referencia que indique la constancia del pago.

14. ¿Qué se debe registrar del comprobante de venta?

Para el comprobante de venta se necesita: el tipo de comprobante que puede ser factura o nota de venta, el número de documento, la fecha de emisión, números de serie de los productos asociados a la orden y los datos del receptor de la orden.

Actualmente la facturación se realiza sobre el mismo sistema del SRI, se requiere información del cliente y detalles de la compra. Por lo cual el sistema actual debe únicamente guardar el registro del número de factura que genera el sistema del SRI en el número de documento del comprobante de venta.

15. ¿Las órdenes se pueden cancelar?

Se debería cancelar las órdenes dentro de un tiempo que el cliente no haya registrado el pago correspondiente a una orden. De igual manera el operador puede cancelar una orden si al verificar un pago confirma que este no es válido.

16. ¿Cómo se debe gestionar los servicios?



unl

Universidad
Nacional
de Loja



Centro de Ingeniería en
Sistemas / Computación

Área de la Energía, las Industrias y los Recursos Naturales No Renovables

Los servicios se deben registrar como un tipo de producto más dentro del sistema, solamente que estos no tienen un registro de existencias dentro del inventario. Lo importante es guardar datos como el nombre del servicio y una descripción del mismo.

Observaciones (Opcional)

FIRMAS



Firmado digitalmente por:
**VICTOR ANDRES
ROJAS CURIMILMA**

Entrevistador
Víctor Andrés Rojas Curimilma



Firmado
digitalmente por
**MARIO ENRIQUE
CUEVA HURTADO**

Entrevistada/o
Ing. Mario Enrique Cueva Hurtado

Anexo 2. Aprobación de los requerimientos de software



Facultad de la Energía, las Industrias y los Recursos Naturales no Renovables

Aprobación de Requerimientos de software

Nombre del proyecto: Desarrollo de una Aplicación Web Progresiva para la gestión y comercialización de productos y servicios para la Empresa “Compumars” de la ciudad de Loja.

Nombre del desarrollador: Víctor Andrés Rojas Curimilma

Aprobación:



Firmado
digitalmente por
MARIO ENRIQUE
CUEVA HURTADO

Ing. Mario Enrique Cueva Hurtado

Gerente-Propietario de Compumars

Anexo 3. Modelado de arquitectura física y lógica en base a los requerimientos del usuario utilizando el Modelamiento de desarrollo de software dirigido por ICONIX

1. Modelado de Requerimientos

1.1. Glosario de términos

El sistema debe ser capaz de administrar los siguientes conceptos:

- **Administrador;** rol encargado de la gestión de usuarios con sus respectivos roles (cajero, bodeguero, operador y administrador). Para estos usuarios se debe almacenar la siguiente información: nombre, apellido, dirección de correo electrónico y clave.
- **Cliente;** rol que tiene la facultad de navegar por las categorías de productos, realizar órdenes en línea y gestionar su propia cuenta de usuario. Asimismo, puede registrar su propia cuenta de usuario con los siguientes datos: nombre, apellido, dirección de correo electrónico, cédula/RUC, dirección, clave, celular.
- **Cajero;** rol encargado de registrar ventas inmediatas (desde la empresa misma) sin previo pedido desde el servicio en línea.
- **Bodeguero;** rol encargado de registrar el ingreso de productos por adquisición, así como registrar proveedores. Los datos para registrar un nuevo proveedor son: nombre, apellido, razón social, dirección de correo electrónico, cédula/RUC, dirección, celular y teléfono de contacto.
- **Operador;** rol encargado de dar seguimiento a las órdenes en línea, así como validar pagos realizados por clientes y registrar el comprobante de venta asociado a la orden cuando el pago ha sido validado con éxito.
- **Inventario;** es el registro de la información de los productos que existen en bodega, en donde se registran datos como: cantidad disponible en el inventario, cantidad reservada por una orden, cantidad física existente, precio de venta del producto, impuesto del producto y la información del producto asociado.
- **Producto;** es un artículo específico que posee la empresa, el cual va a ser comercializado y posee la siguiente información: código, nombre, marca, modelo, color, descripción, tamaño, foto, precio, código auxiliar, categoría, código de barras.
- **Orden de venta;** es un tipo de orden realizada para los clientes y representa la salida de productos que existen en bodega, en donde cada una de estas

órdenes almacena: código de referencia de la orden, estado de la orden, información del cliente (nombre, apellido, cédula/RUC, número de celular, dirección de correo electrónico, dirección de envío, dirección de comprobante de venta), información de los productos (código del producto, precio, cantidad), subtotal, costos de envío, impuestos, total.

- **Orden de compra;** es un tipo de orden que representa la entrada de productos en bodega, en donde cada una de estas órdenes almacena: código de referencia de la orden, información del proveedor (nombre, apellido, razón social, cédula/RUC, número de celular, dirección de correo electrónico, dirección), información de los productos (código del producto, precio, cantidad), subtotal, impuestos, total.
- **Pago;** Es una constancia que representa la cancelación de una orden, este pago puede ser por depósito o transferencia bancaria o efectivo y posee una referencia que indica la constancia del pago, la cual debe ser validada por el rol operador.
- **Servicio;** un servicio para la empresa Compumars es un tipo de producto más, pero que no posee un registro en el inventario. Un servicio destaca los datos de: nombre del servicio y descripción.

1.2. Requerimientos Funcionales

El sistema permitirá a **todos los usuarios:**

Tabla 23. Requerimientos funcionales de Todos los usuarios

Código	Descripción	Tipo
RF01	Acceder a su cuenta de usuario mediante sus credenciales como dirección de correo electrónico y clave	Evidente
RF02	Actualizar su clave de acceso	Inferenciado
RF03	Modificar su perfil de usuario (información personal)	Inferenciado
RF04	Recuperar su clave de acceso	Inferenciado

El sistema permitirá al **administrador:**

Tabla 24. Requerimientos funcionales del Administrador

Código	Descripción	Tipo
RF05	Crear nuevas cuentas de usuario con la información de: nombre, apellido, dirección de correo electrónico y clave.	Evidente
RF06	Gestionar (agregar, modificar) los siguientes roles (administrador, cajero, bodeguero y operador) a las cuentas de usuario.	Evidente
RF07	Buscar cuentas de usuarios registradas a través de nombres, apellidos o dirección de correo electrónico.	Inferenciado
RF08	Dar de baja a cuentas de usuario.	Evidente

El sistema permitirá al **cliente**:

Tabla 25. Requerimientos funcionales del Cliente

Código	Descripción	Tipo
RF09	Registrar su cuenta de usuario. Los datos para el registro son: nombre, apellido, dirección de correo electrónico, cédula/RUC, dirección, clave, celular. Su cuenta de usuario se registrará por defecto con el rol de Cliente.	Evidente
RF10	Buscar productos a través de categorías.	Evidente
RF11	Agregar productos al carrito de compras.	Evidente
RF12	Modificar cantidad de productos en el carrito de compras.	Inferenciado
RF13	Quitar productos del carrito de compras.	Inferenciado
RF14	Realizar órdenes de compra en línea (para el sistema será representado a través de órdenes de venta) a través del carrito de compras. Estas órdenes están compuestas por información del cliente (nombre, apellido, cédula/RUC, número de celular, dirección de correo electrónico, dirección de envío, dirección de comprobante de venta), información de los productos (código del producto, precio, cantidad), subtotal, costos de envío, impuestos, total.	Evidente
RF15	Registrar el pago para las órdenes de compra. Este registro posee la referencia que indica la constancia del pago y la forma de pago será por medio de transferencia bancaria.	Evidente
RF16	Visualizar el estado de sus órdenes de compra, el estado de la orden será: Por pagar, Cancelada, Por verificar pago, En proceso, Completado.	Inferenciado

El sistema permitirá al **cajero**:

Tabla 26. Requerimientos funcionales del Cajero

Código	Descripción	Tipo
RF17	Registrar órdenes por concepto de venta, éstas se encuentran compuestas por: estado actual de la orden, datos del cliente (nombre, apellido, cédula/RUC, número de celular, dirección de correo electrónico, dirección de envío, dirección de comprobante de venta), información de los productos (código del producto, precio, cantidad), subtotal, impuestos, total.	Evidente
RF18	Buscar ventas registradas a través del código de referencia de la orden.	Inferenciado

El sistema permitirá al **bodeguero**:

Tabla 27. Requerimientos funcionales del Bodeguero

Código	Descripción	Tipo
RF19	Registrar productos. Los datos de un nuevo producto son: código, nombre, marca, modelo, color, descripción, tamaño, foto, precio, código auxiliar, código de barras.	Evidente
RF20	Gestionar categorías (Crear, Modificar).	Inferenciado
RF21	Asignar productos a categorías.	Inferenciado
RF22	Buscar productos a través del código de producto, código auxiliar o nombre del producto.	Inferenciado
RF23	Modificar detalles de los productos.	Evidente
RF24	Dar de baja productos.	Evidente
RF25	Registrar órdenes por concepto de compra. Estas órdenes están compuestas por: datos del proveedor (nombre, apellido, razón social, dirección de correo electrónico, cédula/RUC, dirección, celular), información de los productos (código del producto, precio, cantidad), subtotal, impuestos, total, pago y comprobante de venta.	Evidente
RF26	Registrar un nuevo proveedor. Los datos a registrar son: nombre, apellido, razón social, dirección de correo electrónico, cédula/RUC, dirección, celular y teléfono de contacto.	Inferenciado

El sistema permitirá al **operador**:

Tabla 28. Requerimientos funcionales del Operador

Código	Descripción	Tipo
RF27	Buscar órdenes de compra (para el sistema será representado a través de órdenes de venta) realizadas en línea por el cliente.	Evidente
RF28	Visualizar pagos de órdenes de compra realizadas en línea por el cliente.	Inferenciado
RF29	Validar pagos de órdenes de compra realizadas en línea por el cliente.	Evidente
RF30	Generar el comprobante de venta cuando un pago de la orden de compra ha sido validado.	Evidente

1.3. Requerimientos no funcionales

El sistema deberá:

Tabla 29. Requerimientos no funcionales

Código	Descripción	Categoría
RNF01	Proveer tiempos de respuesta accesibles (entre 2 a 5 segundos) por cada petición u operación	Tiempos de respuesta
RNF02	Permitir que cada rol de usuario tenga sus propios privilegios de acceso a las distintas funcionalidades de la aplicación	Autorización
RNF03	Encriptar la clave de cada usuario antes de su almacenamiento en la base de datos	Seguridad
RNF04	Estar disponible las 24 horas de cada día	Disponibilidad
RNF05	Poseer la facultad de ser utilizada en cualquier navegador web. Y gracias a las características de las PWA también podrá ser instalada y usada en sistemas operativos móviles IOS o Android.	Tipo de aplicación
RNF06	Tener una interfaz amigable e intuitiva para facilitar las tareas del usuario	Interfaz gráfica

2. Modelado del diagrama del dominio

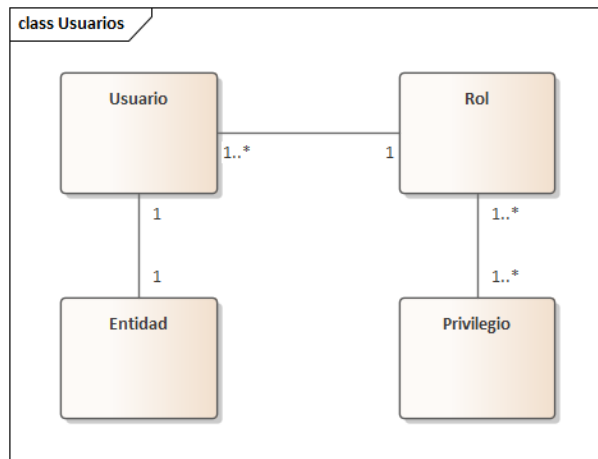


Figura 63. Diagrama del dominio de Usuarios

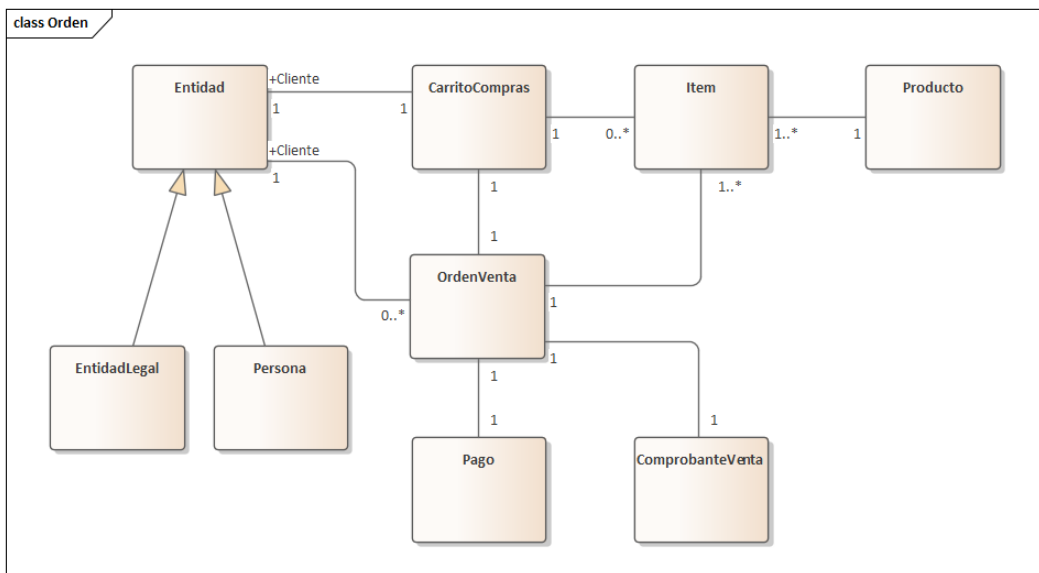


Figura 64. Diagrama del dominio de Orden

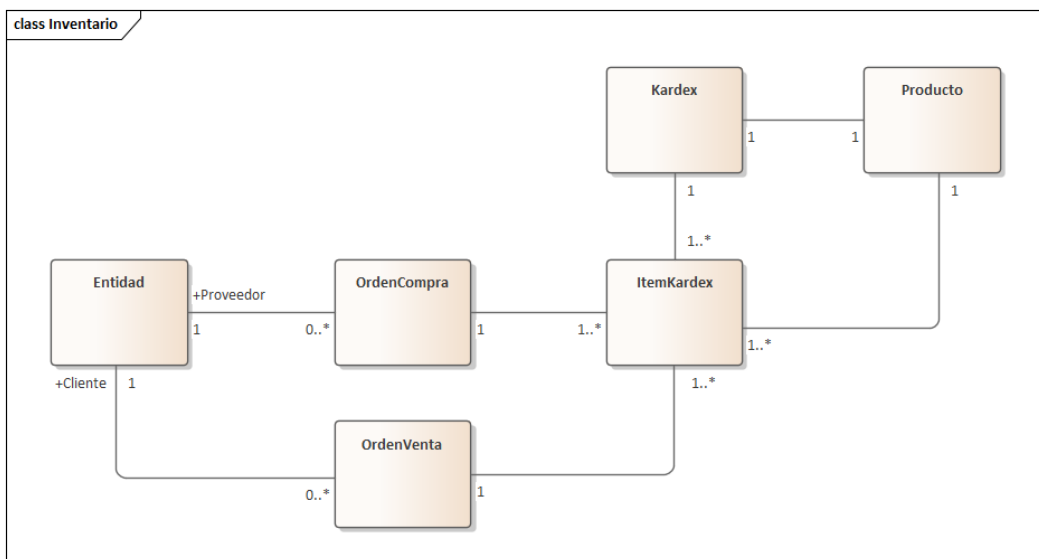


Figura 65. Diagrama del dominio de Inventario

3. Modelado de casos de uso

El diagrama consta de cinco actores: Administrador, Cliente, Cajero, Operador y Bodeguero.

3.1. Caso de uso General

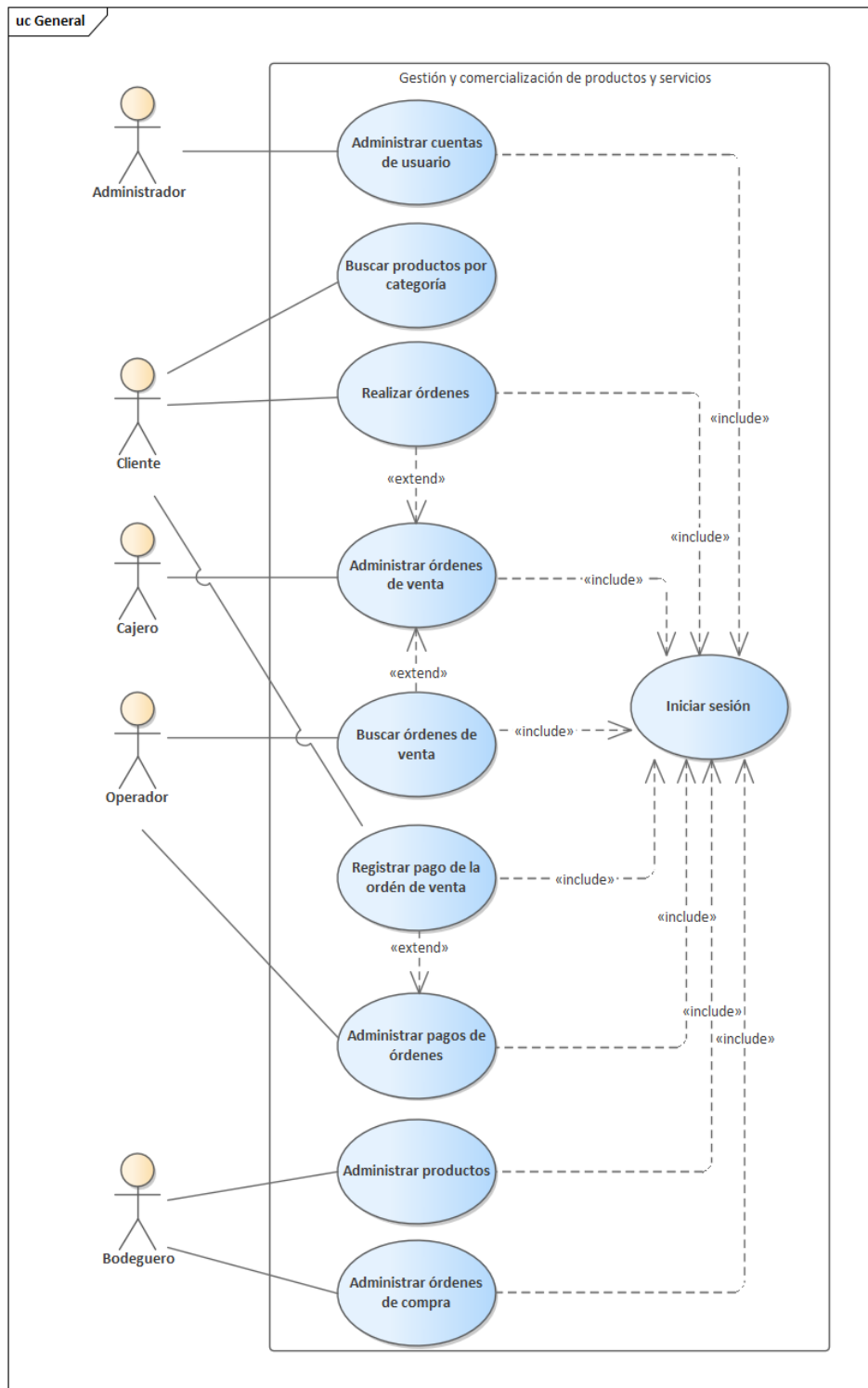


Figura 66. Diagrama de casos de uso General

3.2. Casos de uso Explotados

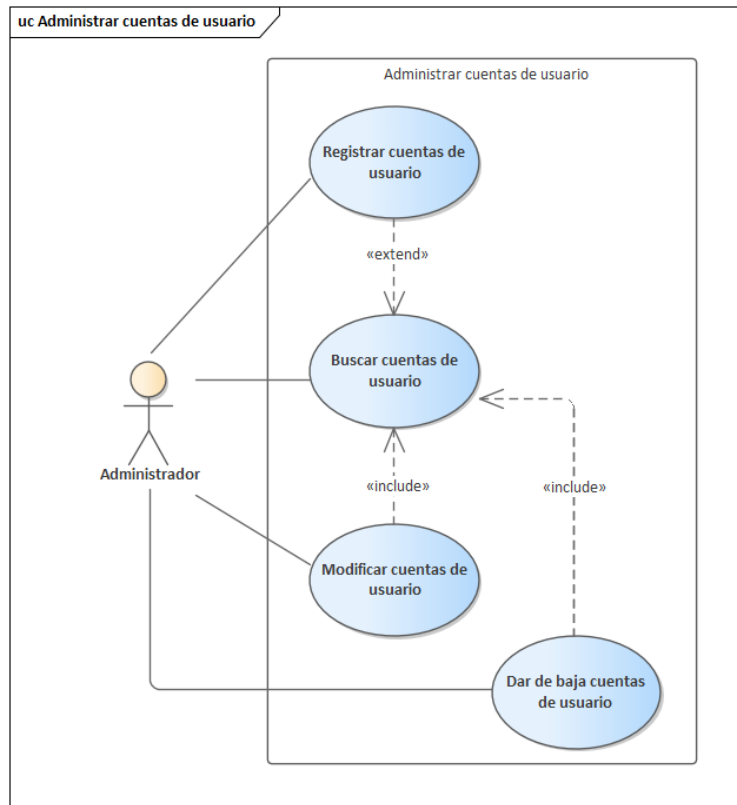


Figura 67. Diagrama de casos de uso de Administrar cuentas de usuario

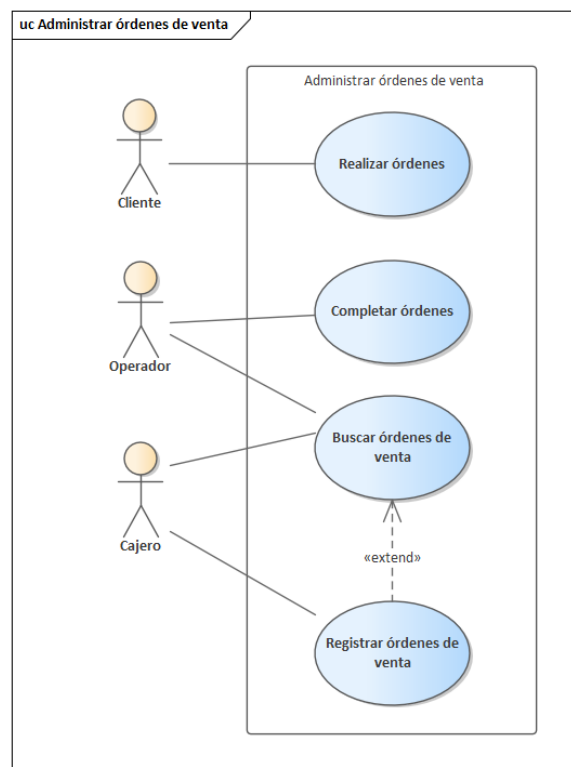


Figura 68. Diagrama de casos de uso de Administrar órdenes de venta

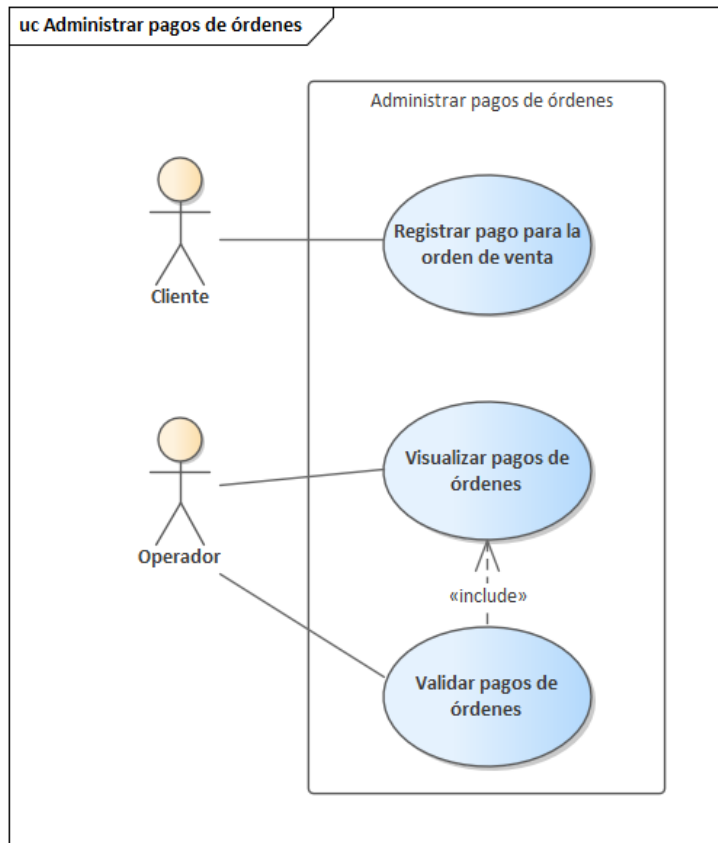


Figura 69. Diagrama de casos de uso de Administrar pagos de órdenes

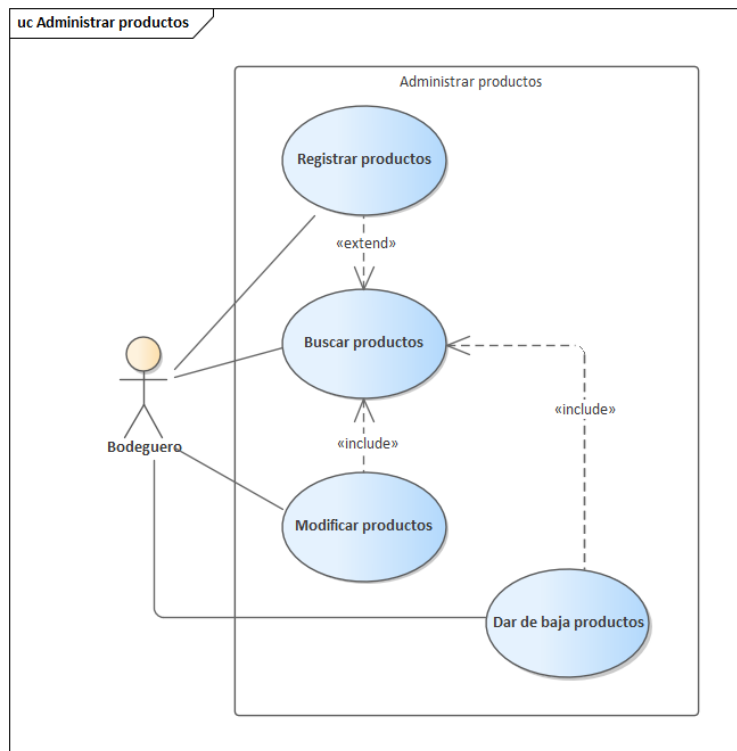


Figura 70. Diagrama de casos de uso de Administrar productos

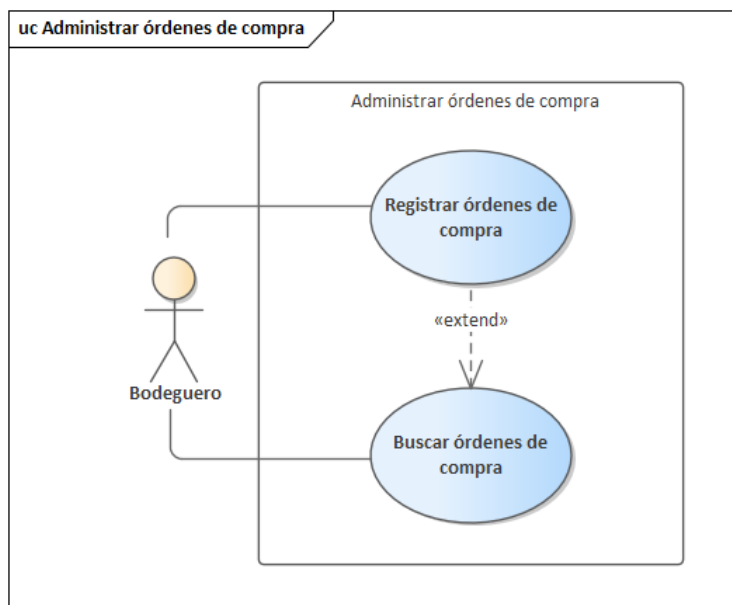


Figura 71. Diagrama de casos de uso de Administrar órdenes de compra

3.3. Especificación de casos de uso

3.3.1. General

Tabla 30. Especificación del caso de uso Iniciar Sesión

Identificador	CU01
Nombre	Iniciar Sesión
Actores	Administrador, Cliente, Cajero, Operador, Bodeguero
Descripción	Autenticación del usuario en el sistema
Precondiciones	<ul style="list-style-type: none"> a. Haberse registrado en el sistema b. Encontrarse en la página "Iniciar Sesión"
Flujo Principal	<ol style="list-style-type: none"> 1. El sistema muestra el formulario de Iniciar sesión 2. El usuario ingresa sus credenciales de acceso (dirección de correo electrónico y clave) 3. El usuario pulsa el botón "Iniciar sesión" del formulario 4. El sistema valida la información ingresada en el formulario 5. El sistema autentica al usuario y carga su información y privilegios 6. El caso de uso finaliza
Flujo Alternativo	<ul style="list-style-type: none"> a. Campos obligatorios vacíos <ol style="list-style-type: none"> 4.1. El sistema debe mostrar un mensaje indicando que existen campos obligatorios vacíos b. Credenciales de acceso no válidas <ol style="list-style-type: none"> 5.1. El sistema debe mostrar un mensaje que dice: Credenciales de acceso no válidas

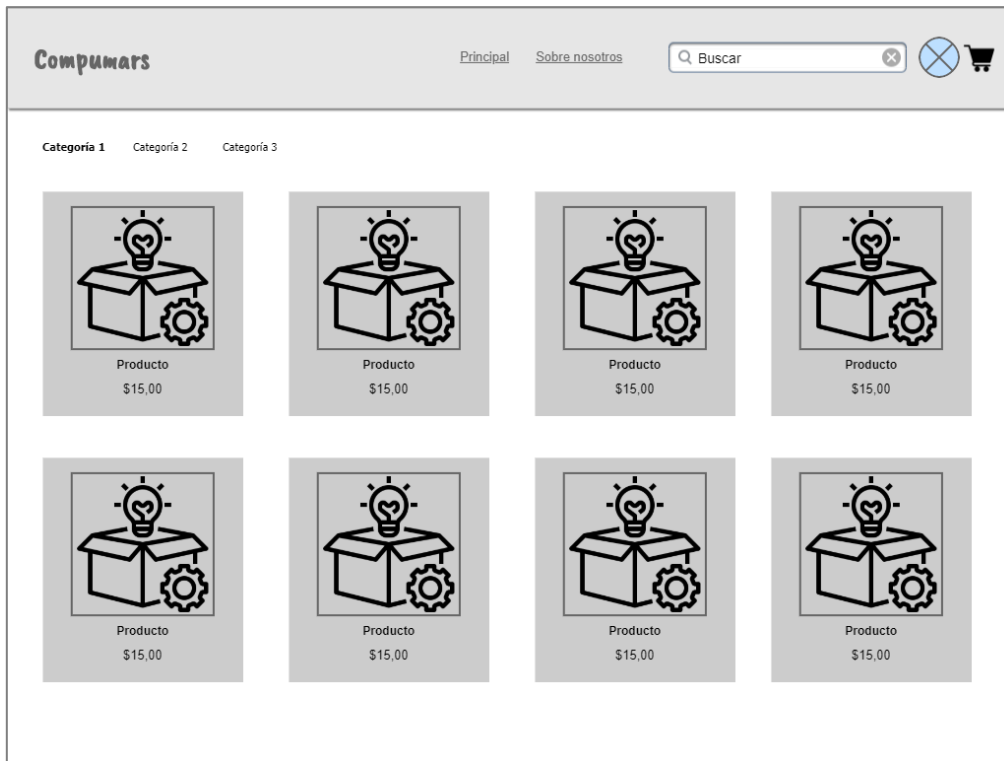


Figura 72. Wireframe de Buscar productos por categoría

Tabla 31. Especificación del caso de uso Buscar productos por categoría

Identificador	CU02
Nombre	Buscar productos por categoría
Actores	Cliente
Descripción	El cliente navega por las categorías de productos.
Precondiciones	
Flujo Principal	<ol style="list-style-type: none"> 1. El sistema mostrará en la Página principal la lista de productos principales y las categorías existentes. 2. El cliente pulsa una categoría 3. El sistema mostrará una lista de todos los productos pertenecientes a esa categoría 4. El caso de uso finaliza
Flujo Alternativo	

3.3.2. Administrar cuentas de usuario

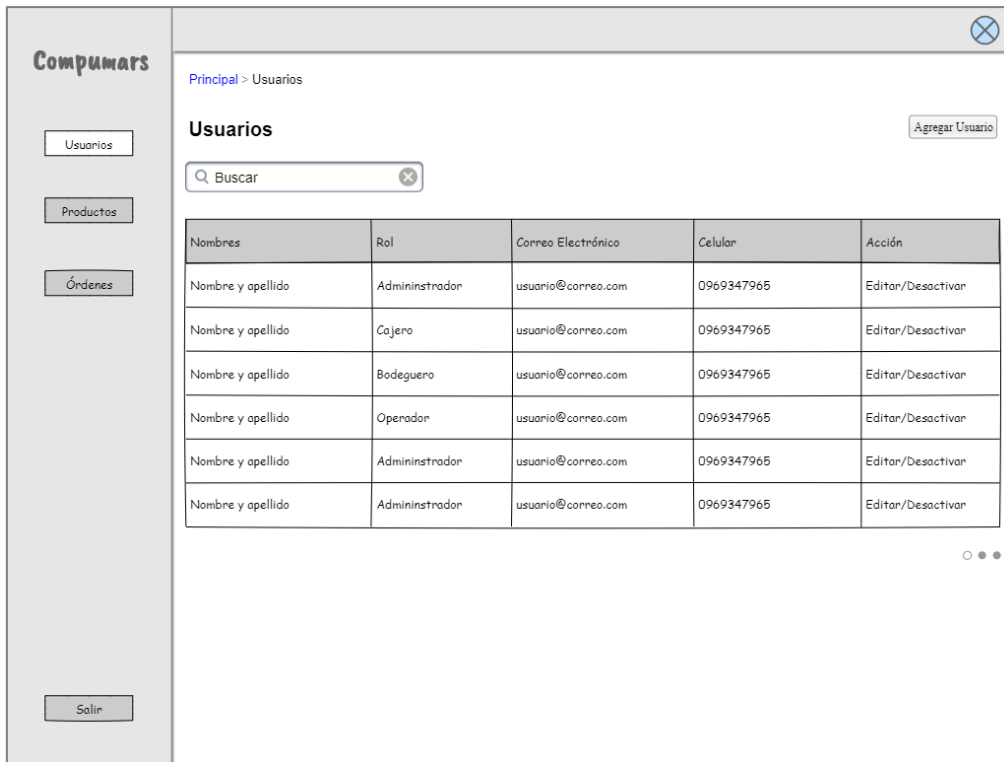


Figura 73. Wireframe de Administrar cuentas de usuario

Tabla 32. Especificación del caso de uso Buscar cuentas de usuario

Identificador	CU03
Nombre	Buscar cuentas de usuario
Actores	Administrador
Descripción	El administrador busca una cuenta de usuario dentro de la lista de usuarios registrados
Precondiciones	a. Administrador autenticado
Flujo Principal	<ol style="list-style-type: none"> 1. El administrador accede a la página “Cuentas de usuario” 2. El administrador ingresa el texto (dirección de correo electrónico, nombres o apellidos) que desea buscar 3. El administrador presiona el botón Buscar 4. El sistema muestra el listado de todas las cuentas de usuario que coincidan con el texto a buscar en la tabla de la página Cuentas de usuario 5. El caso de uso finaliza
Flujo Alternativo	<p>a. Registrar nueva cuenta de usuario</p> <p>1.1. Continúa con la extensión del caso de uso Registrar cuentas de usuario</p> <p>b. Modificar cuenta de usuario</p> <p>4.1. Continúa con la inclusión del caso de uso Modificar cuentas de usuario</p> <p>c. Dar de baja cuenta de usuario</p>

	4.1. Continúa con la inclusión del caso de uso Dar de baja cuentas de usuario
--	-------------------------------------------------------------------------------

Tabla 33. Especificación del caso de uso Registrar cuentas de usuario

Identificador	CU04
Nombre	Registrar cuentas de usuario
Actores	Administrador
Descripción	El administrador registra un nuevo usuario
Precondiciones	<ul style="list-style-type: none"> a. Administrador autenticado b. Encontrarse en la página Cuentas de usuario c. Es una extensión del caso de uso Buscar cuentas de usuario
Flujo Principal	<ol style="list-style-type: none"> 1. El administrador pulsa el botón “Registrar cuenta de usuario” de la página Cuentas de usuario 2. El sistema muestra el formulario de nueva cuenta de usuario 3. El administrador registra los datos de la cuenta de usuario 4. El administrador le asigna un rol a la cuenta de usuario 5. El administrador presiona el botón Guardar del formulario 6. El sistema valida la información ingresada en el formulario 7. El sistema verifica que la dirección de correo electrónico del usuario ingresado no existe 8. El sistema crea el nuevo usuario y muestra un mensaje indicando: Usuario creado correctamente 9. El sistema redirige al administrador a la página “Cuentas de usuario” 10. El caso de uso finaliza
Flujo Alternativo	<ul style="list-style-type: none"> a. Campos obligatorios vacíos 6.1. El sistema debe mostrar un mensaje indicando que existen campos obligatorios vacíos b. Cuenta de usuario ya existe 7.1. El sistema debe mostrar un mensaje indicando que la cuenta de usuario ingresada ya se encuentra registrada

Tabla 34. Especificación del caso de uso Modificar cuentas de usuario

Identificador	CU05
Nombre	Modificar cuentas de usuario
Actores	Administrador
Descripción	El administrador modifica una cuenta de usuario dentro de la lista de usuarios registrados
Precondiciones	<ul style="list-style-type: none"> a. Administrador autenticado b. Encontrarse en la página Cuentas de usuario

	c. Incluye el caso de uso Buscar cuentas de usuario
Flujo Principal	<ol style="list-style-type: none"> 1. El administrador selecciona una cuenta de usuario de la tabla de la página Cuentas de usuario y presiona el botón Modificar 2. El sistema muestra el formulario de modificar cuenta de usuario 3. El administrador registra los datos a modificar de la cuenta de usuario 4. El administrador presiona el botón Guardar del formulario 5. El sistema valida la información ingresada en el formulario 6. El sistema modifica los datos del usuario y muestra un mensaje indicando: Usuario modificado correctamente 7. El caso de uso finaliza
Flujo Alternativo	<p>a. Campos obligatorios vacíos</p> <p>5.1. El sistema debe mostrar un mensaje indicando que existen campos obligatorios vacíos</p>

Tabla 35. Especificación del caso de uso Dar de baja cuentas de usuario

Identificador	CU06
Nombre	Dar de baja cuentas de usuario
Actores	Administrador
Descripción	El administrador da de baja una cuenta de usuario dentro de la lista de usuarios registrados
Precondiciones	<ol style="list-style-type: none"> a. Administrador autenticado b. Encontrarse en la página Cuentas de usuario c. Incluye el caso de uso Buscar cuentas de usuario
Flujo Principal	<ol style="list-style-type: none"> 1. El administrador selecciona una cuenta de usuario de la tabla de la página Cuentas de usuario y presiona el botón Dar de baja 2. El sistema muestra un cuadro de diálogo para confirmar la operación 3. El administrador selecciona la opción Confirmar 4. El sistema da de baja al usuario y muestra un mensaje indicando: El usuario ha sido dado de baja 5. El caso de uso finaliza
Flujo Alternativo	<p>a. Cancelar la operación</p> <p>3.1 El administrador selecciona la opción Cancelar</p> <p>3.2 El sistema cancela la operación</p>

3.3.3. Administrar órdenes de venta

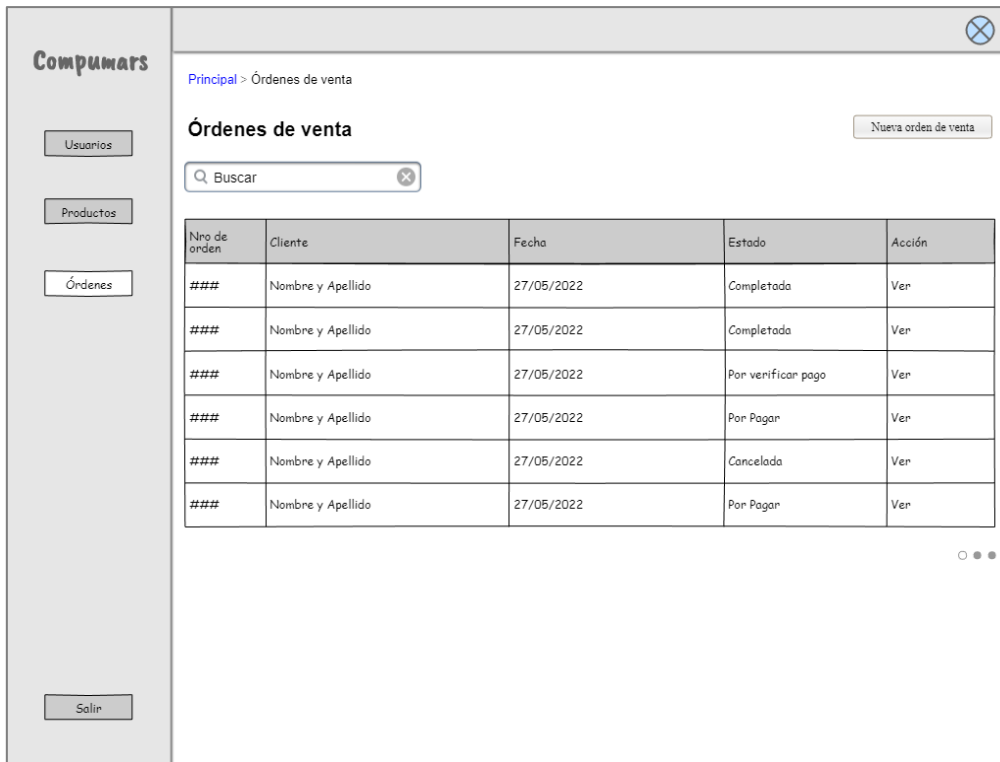


Figura 74. Wireframe de Administrar órdenes de venta

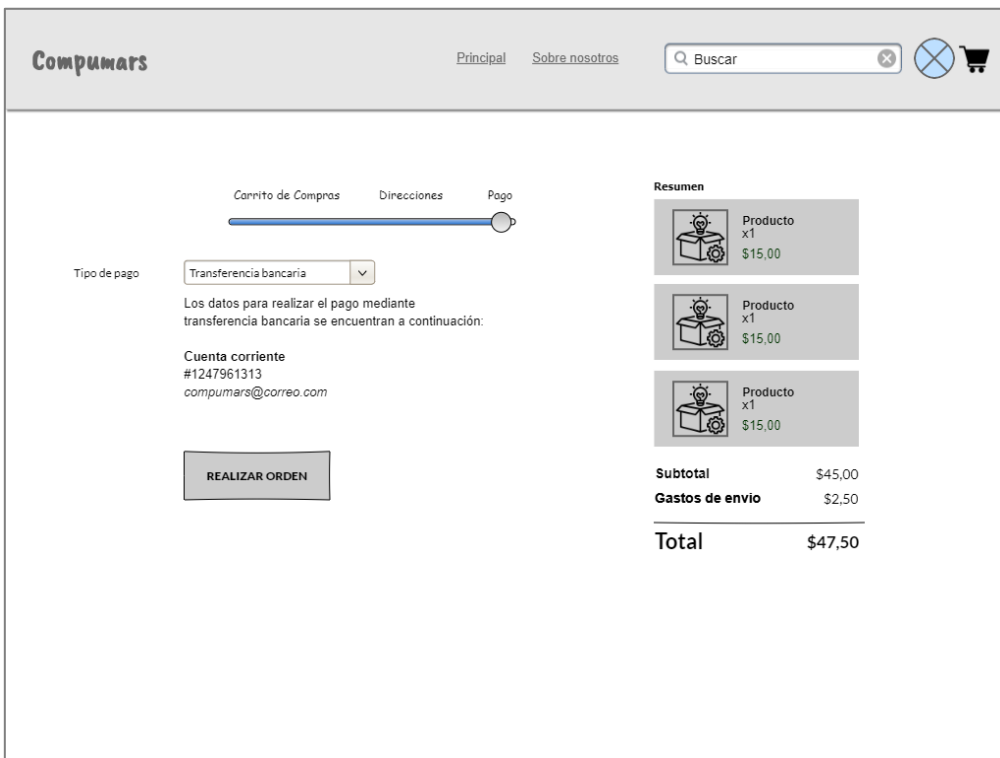


Figura 75. Wireframe de Realizar órdenes

Tabla 36. Especificación del caso de uso Realizar órdenes

Identificador	CU07
Nombre	Realizar órdenes
Actores	Cliente
Descripción	El cliente realiza una nueva orden en base a los productos agregados en su carrito de compras
Precondiciones	a. Cliente autenticado
Flujo Principal	<ol style="list-style-type: none"> 1. El cliente añade productos al carrito de compras 2. El cliente pulsa el botón Ver carrito de compras del menú de navegación 3. El sistema muestra el listado de todos los productos añadidos al carrito de compras 4. El sistema muestra el formulario de agregar una dirección de envío y una dirección para el comprobante de venta. 5. El cliente registra los datos de dirección de envío y dirección de comprobante de venta 6. El sistema muestra información acerca del método de pago 7. El cliente presiona el botón Realizar orden 8. El sistema valida la información ingresada en el formulario 9. El sistema crea una nueva orden de venta y muestra un mensaje indicando: Orden creada correctamente 10. El sistema envía la confirmación de la orden al correo electrónico del cliente 11. El sistema muestra los detalles de la orden registrada 12. El caso de uso finaliza
Flujo Alternativo	<p>b. Carrito de compras vacío</p> <p>7.1. El sistema debe deshabilitar el botón Realizar orden ya que no existen productos en el carrito de compras</p> <p>c. Campos obligatorios vacíos</p> <p>8.1. El sistema debe mostrar un mensaje indicando que existen campos obligatorios vacíos</p>

Tabla 37. Especificación del caso de uso Buscar órdenes de venta

Identificador	CU08
Nombre	Buscar órdenes de venta
Actores	Cajero, Operador
Descripción	El usuario busca una orden de venta dentro de la lista de órdenes registradas
Precondiciones	a. Usuario autenticado
Flujo Principal	<ol style="list-style-type: none"> 1. El usuario accede a la página "Órdenes de venta"

	<ol style="list-style-type: none"> 2. El usuario ingresa el texto (código de referencia de la orden) que desea buscar 3. El administrador presiona el botón Buscar 4. El sistema muestra el listado de todas las órdenes de venta que coincidan con el texto a buscar en la tabla de la página Órdenes de venta 5. El caso de uso finaliza
Flujo Alternativo	<ol style="list-style-type: none"> a. Registrar nueva orden de venta <ol style="list-style-type: none"> 1.1. Continúa con la extensión del caso de uso Registrar órdenes de venta b. Visualizar resumen de ventas diarias <ol style="list-style-type: none"> 1.1. Continúa con la extensión del caso de uso Visualizar resumen de ventas diarias

Figura 76. Wireframe de Registrar órdenes de venta

Tabla 38. Especificación del caso de uso Registrar órdenes de venta

Identificador	CU09
Nombre	Registrar órdenes de venta
Actores	Cajero
Descripción	El cajero registra una nueva orden de venta
Precondiciones	<ol style="list-style-type: none"> a. Cajero autenticado b. Encontrarse en la página Órdenes de venta c. Es una extensión del caso de uso Buscar órdenes de venta

Flujo Principal	<ol style="list-style-type: none"> 1. El cajero pulsa el botón “Registrar orden de venta” de la página Órdenes de venta 2. El sistema muestra el formulario de nueva orden de venta 3. El cajero registra los datos de la orden de venta (datos del cliente, dirección de envío y comprobante de venta, lista de productos, información del pago) 4. El cajero presiona el botón Guardar del formulario 5. El sistema valida la información ingresada en el formulario 6. El sistema crea una nueva orden de venta y muestra un mensaje indicando: Orden creada correctamente 7. El sistema envía la confirmación de la orden al correo electrónico del cliente 8. El sistema redirige al cajero a la página “Órdenes de venta” 9. El caso de uso finaliza
Flujo Alternativo	<p>b. Campos obligatorios vacíos</p> <ol style="list-style-type: none"> 5.1. El sistema debe mostrar un mensaje indicando que existen campos obligatorios vacíos

Tabla 39. Especificación del caso de uso Completar órdenes

Identificador	CU10
Nombre	Completar órdenes
Actores	Operador
Descripción	El operador actualiza una orden al estado de Completado, lo que indica que la orden se ha realizado con éxito
Precondiciones	<ol style="list-style-type: none"> a. Operador autenticado b. Encontrarse en la página Ver detalles de una orden
Flujo Principal	<ol style="list-style-type: none"> 1. El operador presiona el botón Actualizar estado. 2. El sistema muestra una ventana emergente solicitando que el operador confirme la operación para completar la orden. 3. El operador presiona el botón Confirmar 4. El sistema establece el estado de la orden en Completada y muestra un mensaje indicando: Orden modificada correctamente 5. El sistema actualiza el stock de los productos asociados a la orden 6. El sistema envía la comprobante en formato pdf al correo electrónico del cliente de la orden 7. El caso de uso finaliza
Flujo Alternativo	<p>a. La orden no se encuentra en el estado Procesando</p> <ol style="list-style-type: none"> 2.1. El sistema mostrara el formulario necesario para primeramente establecer la orden en el estado Procesando. <p>b. Cancelar la operación</p> <ol style="list-style-type: none"> 3.1. El bodeguero selecciona la opción Cancelar 3.2. El sistema cancela la operación

3.3.4. Administrar pagos de órdenes

Figura 77. Wireframe de Administrar pagos de órdenes

Tabla 40. Especificación del caso de uso Registrar pago para la orden de venta

Identificador	CU11
Nombre	Registrar pago para la orden de venta
Actores	Cliente
Descripción	El cliente registra el comprobante de pago para una orden realizada previamente
Precondiciones	<ul style="list-style-type: none"> a. Cliente autenticado b. Haber realizado una orden previamente
Flujo Principal	<ol style="list-style-type: none"> 1. El cliente pulsa el botón Ver órdenes del menú de navegación 2. El sistema muestra el listado de todas las órdenes realizadas por el cliente y marcará aquellas que requieran registrar el pago 3. El cliente selecciona una orden del listado 4. El sistema muestra información acerca de la orden y el formulario de registrar la referencia del pago de la orden 5. El cliente registra los datos del pago de la orden 6. El cliente presiona el botón Guardar del formulario 7. El sistema valida la información ingresada en el formulario 8. El sistema registra el pago de la orden de venta y muestra un mensaje indicando: Pago registrado correctamente 9. El caso de uso finaliza
Flujo Alternativo	b. Pago ya registrado

	<p>5.1. El sistema no mostrará el formulario para registrar la referencia del pago de la orden</p> <p>c. Campos obligatorios vacíos</p> <p>7.1. El sistema debe mostrar un mensaje indicando que existen campos obligatorios vacíos</p>
--	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Tabla 41. Especificación del caso de uso Visualizar pagos de órdenes

Identificador	CU12
Nombre	Visualizar pagos de órdenes
Actores	Operador
Descripción	El operador visualiza el pago de una orden registrada
Precondiciones	a. Operador autenticado
Flujo Principal	<ol style="list-style-type: none"> 1. El operador accede a la página “Órdenes de venta” 2. El sistema muestra el listado de todas las órdenes de venta en la tabla de la página Órdenes de venta 3. El operador selecciona una orden de venta de la tabla de la página Órdenes de venta y presiona el botón Ver detalles 4. El sistema redirige al operador a la página “Ver detalles” de la orden seleccionada 5. El sistema muestra la información principal de la orden y un apartado para visualizar el pago registrado 6. El caso de uso finaliza
Flujo Alternativo	<p>a. Validar el pago de una orden</p> <p>5.1. Continúa con la inclusión del caso de uso Validar pagos de órdenes</p>

Tabla 42. Especificación del caso de uso Validar pagos de órdenes

Identificador	CU13
Nombre	Validar pagos de órdenes
Actores	Operador
Descripción	El operador valida el pago registrado de una orden
Precondiciones	<ol style="list-style-type: none"> a. Operador autenticado b. Encontrarse en la página Ver detalles de una orden c. Incluye el caso de uso Visualizar pagos de órdenes
Flujo Principal	<ol style="list-style-type: none"> 1. El operador verifica que el pago registrado por el cliente es correcto 2. El operador actualiza el estado de la orden a Procesando 3. El sistema muestra el formulario de registro del comprobante de venta asociado a la orden

	<ol style="list-style-type: none"> 4. El operador registra los datos del comprobante de venta (Tipo de documento, número de comprobante, fecha de emisión, número de serie de los productos de la orden) 5. El operador presiona el botón Guardar del formulario 6. El sistema valida la información ingresada en el formulario 7. El sistema establece el estado de la orden en Procesando y muestra un mensaje indicando: Orden modificada correctamente 8. El sistema crea el comprobante de venta de la orden y muestra un mensaje indicando: Comprobante de venta creado correctamente 9. El caso de uso finaliza
<p>Flujo Alterno</p>	<p>b. Pago no válido</p> <ol style="list-style-type: none"> 2.1. El operador actualiza el estado de la orden a Cancelada 2.2. El sistema establece el estado de la orden en Cancelada y muestra un mensaje indicando: Orden cancelada <p>c. Campos obligatorios vacíos</p> <ol style="list-style-type: none"> 6.1. El sistema debe mostrar un mensaje indicando que existen campos obligatorios vacíos

3.3.5. Administrar productos

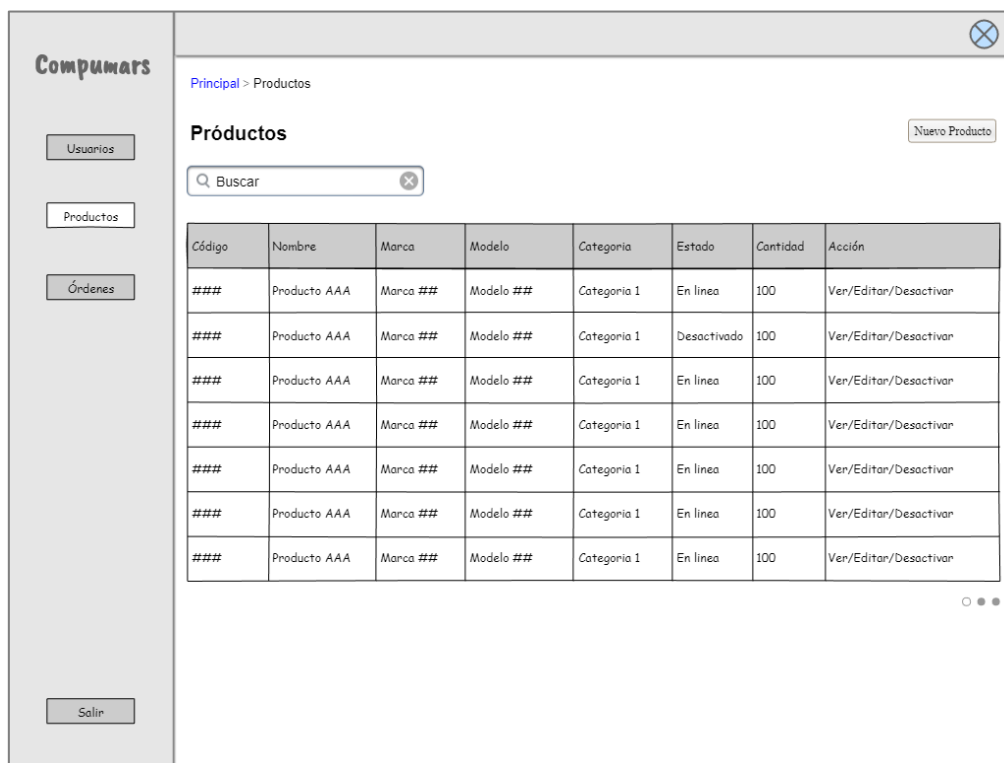


Figura 78. Wireframe de Administrar productos

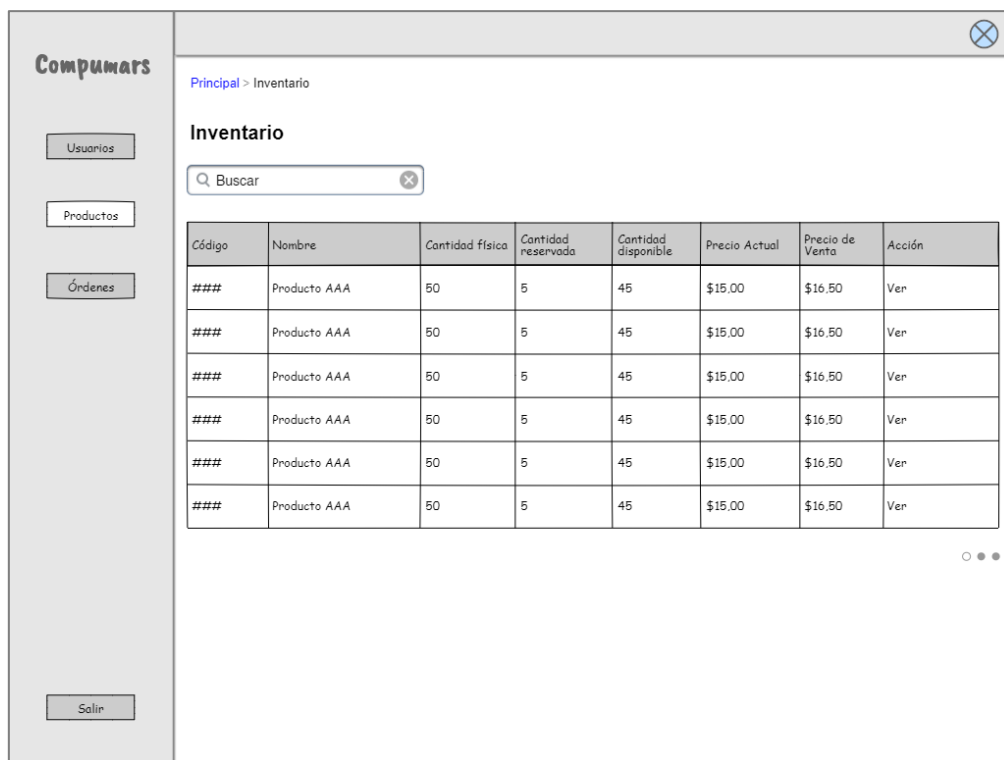


Figura 79. Wireframe de Inventario

Tabla 43. Especificación del caso de uso Buscar productos

Identificador	CU14
Nombre	Buscar productos
Actores	Bodeguero
Descripción	El bodeguero busca un producto dentro de la lista de productos registrados
Precondiciones	a. Bodeguero autenticado
Flujo Principal	<ol style="list-style-type: none"> 1. El bodeguero accede a la página “Productos” 2. El bodeguero ingresa el texto (código de producto, código auxiliar o nombre del producto) que desea buscar 3. El bodeguero presiona el botón Buscar 4. El sistema muestra el listado de todos los productos que coincidan con el texto a buscar en la tabla de la página Productos <ol style="list-style-type: none"> 1. El caso de uso finaliza
Flujo Alternativo	<p>a. Registrar nuevo producto</p> <ol style="list-style-type: none"> 1.1. Continúa con la extensión del caso de uso Registrar productos <p>b. Modificar productos</p> <ol style="list-style-type: none"> 4.1. Continúa con la inclusión del caso de uso Modificar detalles del producto <p>c. Dar de baja productos</p> <ol style="list-style-type: none"> 4.1. Continúa con la inclusión del caso de uso Dar de baja productos

Tabla 44. Especificación del caso de uso Registrar productos

Identificador	CU15
Nombre	Registrar productos
Actores	Bodeguero
Descripción	El bodeguero registra un nuevo producto
Precondiciones	<ul style="list-style-type: none"> a. Bodeguero autenticado b. Encontrarse en la página Productos c. Es una extensión del caso de uso Buscar productos
Flujo Principal	<ol style="list-style-type: none"> 1. El bodeguero pulsa el botón “Registrar producto” de la página Productos 2. El sistema muestra el formulario de nuevo producto 3. El bodeguero registra los datos del producto 4. El bodeguero le asigna una categoría al producto 5. El bodeguero presiona el botón Guardar del formulario 6. El sistema valida la información ingresada en el formulario 7. El sistema verifica que el código del producto ingresado no existe 8. El sistema crea el nuevo producto y muestra un mensaje indicando: Producto creado correctamente 9. El sistema redirige al bodeguero a la página Productos 10. El caso de uso finaliza
Flujo Alternativo	<ul style="list-style-type: none"> a. Campos obligatorios vacíos 6.1. El sistema debe mostrar un mensaje indicando que existen campos obligatorios vacíos b. Producto ya existe 7.1. El sistema debe mostrar un mensaje indicando que el producto ya se encuentra registrado

Tabla 45. Especificación del caso de uso Modificar productos

Identificador	CU16
Nombre	Modificar productos
Actores	Bodeguero
Descripción	El bodeguero modifica un producto dentro de la lista de productos registrados
Precondiciones	<ul style="list-style-type: none"> a. Bodeguero autenticado b. Encontrarse en la página Productos c. Incluye el caso de uso Buscar productos
Flujo Principal	<ol style="list-style-type: none"> 1. El bodeguero selecciona un producto de la tabla de la página Productos y presiona el botón Modificar 2. El sistema muestra el formulario de modificar producto 3. El bodeguero registra los datos a modificar del producto

	<ol style="list-style-type: none"> 4. El bodeguero presiona el botón Guardar del formulario 5. El sistema valida la información ingresada en el formulario 6. El sistema modifica los datos del producto y muestra un mensaje indicando: Producto modificado correctamente 7. El caso de uso finaliza
Flujo Alternativo	<p>a. Campos obligatorios vacíos</p> <p>5.1. El sistema debe mostrar un mensaje indicando que existen campos obligatorios vacíos</p>

Tabla 46. Especificación del caso de uso Dar de baja productos

Identificador	CU17
Nombre	Dar de baja productos
Actores	Bodeguero
Descripción	El bodeguero da de baja un producto dentro de la lista de productos registrados
Precondiciones	<ol style="list-style-type: none"> a. Bodeguero autenticado b. Encontrarse en la página Productos c. Incluye el caso de uso Buscar productos
Flujo Principal	<ol style="list-style-type: none"> 1. El bodeguero selecciona un producto de la tabla de la página Productos y presiona el botón Dar de baja 2. El sistema muestra un cuadro de diálogo para confirmar la operación 3. El bodeguero selecciona la opción Confirmar 4. El sistema da de baja el producto y muestra un mensaje indicando: El producto ha sido dado de baja 5. El caso de uso finaliza
Flujo Alternativo	<p>a. Cancelar la operación</p> <p>3.1. El bodeguero selecciona la opción Cancelar</p> <p>3.2. El sistema cancela la operación</p>

3.3.6. Administrar órdenes de compra

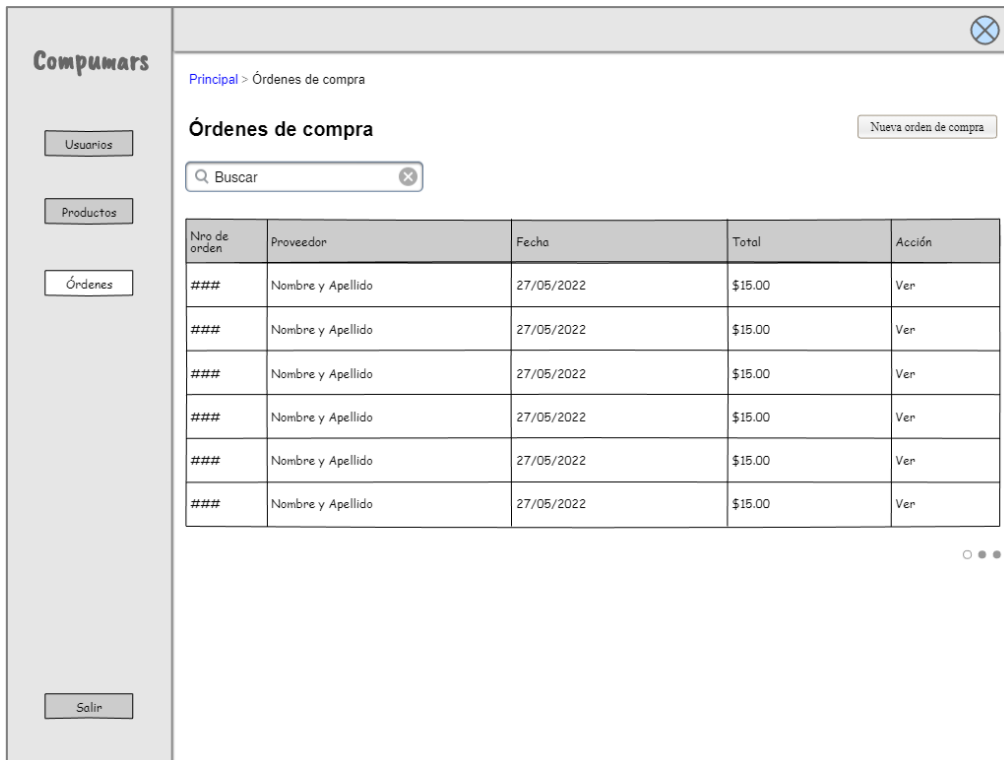


Figura 80. Wireframe de Administrar órdenes de compra

Tabla 47. Especificación del caso de uso Buscar órdenes de compra

Identificador	CU18
Nombre	Buscar órdenes de compra
Actores	Bodeguero
Descripción	El bodeguero busca una orden de compra dentro de la lista de órdenes registradas
Precondiciones	a. Bodeguero autenticado
Flujo Principal	<ol style="list-style-type: none"> 1. El bodeguero accede a la página “Órdenes de compra” 2. El bodeguero ingresa el texto (código de referencia de la orden) que desea buscar 3. El bodeguero presiona el botón Buscar 4. El sistema muestra el listado de todas las órdenes de compra que coincidan con el texto a buscar en la tabla de la página Órdenes de compra 5. El caso de uso finaliza
Flujo Alternativo	<p>a. Registrar nueva orden de compra</p> <ol style="list-style-type: none"> 1.1. Continúa con la extensión del caso de uso Registrar órdenes de compra

Figura 81. Wireframe de Registrar órdenes de compra

Tabla 48. Especificación del caso de uso Registrar órdenes de compra

Identificador	CU19
Nombre	Registrar órdenes de compra
Actores	Bodeguero
Descripción	El bodeguero registra una nueva orden de compra
Precondiciones	<ul style="list-style-type: none"> a. Bodeguero autenticado b. Encontrarse en la página Órdenes de compra c. Es una extensión del caso de uso Buscar órdenes de compra
Flujo Principal	<ol style="list-style-type: none"> 1. El bodeguero pulsa el botón “Registrar orden de compra” de la página Órdenes de compra 2. El sistema muestra el formulario de nueva orden de compra 3. El cajero registra los datos de la orden de compra (datos del proveedor, datos del comprobante de venta, lista de productos) 4. El cajero presiona el botón Guardar del formulario 5. El sistema valida la información ingresada en el formulario 6. El sistema crea una nueva orden de compra y muestra un mensaje indicando: Orden creada correctamente 7. El sistema actualiza el stock de los productos asociados a la orden de compra 8. El sistema redirige al cajero a la página “Órdenes de compra” 9. El caso de uso finaliza
Flujo Alternativo	<p>b. Campos obligatorios vacíos</p> <ol style="list-style-type: none"> 5.1. El sistema debe mostrar un mensaje indicando que existen campos obligatorios vacíos

4. Diagramas de Secuencia

4.1. General

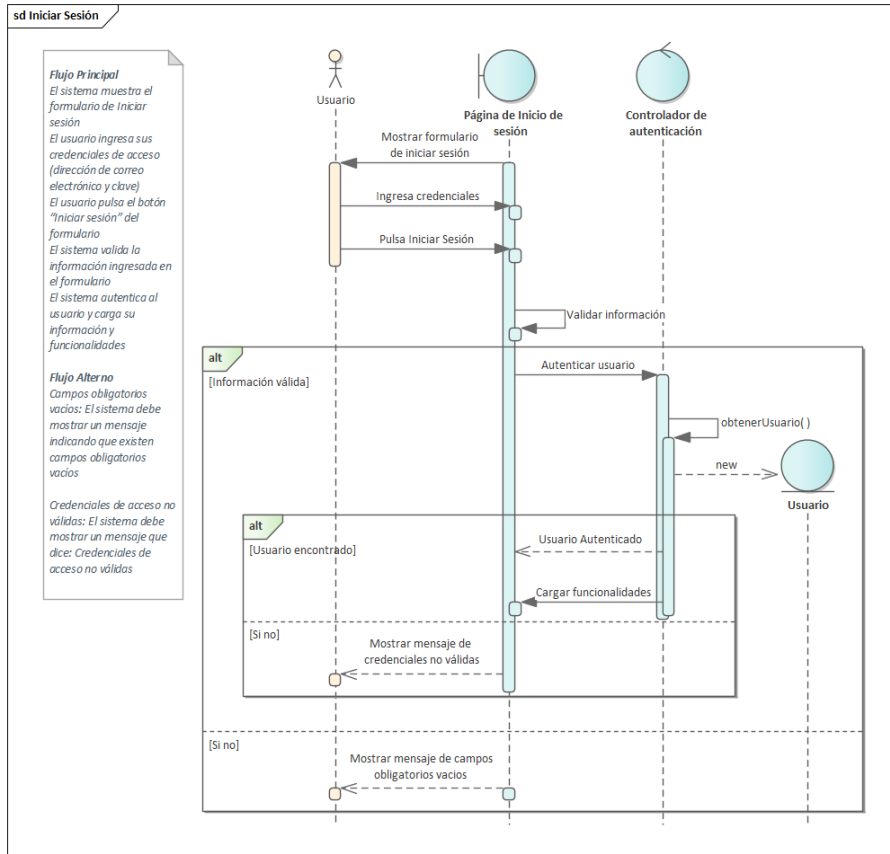


Figura 82. Diagrama de secuencia de Iniciar sesión

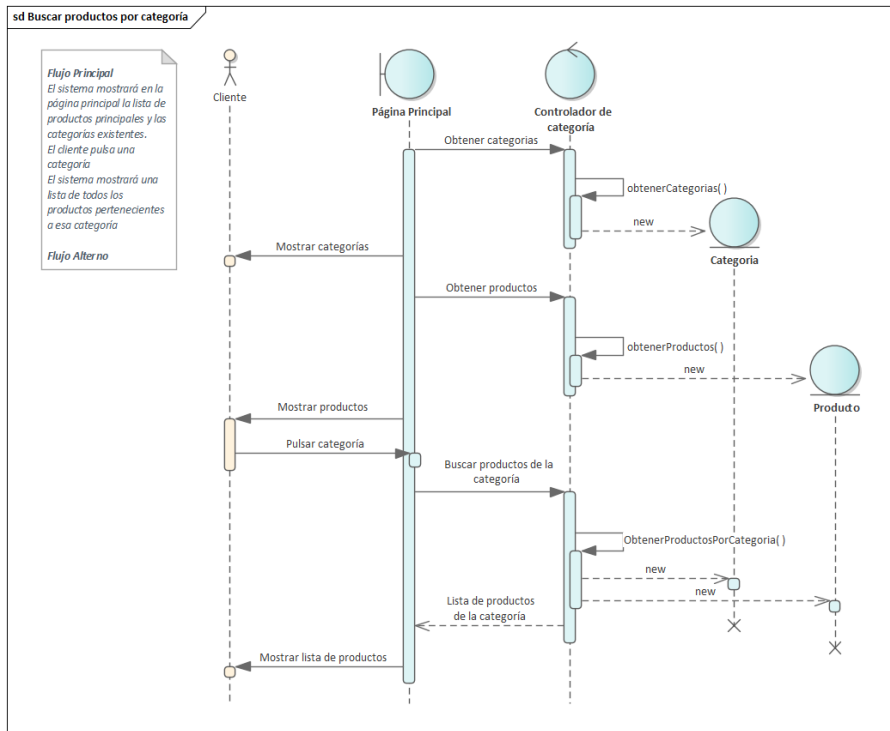


Figura 83. Diagrama de secuencia de Buscar productos por categoría

4.2. Administrar cuentas de usuario

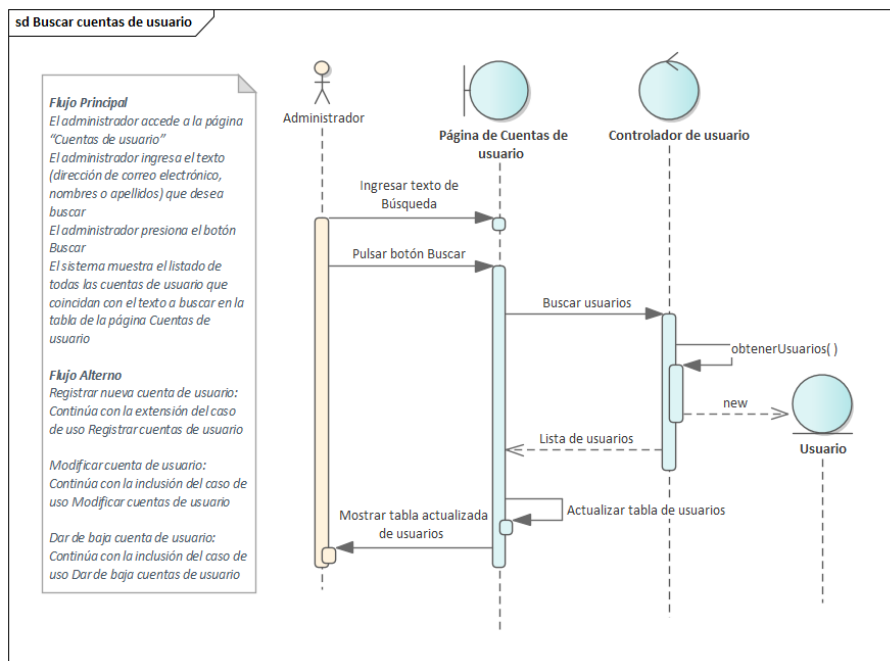


Figura 84. Diagrama de secuencia de Buscar cuentas de usuario

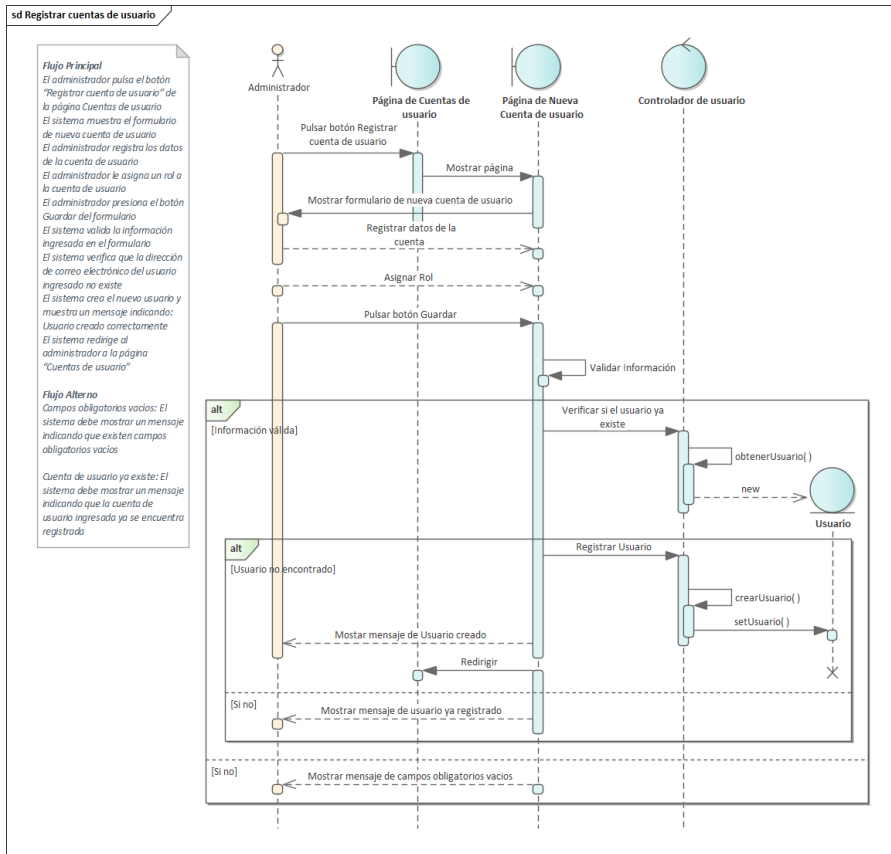


Figura 85. Diagrama de secuencia de Registrar cuentas de usuario

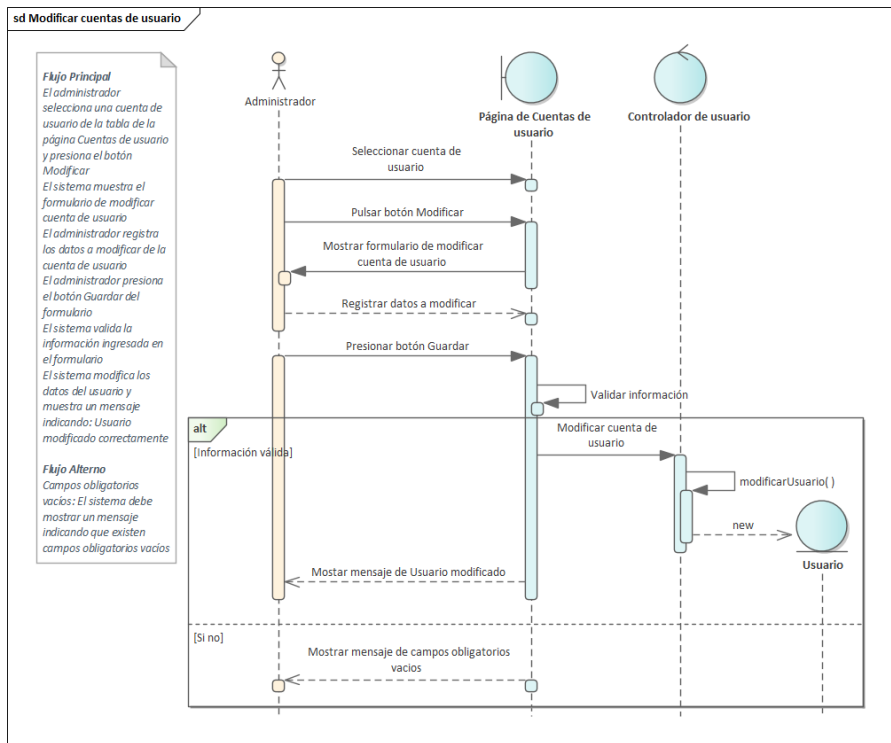


Figura 86. Diagrama de secuencia de Modificar cuentas de usuario

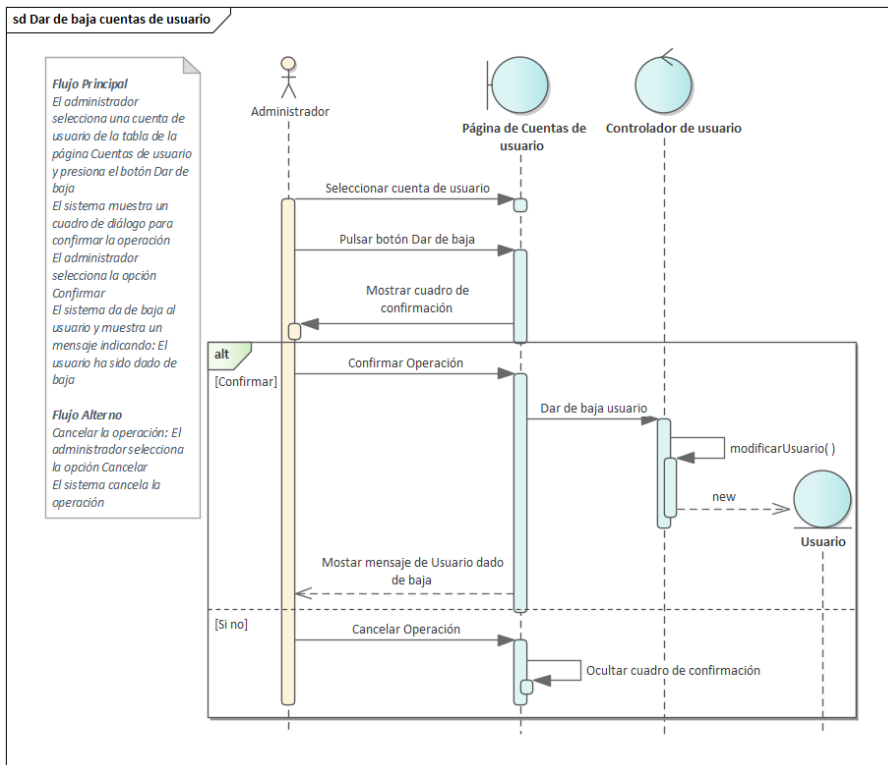


Figura 87. Diagrama de secuencia de Dar de baja cuentas de usuario

4.3. Administrar órdenes de venta

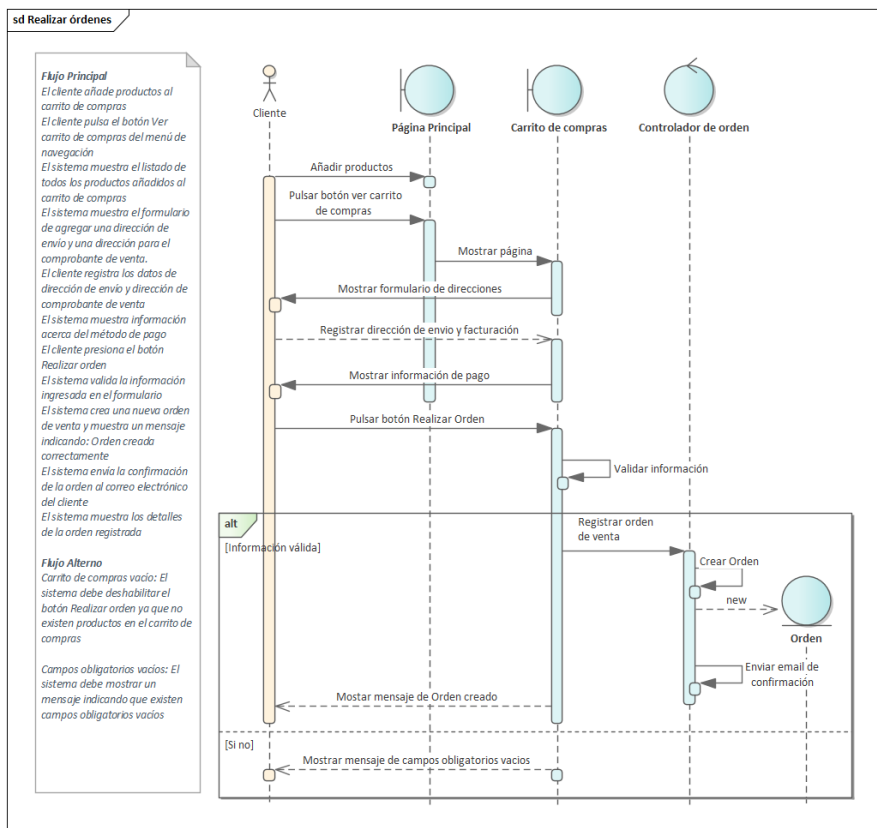


Figura 88. Diagrama de secuencia de Realizar órdenes

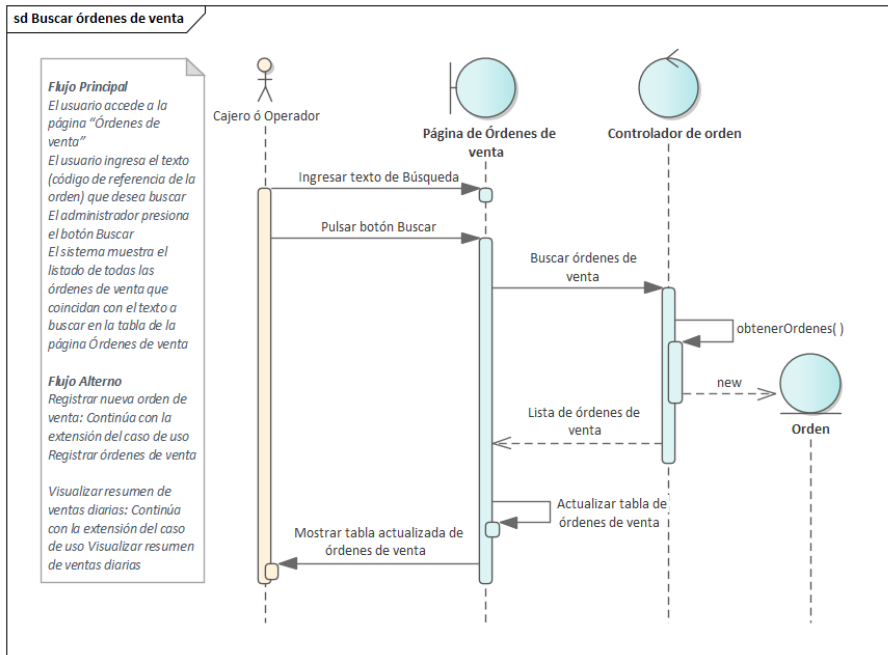


Figura 89. Diagrama de secuencia de Buscar órdenes de venta

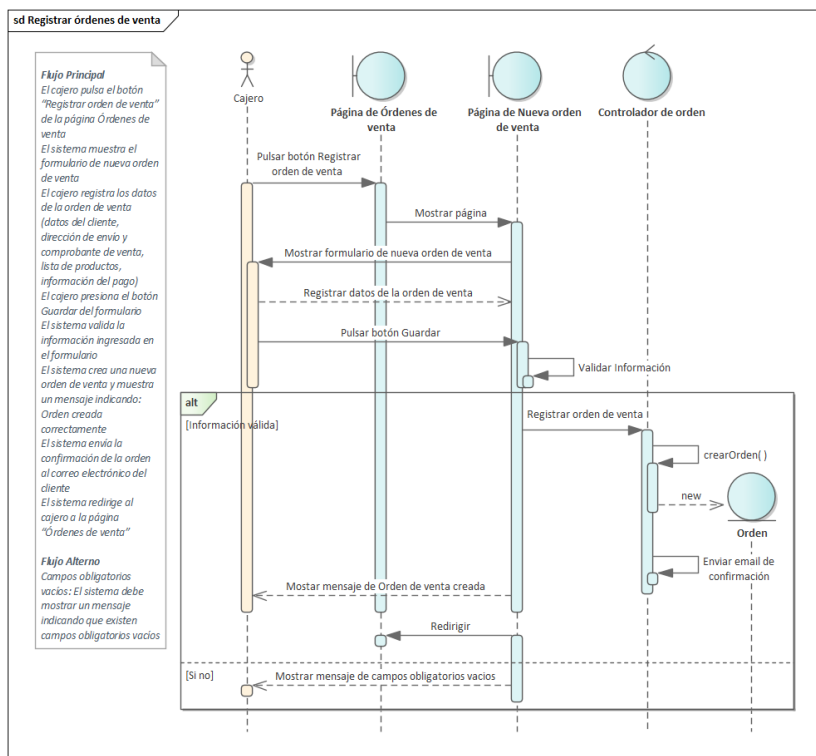


Figura 90. Diagrama de secuencia de Registrar órdenes de venta

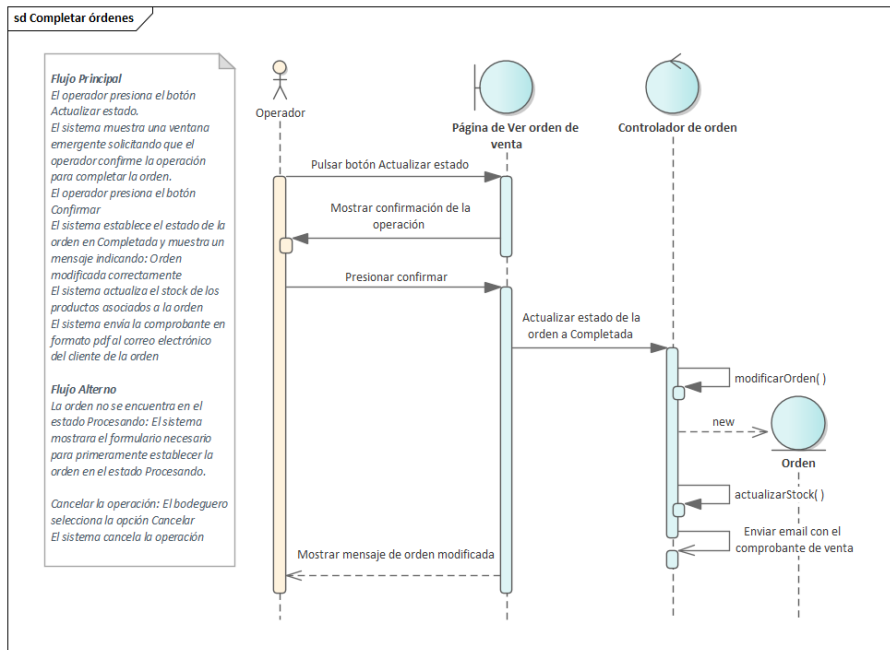


Figura 91. Diagrama de secuencia de Completar órdenes

4.4. Administrar pagos de órdenes

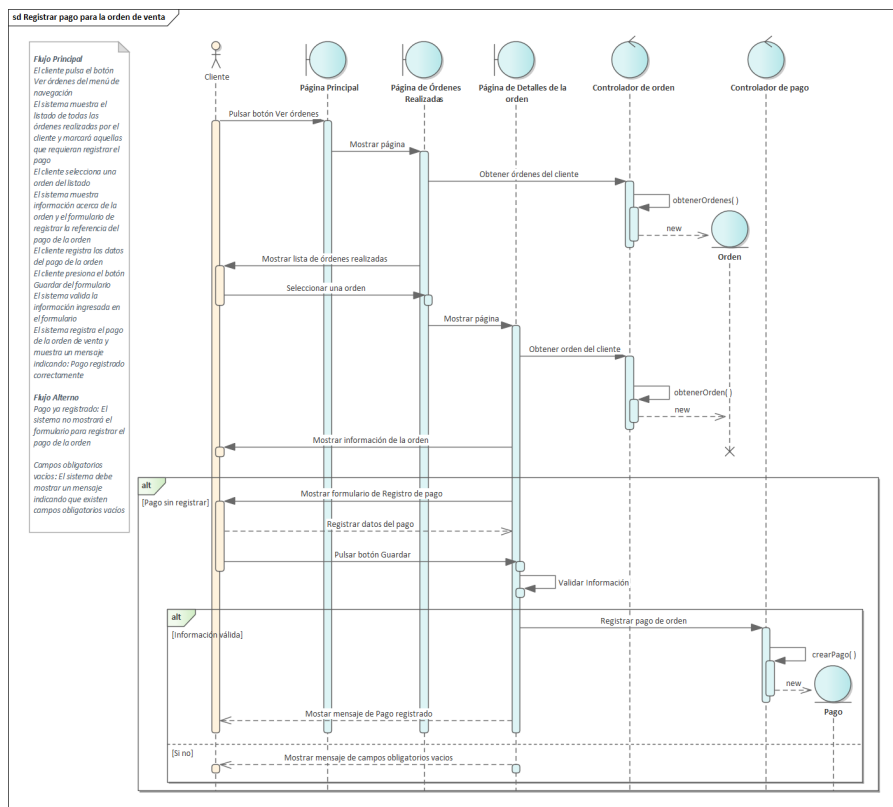


Figura 92. Diagrama de secuencia de Registrar pago para la orden de venta

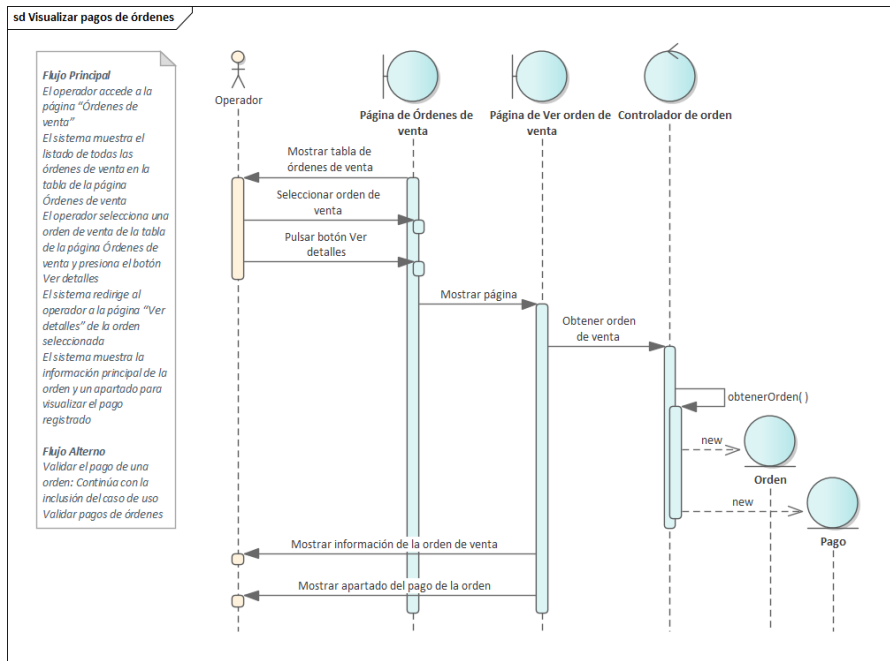


Figura 93. Diagrama de secuencia de Visualizar pagos de órdenes

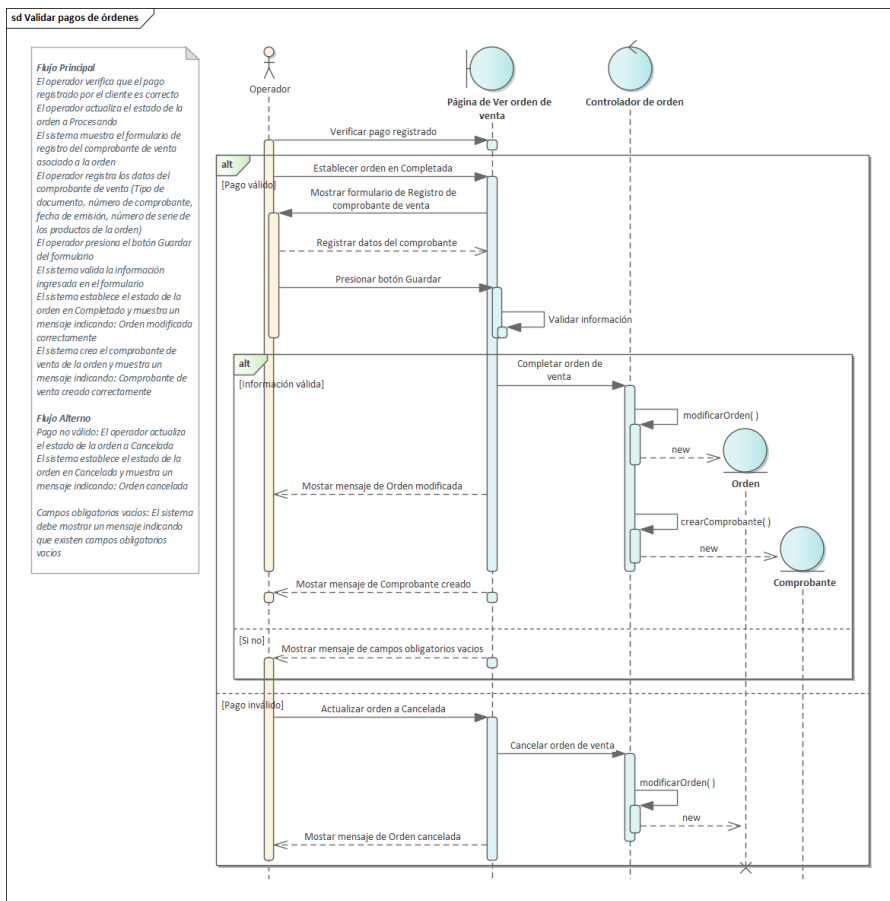


Figura 94. Diagrama de secuencia de Validar pagos de órdenes

4.5. Administrar productos

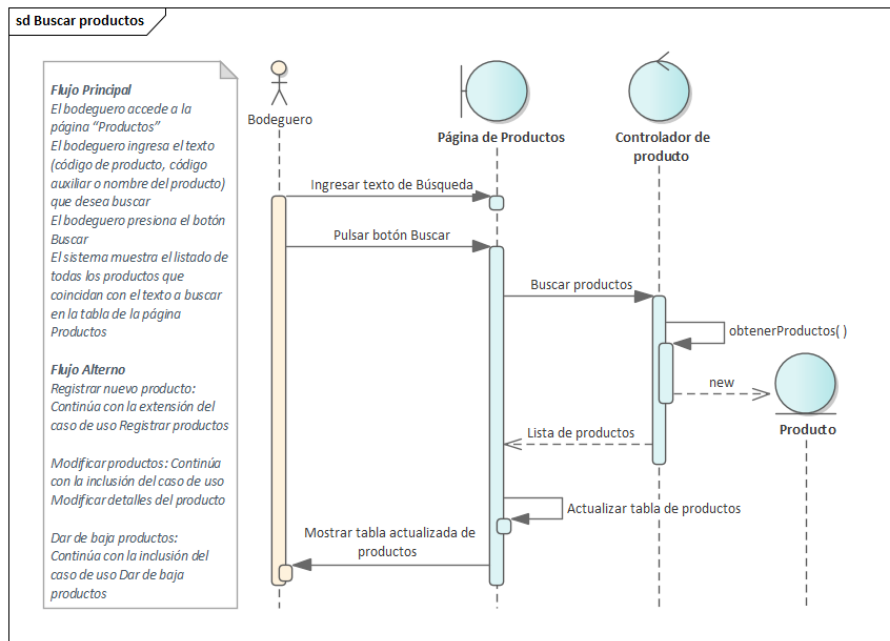


Figura 95. Diagrama de secuencia de Buscar productos

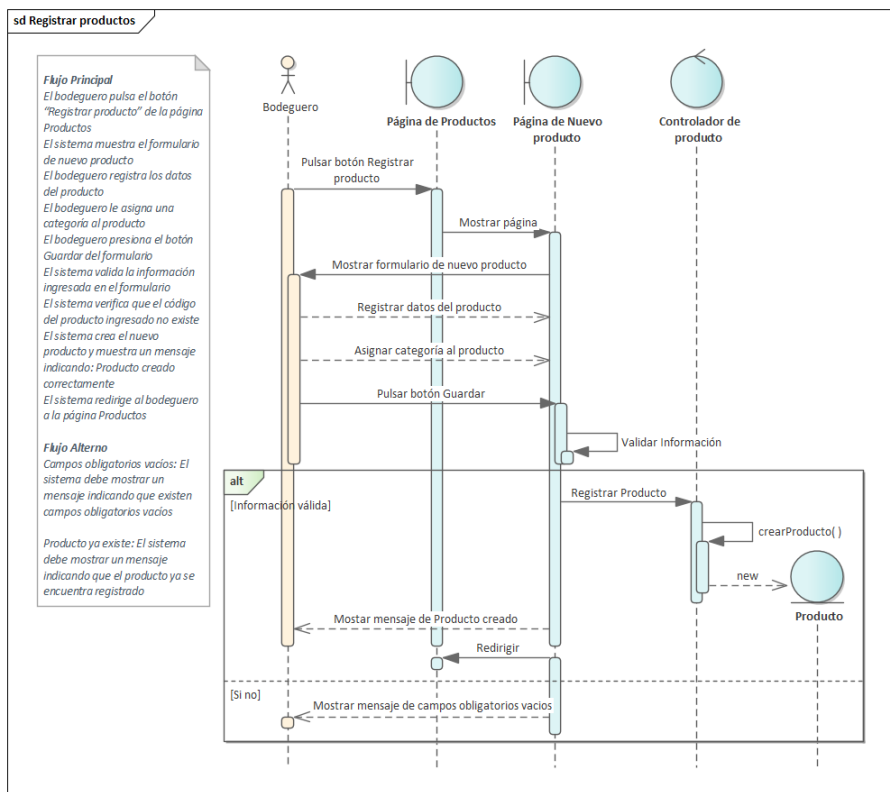


Figura 96. Diagrama de secuencia de Registrar productos

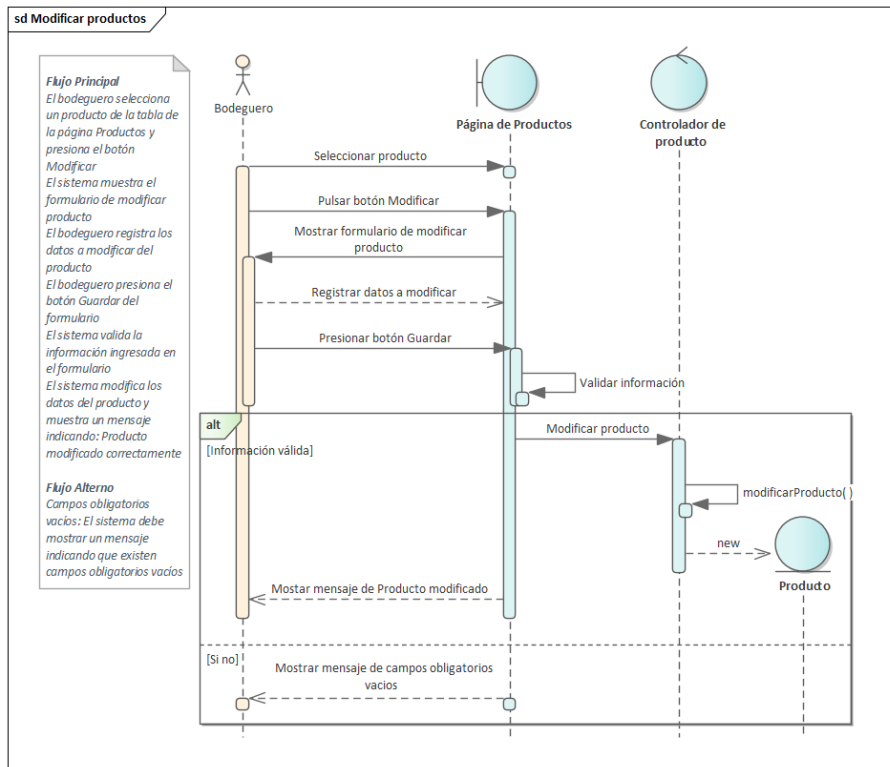


Figura 97. Diagrama de secuencia de Modificar productos

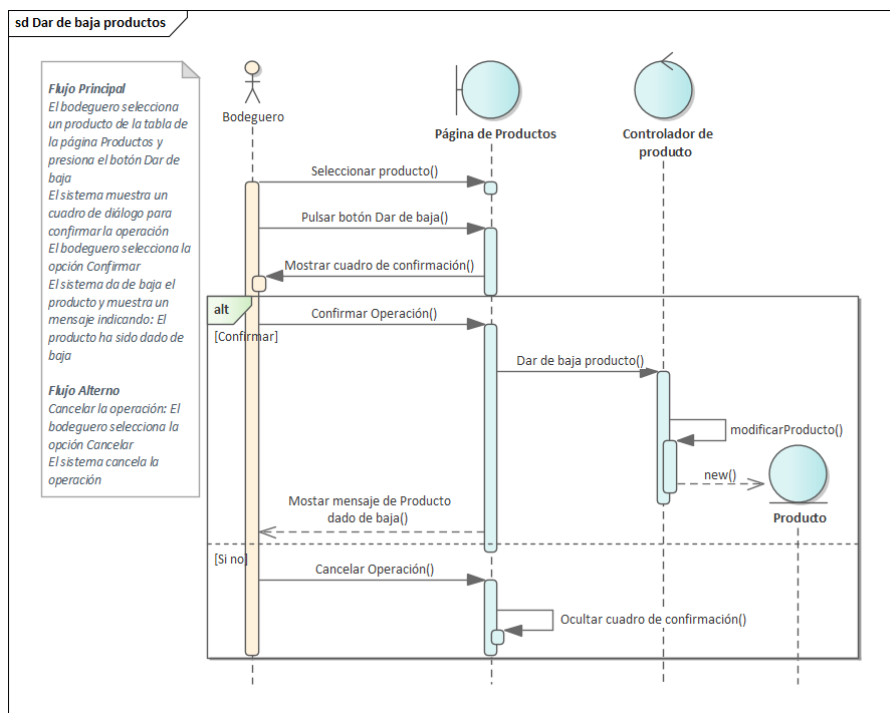


Figura 98. Diagrama de secuencia de Dar de baja productos

4.6. Administrar órdenes de compra

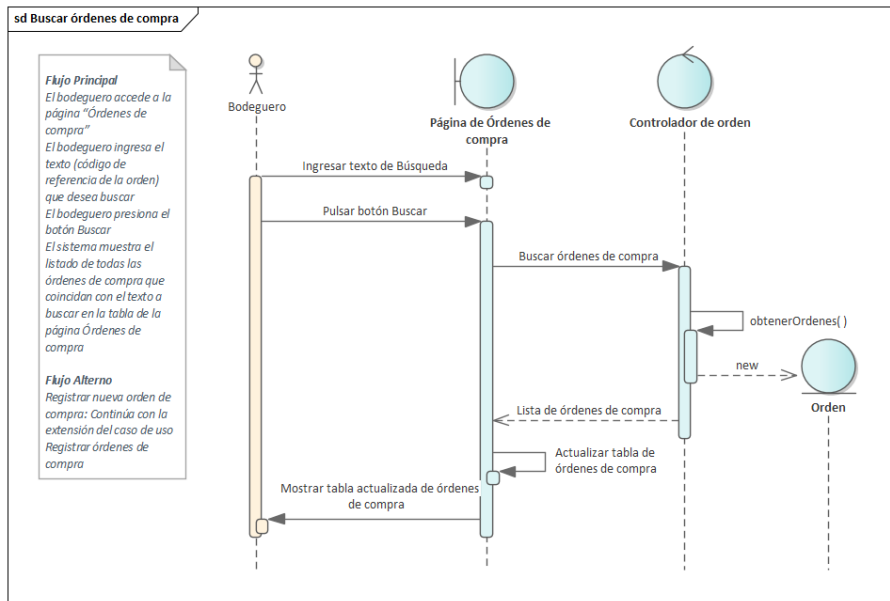


Figura 99. Diagrama de secuencia de Buscar órdenes de compra

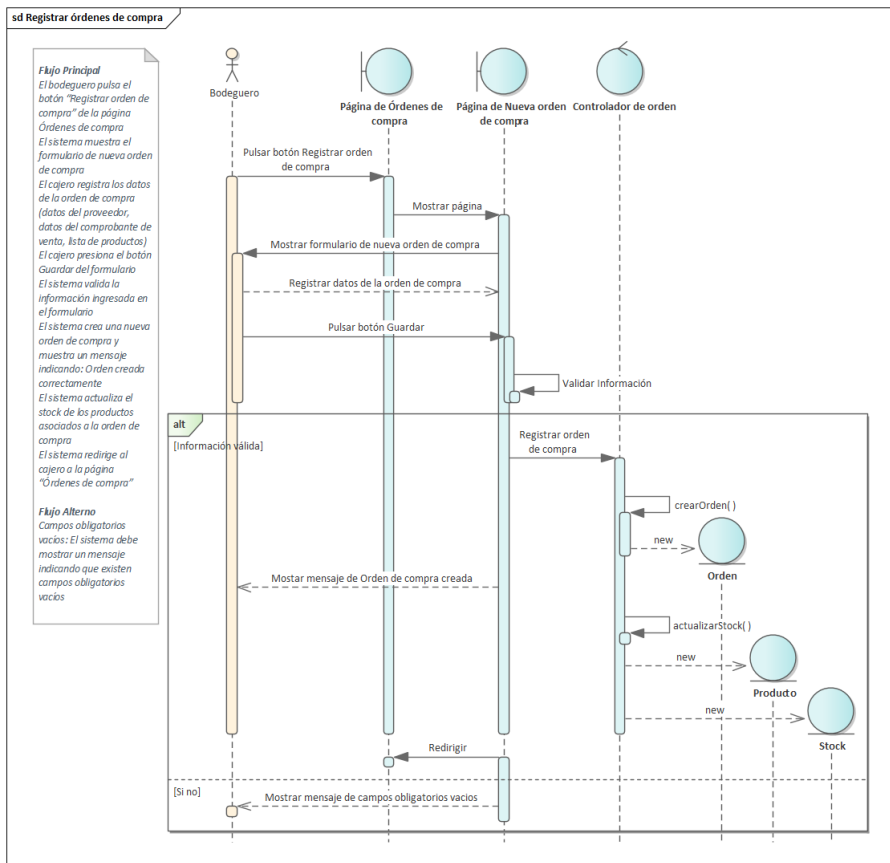


Figura 100. Diagrama de secuencia de Registrar órdenes de compra

5. Diagrama de paquetes

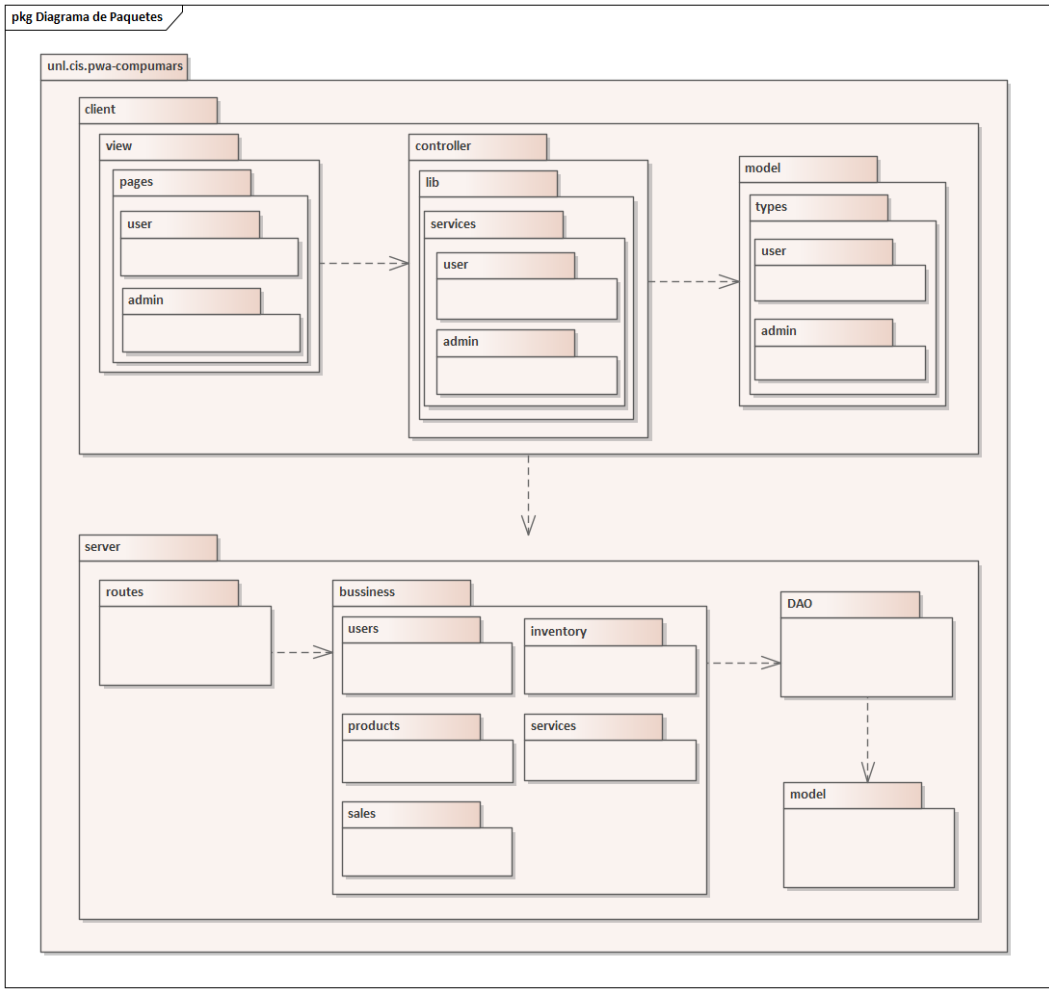


Figura 101. Diagrama de paquetes

6. Diagrama de clases

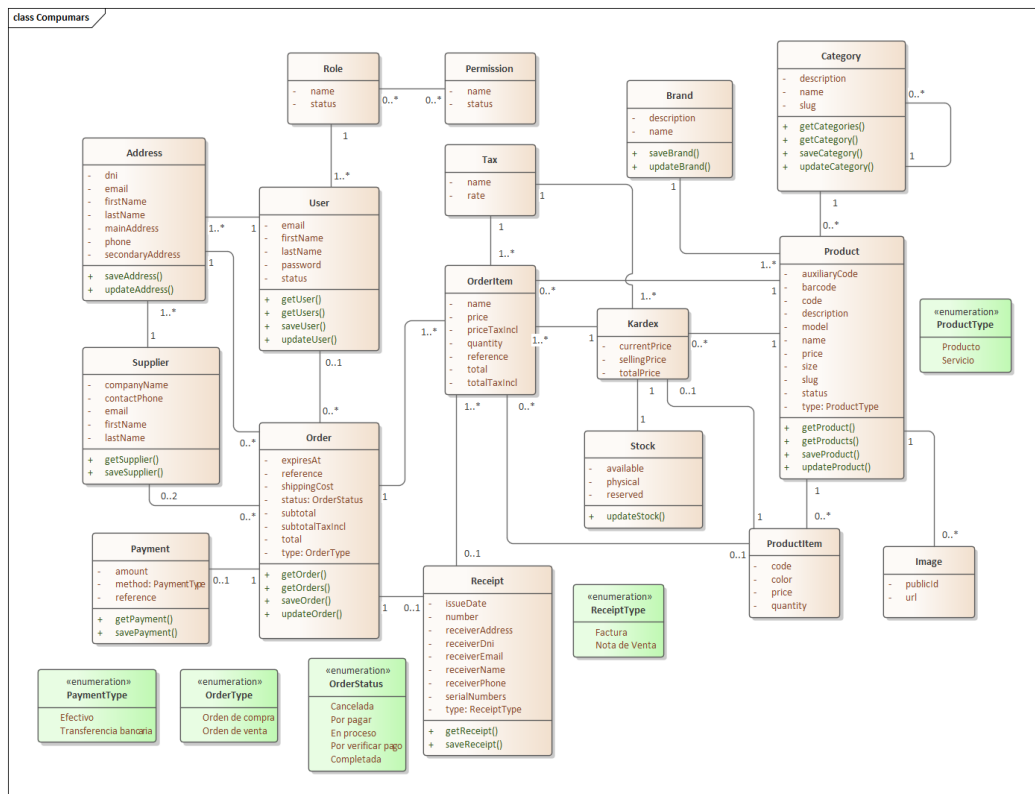


Figura 102. Diagrama de clases

7. Arquitectura Física

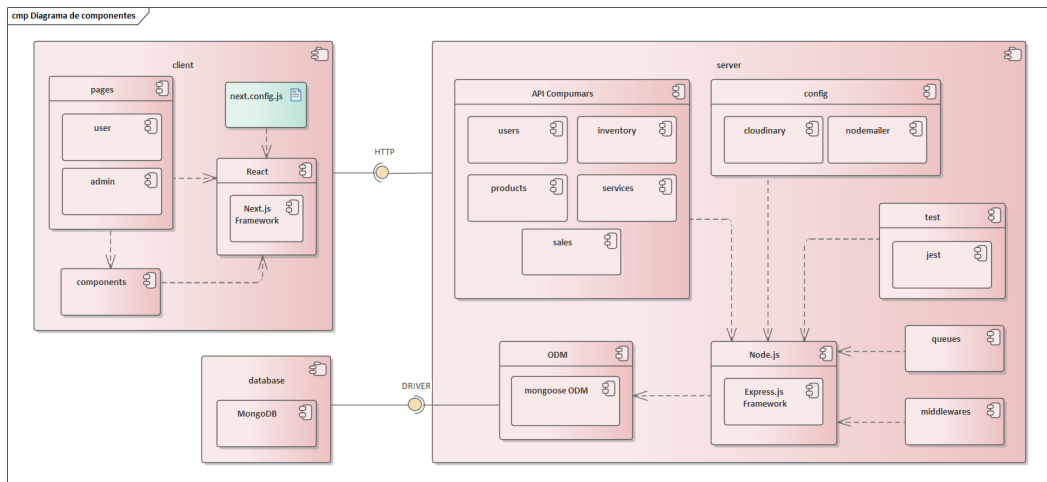


Figura 103. Diagrama de componentes

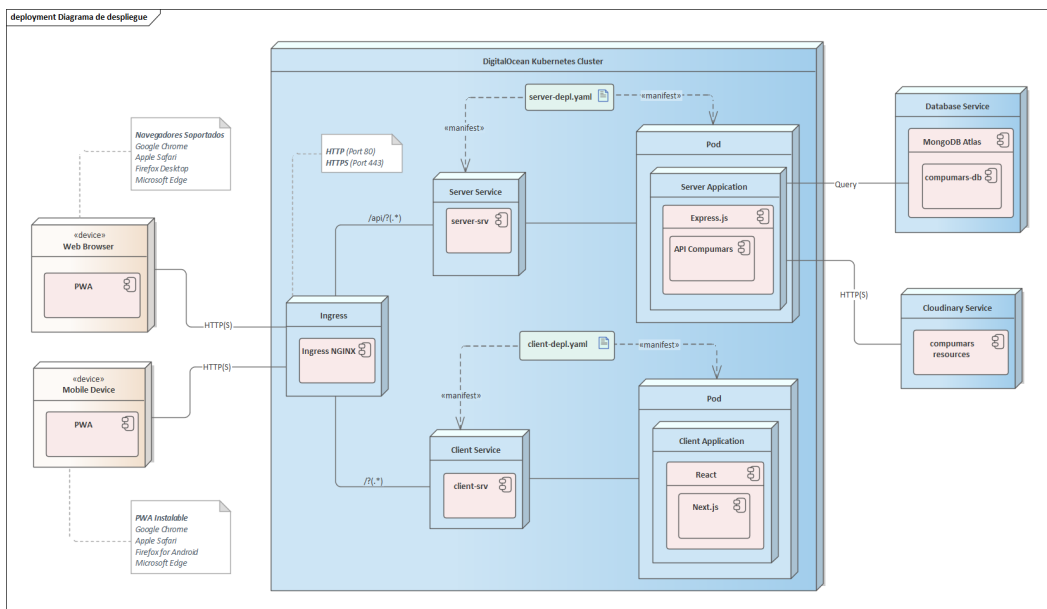


Figura 104. Diagrama de despliegue

Anexo 4. Interfaces gráficas de la implementación de la solución informática

A continuación, se muestran las interfaces gráficas que resultaron de la implementación de la Aplicación Web Progresiva. Se ha dividido en secciones para tener un mejor panorama de a qué módulo pertenece cada ilustración.

1. General

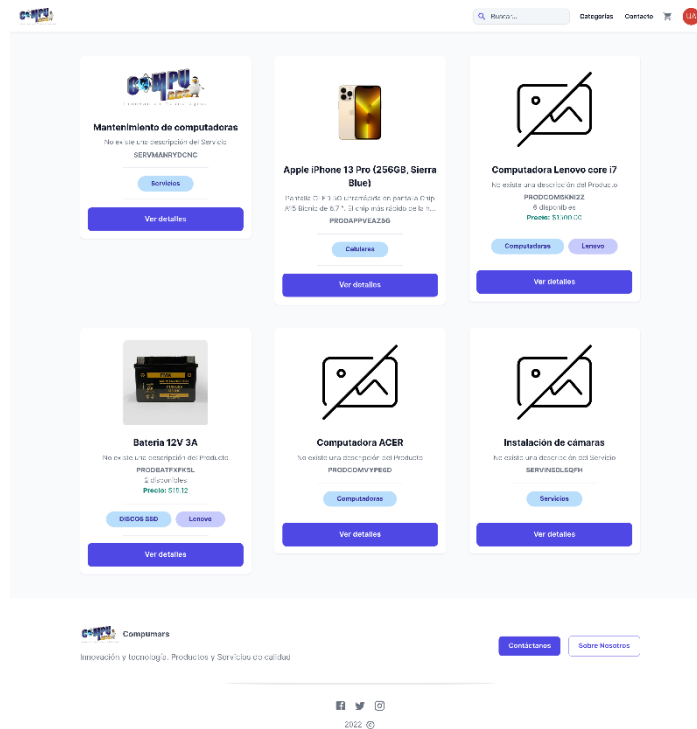


Figura 105. Interfaz gráfica de la Página principal

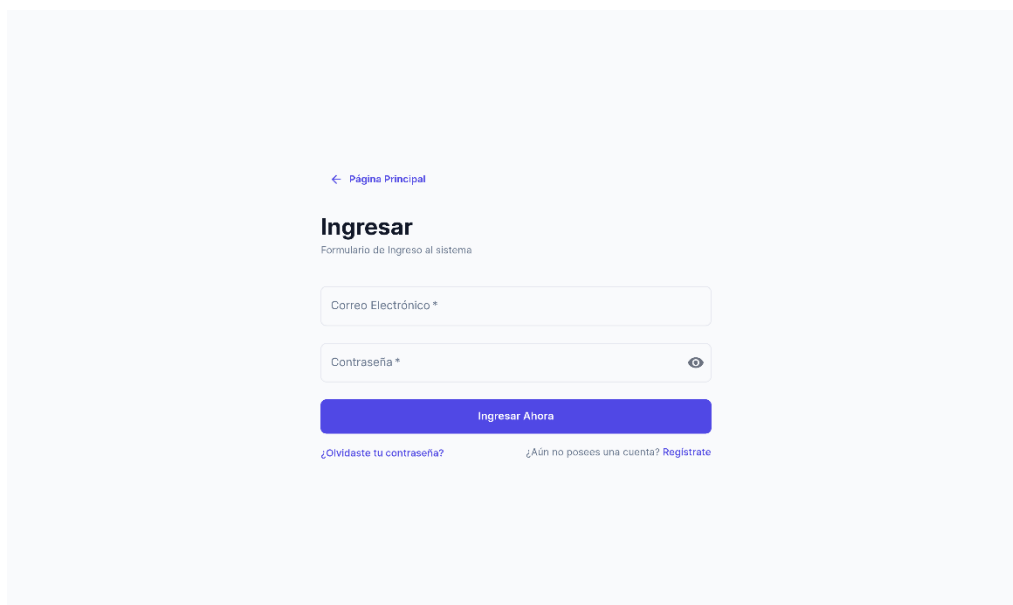


Figura 106. Interfaz gráfica de Ingreso al sistema

← [Página Principal](#)


Registro

Registro de datos de un nuevo usuario

Nombres *

Apellidos *

Correo Electrónico *

Contraseña * 

Guarde su contraseña en un lugar seguro

Al registrarse usted acepta [los Términos y Condiciones](#) propuestos por Compumars

[Registrarme](#)

[¿Ya tienes una cuenta? Ingresa ahora](#)

Figura 107. Interfaz gráfica de Registro en el sistema

← [Iniciar sesión](#)

Recuperar Contraseña

¿Olvidaste tu contraseña? No hay problema, ingresa tu correo electrónico con el cual te registraste a Compumars

Correo Electrónico *

Una vez completado el formulario revise la bandeja de entrada de su correo

[Recuperar Contraseña](#)

Figura 108. Interfaz gráfica de Recuperar contraseña

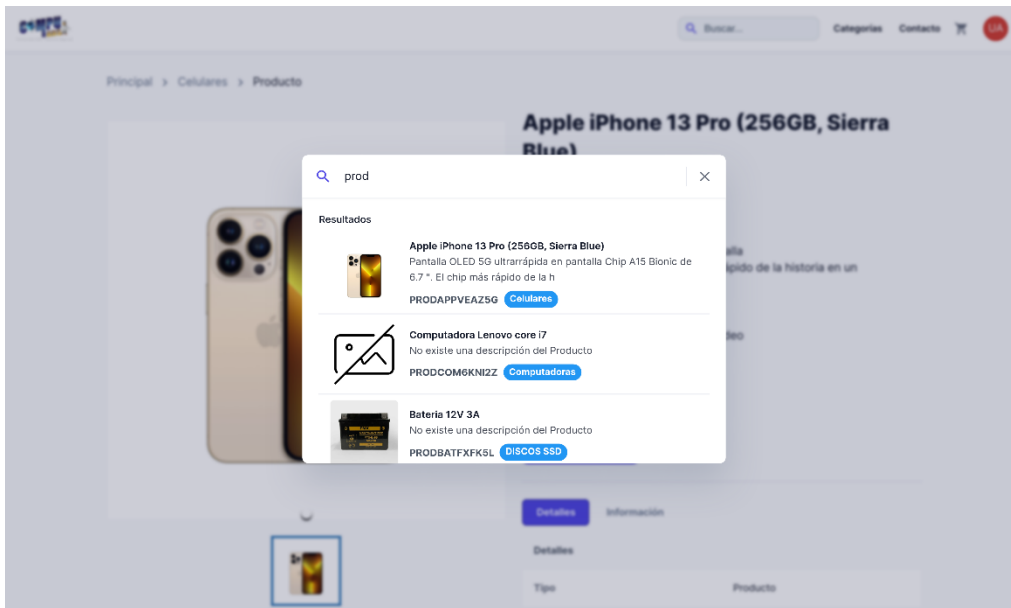


Figura 109. Interfaz gráfica de Buscar Productos

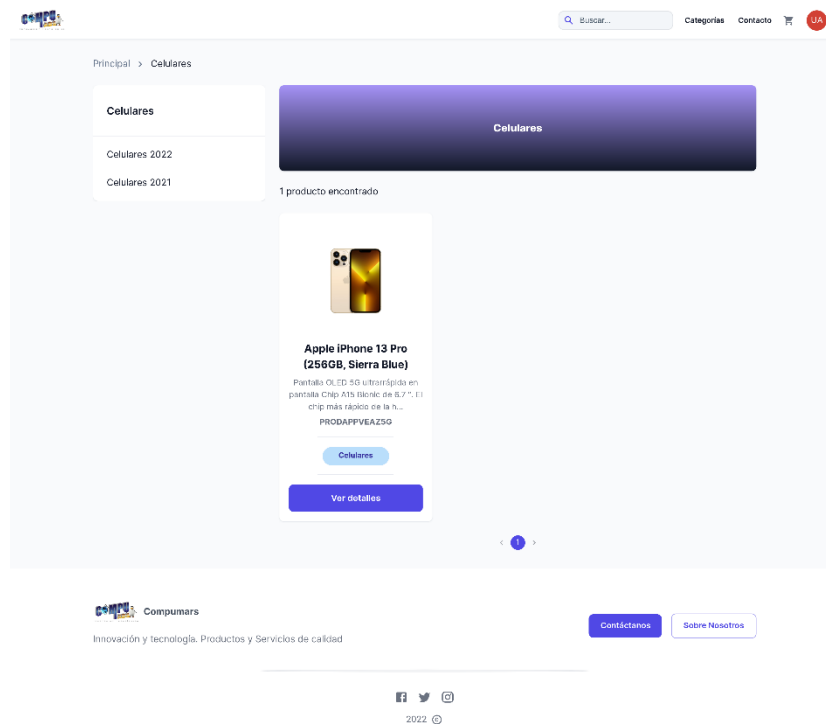


Figura 110. Interfaz gráfica de Buscar productos por categorías

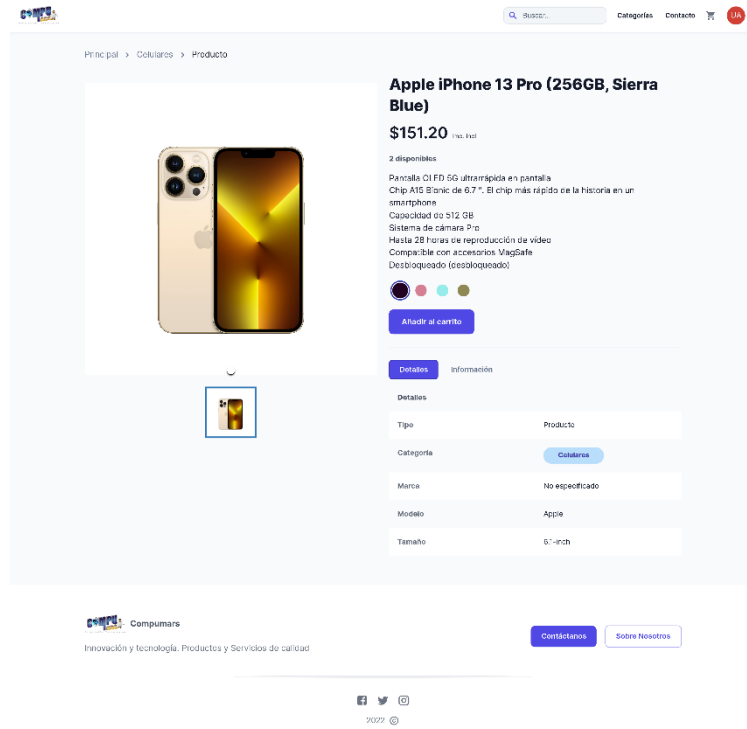


Figura 111. Interfaz gráfica de Ver producto

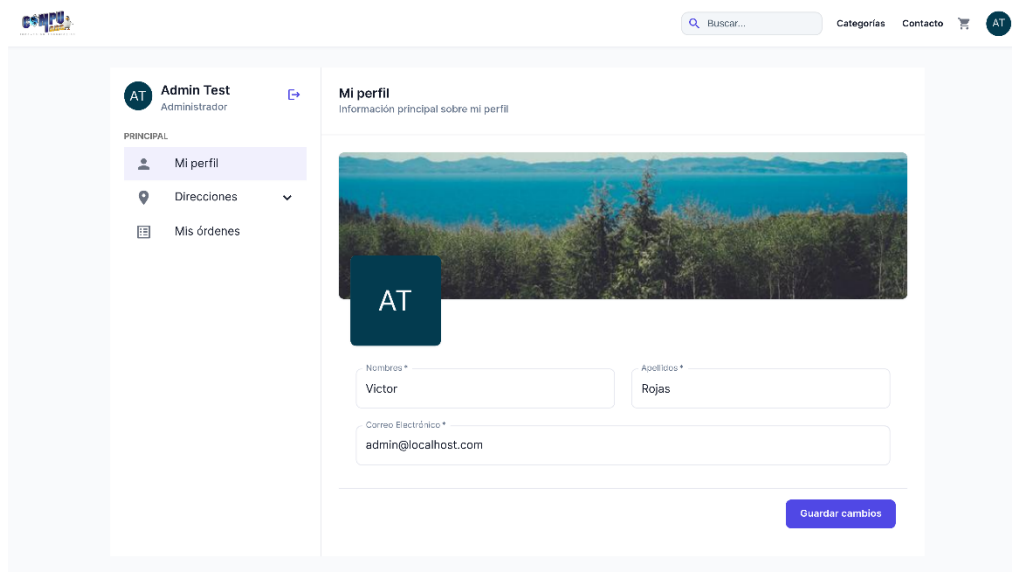


Figura 112. Interfaz gráfica de Perfil de usuario

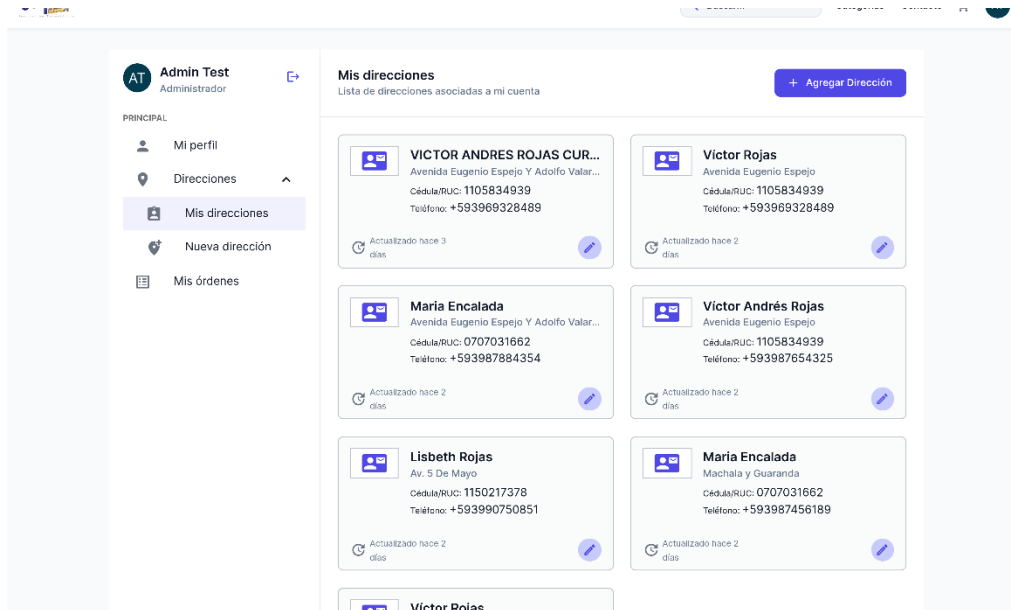


Figura 113. Interfaz gráfica de Listar direcciones del cliente

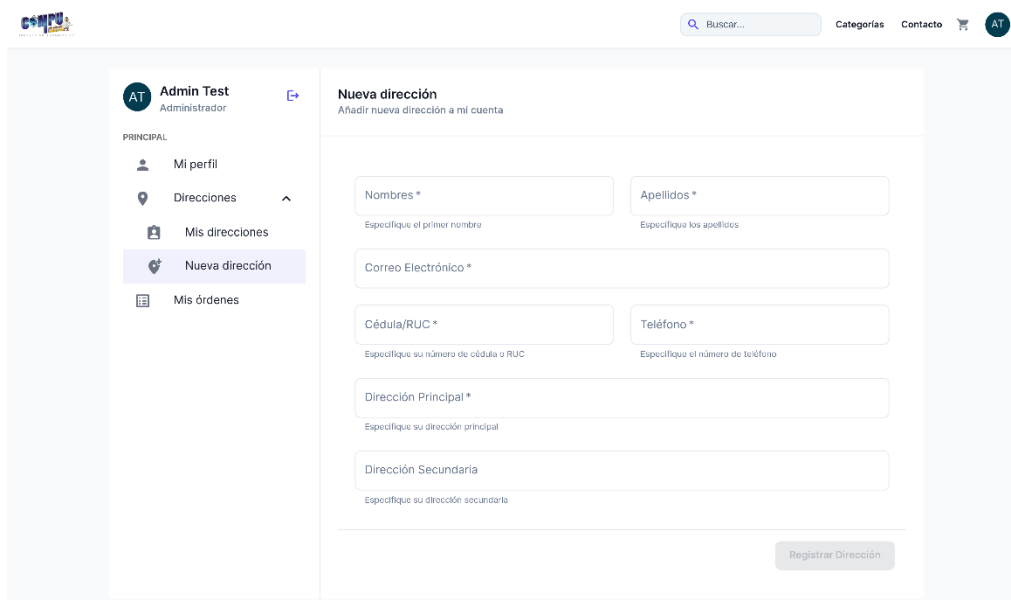


Figura 114. Interfaz gráfica de Nueva dirección

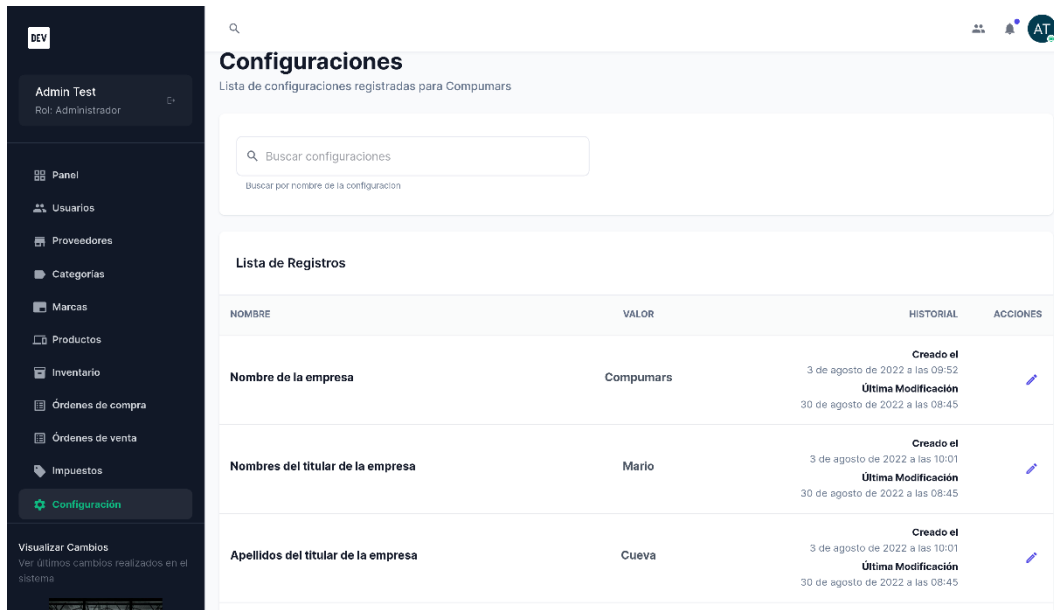


Figura 115. Interfaz gráfica de Ver Configuraciones del sistema

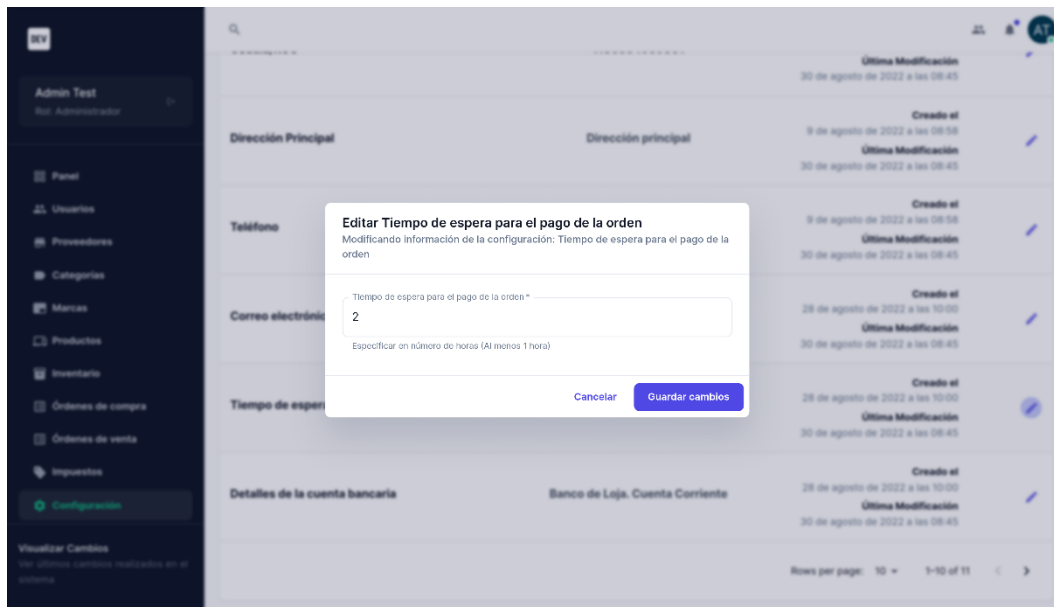
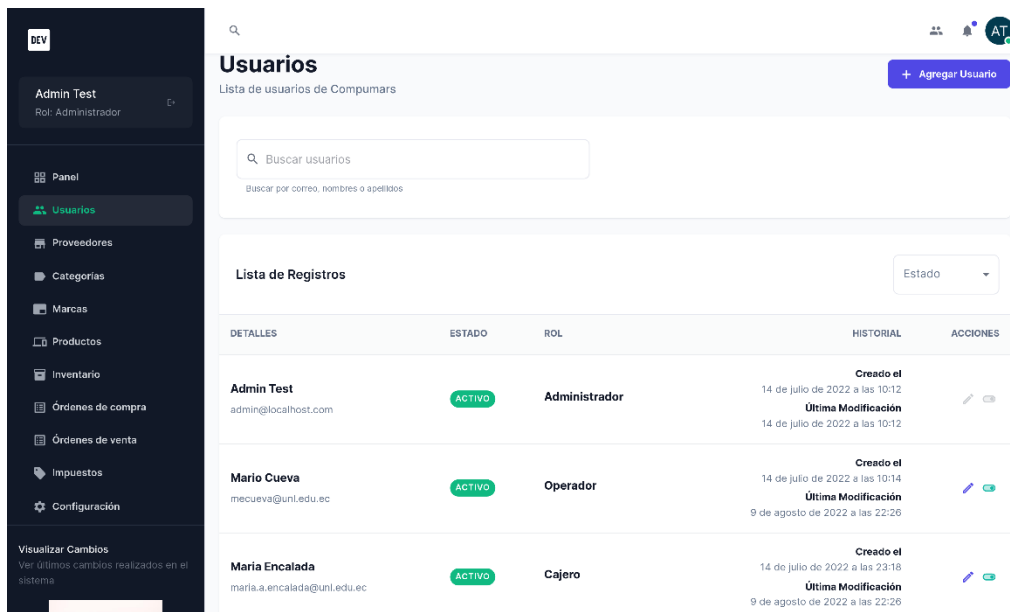


Figura 116. Interfaz gráfica de Modificar parámetro de configuración

2. Módulo de gestión de usuarios



The screenshot displays the 'Usuarios' (Users) management interface. On the left is a dark sidebar with navigation options: Admin Test (Rol: Administrador), Panel, Usuarios (highlighted), Proveedores, Categorías, Marcas, Productos, Inventario, Órdenes de compra, Órdenes de venta, Impuestos, and Configuración. At the bottom of the sidebar is a 'Visualizar Cambios' section. The main content area is titled 'Usuarios' and shows a 'Lista de usuarios de Compumars' with a '+ Agregar Usuario' button. A search bar is present with the placeholder 'Buscar usuarios' and a note 'Buscar por correo, nombres o apellidos'. Below the search bar is a 'Lista de Registros' section with a 'Estado' dropdown menu. A table lists three users:







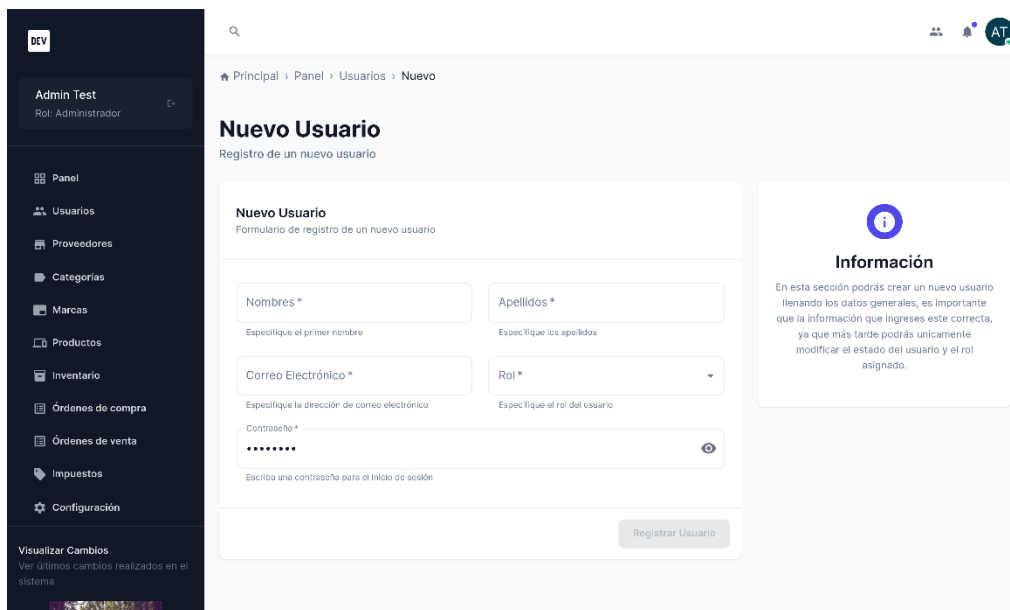
DETALLES	ESTADO	ROL	HISTORIAL	ACCIONES
Admin Test admin@localhost.com	ACTIVO	Administrador	Creado el 14 de julio de 2022 a las 10:12 Última Modificación 14 de julio de 2022 a las 10:12	 
Mario Cueva mecuева@unl.edu.ec	ACTIVO	Operador	Creado el 14 de julio de 2022 a las 10:14 Última Modificación 9 de agosto de 2022 a las 22:26	 
Maria Encalada maria.a.encialada@unl.edu.ec	ACTIVO	Cajero	Creado el 14 de julio de 2022 a las 23:18 Última Modificación 9 de agosto de 2022 a las 22:26	 

Figura 117. Interfaz gráfica de Listar usuarios



The screenshot shows the 'Nuevo Usuario' (New User) registration form. The sidebar is identical to the previous screenshot. The main content area is titled 'Nuevo Usuario' and includes a breadcrumb trail: 'Principal > Panel > Usuarios > Nuevo'. Below the title is the subtitle 'Registro de un nuevo usuario'. The form contains the following fields:

- Nombres ***: Input field with placeholder 'Especifique el primer nombre'.
- Apellidos ***: Input field with placeholder 'Especifique los apellidos'.
- Correo Electrónico ***: Input field with placeholder 'Especifique la dirección de correo electrónico'.
- Rol ***: Dropdown menu with placeholder 'Especifique el rol del usuario'.
- Contraseña ***: Password input field with placeholder 'Escriba una contraseña para el inicio de sesión' and a toggle for visibility.

At the bottom right of the form is a 'Registrar Usuario' button. To the right of the form is an 'Información' box with a blue information icon and the following text: 'En esta sección podrás crear un nuevo usuario llenando los datos generales, es importante que la información que ingreses este correcta, ya que más tarde podrás únicamente modificar el estado del usuario y el rol asignado.'

Figura 118. Interfaz gráfica de Nuevo usuario

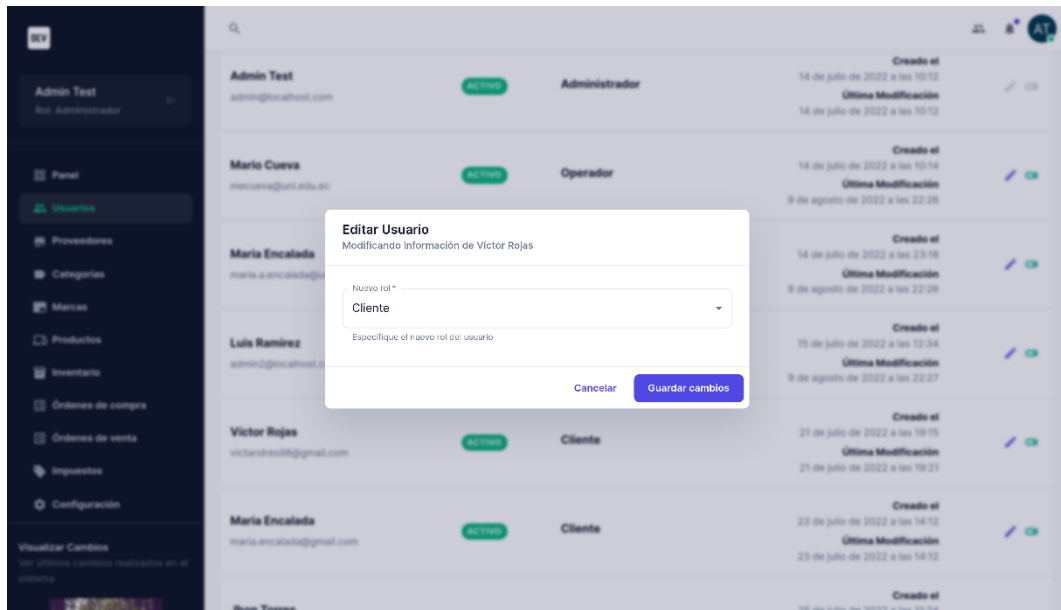


Figura 119. Interfaz gráfica de Modificar usuario

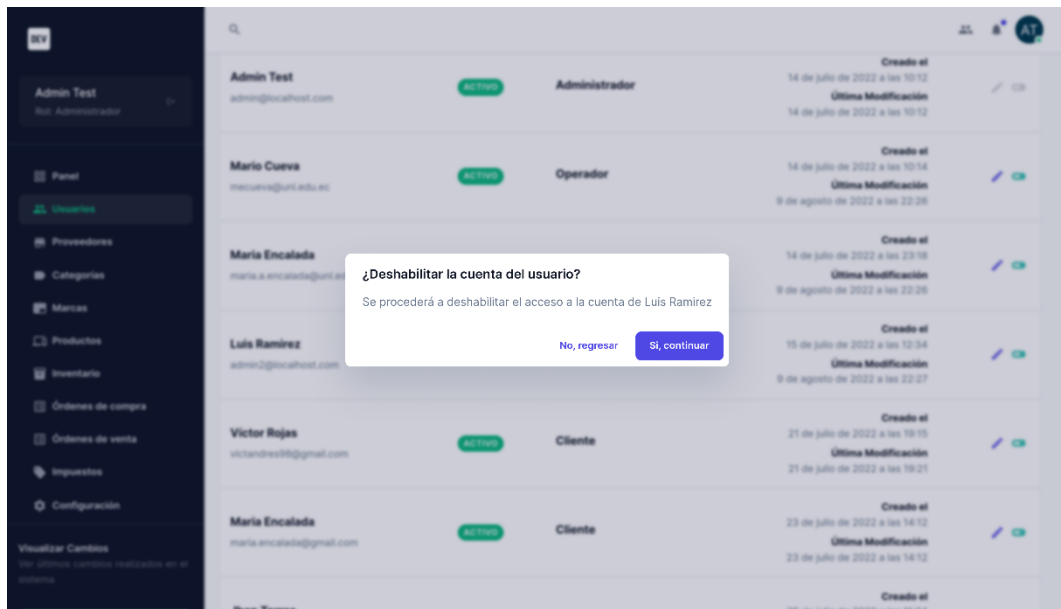


Figura 120. Interfaz gráfica de Dar de baja usuario

Proveedores
Lista de proveedores de Compumars

+ Agregar Proveedor

Buscar proveedores
Buscar por razón social, correo, nombres o apellidos

Lista de Registros

PROVEEDOR	DIRECCIÓN	INFORMACIÓN	HISTORIAL	ACCIONES
VICTOR ANDRES ROJAS CURIMILMA Universidad Nacional de Loja	Avenida Eugenio Espejo Y Adolfo Valarezo, 200-66 Calle secundaria victandres98@gmail.com Correo	1105834939 Cedula/RUC +593969328489 Teléfono	Creado el 14 de julio de 2022 a las 23:28 Última Modificación 9 de agosto de 2022 a las 17:43	
Andrés Peña Andrés Peña	Avenida Eugenio Espejo Y Adolfo Valarezo, 200-66 Calle secundaria user@email.com Correo	222222222 Cedula/RUC +593969328489 Teléfono	Creado el 15 de julio de 2022 a las 12:37 Última Modificación 21 de julio de 2022 a las 09:59	
Lisbeth Rojas Lisbeth Rojas	Avenida Eugenio Espejo Y Adolfo Valarezo 200-66 lisbeth@gmail.com Correo	1105834939 Cedula/RUC +983969328784 Teléfono	Creado el 19 de julio de 2022 a las 08:45 Última Modificación 19 de julio de 2022 a las 08:45	

Figura 121. Interfaz gráfica de Listar proveedores

Nuevo Proveedor
Formulario de registro de un nuevo proveedor

Información
En esta sección podrás crear un nuevo proveedor registrando los datos generales y de dirección. Los proveedores son necesarios cuando se registre una nueva orden de compra.

Nombres *
Especifique el primer nombre

Apellidos *
Especifique los apellidos

Razón Social
Especifique el nombre de la empresa

Correo Electrónico *
Especifique la dirección de correo electrónico

Cédula/RUC *
Especifique el número de cédula o RUC

Teléfono *
Especifique el número de teléfono

Teléfono de contacto
Especifique el número de contacto

Dirección Principal *
Especifique la dirección principal

Dirección Secundaria
Especifique la dirección secundaria

Registrar Proveedor

Figura 122. Interfaz gráfica de Nuevo proveedor

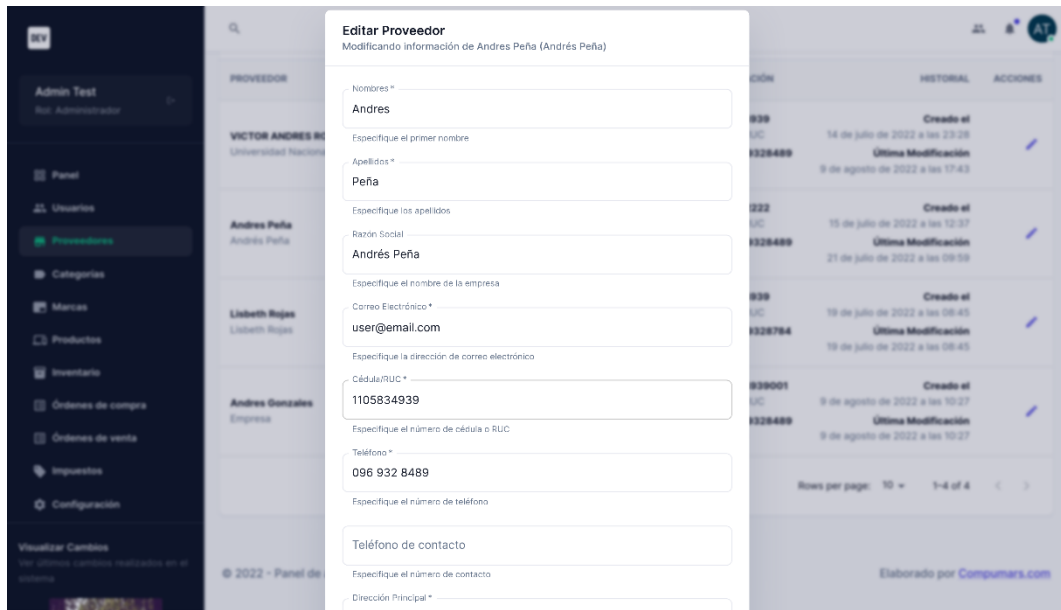


Figura 123. Interfaz gráfica de Modificar proveedor

3. Módulo de gestión de productos

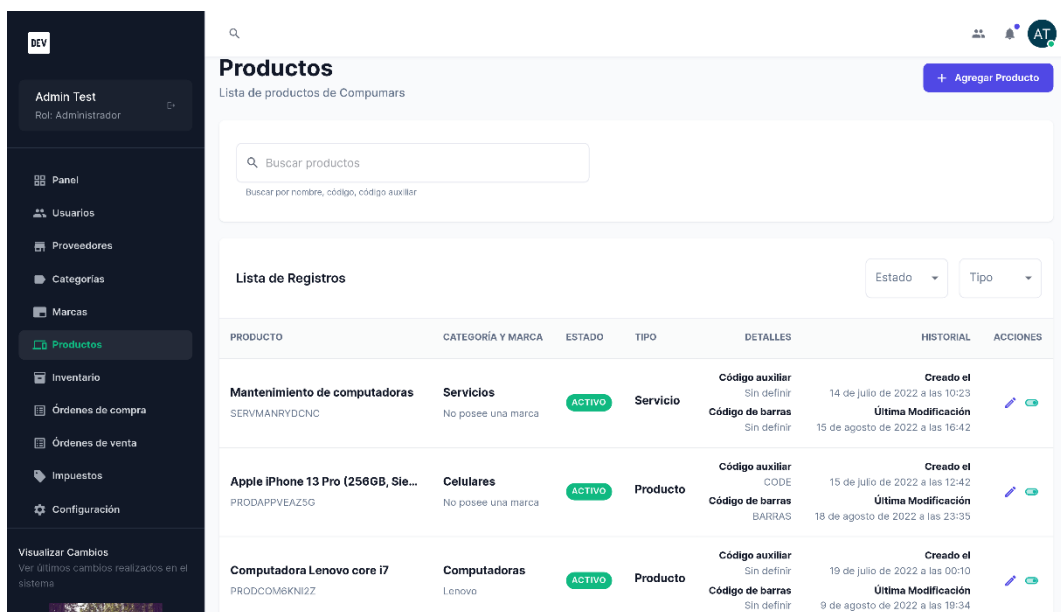


Figura 124. Interfaz gráfica de Listar productos

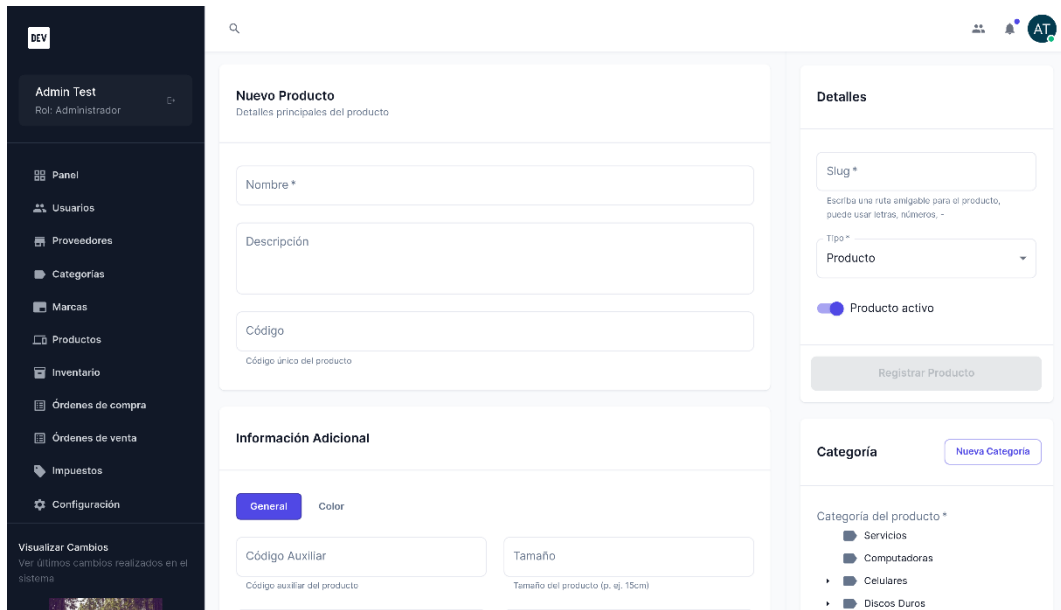


Figura 125. Interfaz gráfica de Nuevo producto

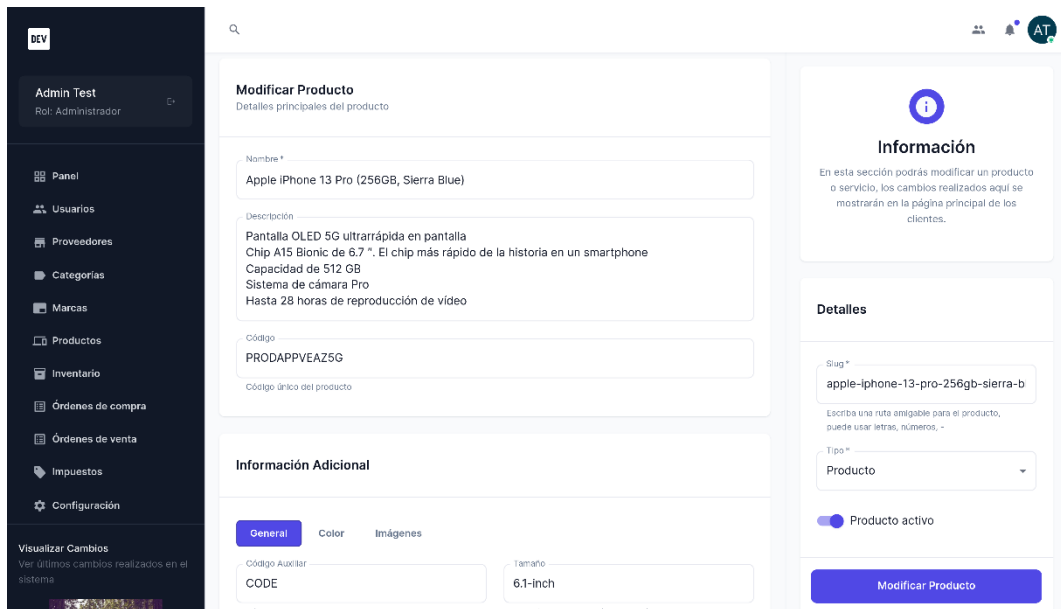


Figura 126. Interfaz gráfica de Modificar producto

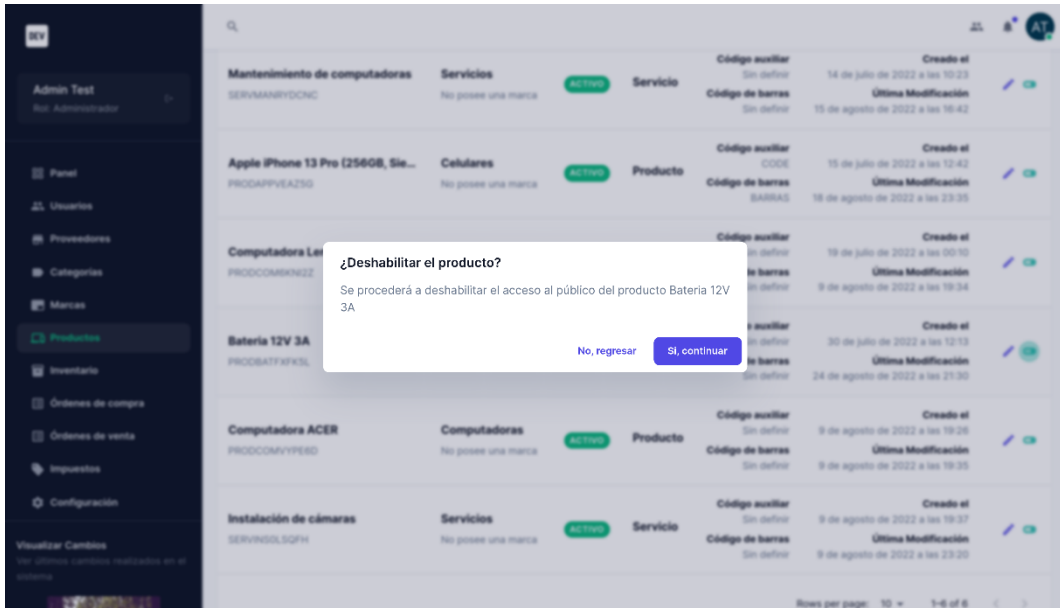


Figura 127. Interfaz gráfica de Dar de baja producto

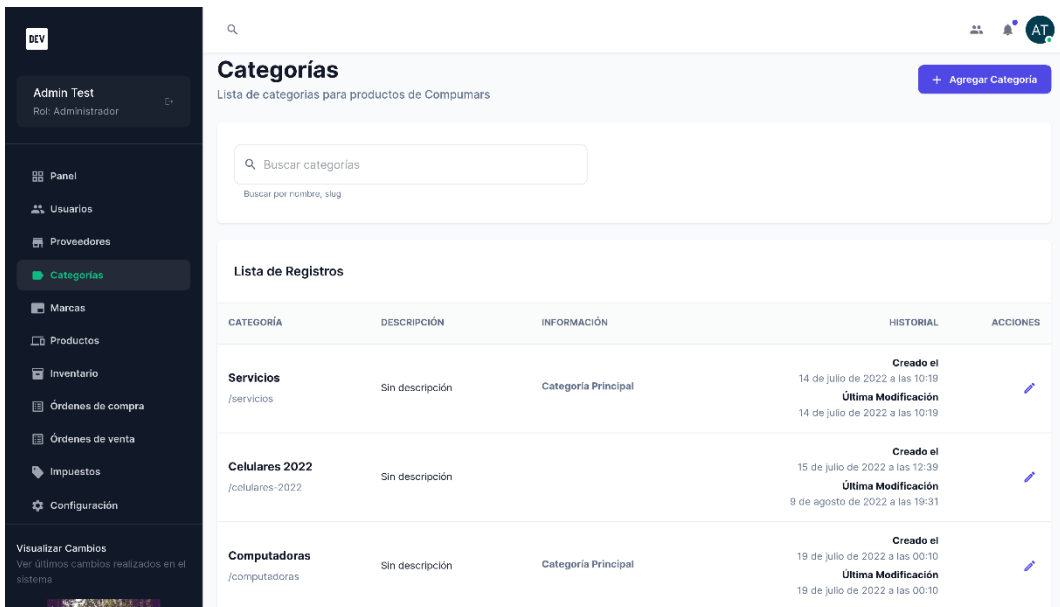


Figura 128. Interfaz gráfica de Listar categorías

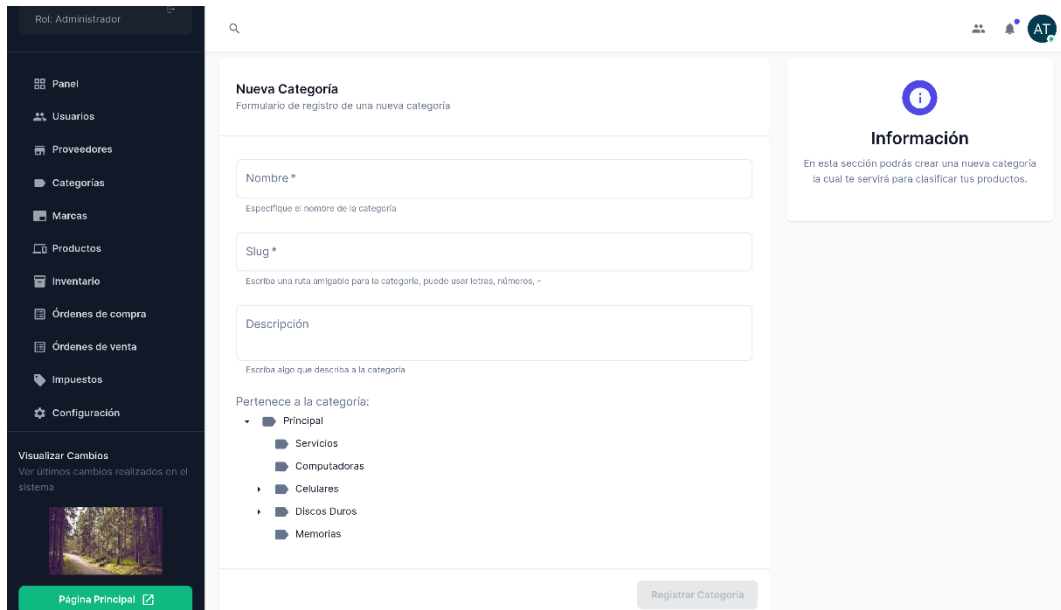


Figura 129. Interfaz gráfica de Nueva categoría

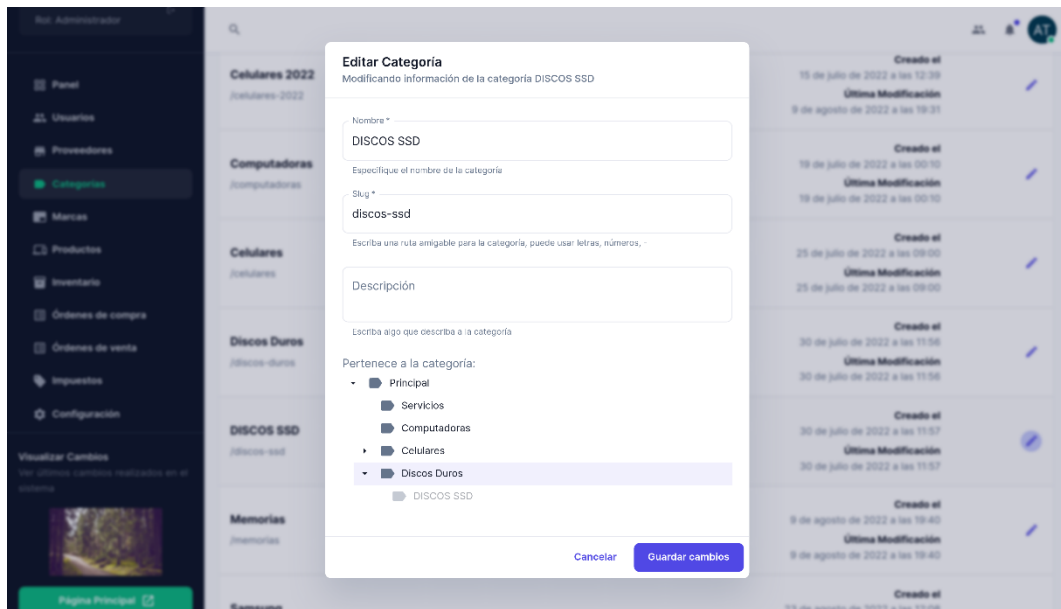


Figura 130. Interfaz gráfica de Modificar categoría

Principal > Panel > Marcas

Marcas

Lista de marcas para productos de Compumars

+ Agregar Marca

Buscar marcas

Buscar por nombre

Lista de Registros

MARCA	DESCRIPCIÓN	HISTORIAL	ACCIONES
Samsung	Marca Samsung	<p>Creado el 15 de julio de 2022 a las 12:40</p> <p>Última Modificación 29 de agosto de 2022 a las 15:19</p>	✎
Lenovo	Sin descripción	<p>Creado el 19 de julio de 2022 a las 00:10</p> <p>Última Modificación 19 de julio de 2022 a las 00:10</p>	✎

Rows per page: 10 | 1-2 of 2

Figura 131. Interfaz gráfica de Listar marcas

Principal > Panel > Marcas > Nueva

Nueva Marca

Registro de una nueva marca

Nueva Marca

Formulario de registro de una nueva marca

Nombre *

Especifique el nombre de la marca

Descripción

Escriba algo que describa a la marca

Registrar Marca

Información

En esta sección podrás crear una nueva marca la cual te servirá para asignar posteriormente a tus productos.

© 2022 - Panel de administración Compumars | Elaborado por Compumars.com

Figura 132. Interfaz gráfica de Nueva marca

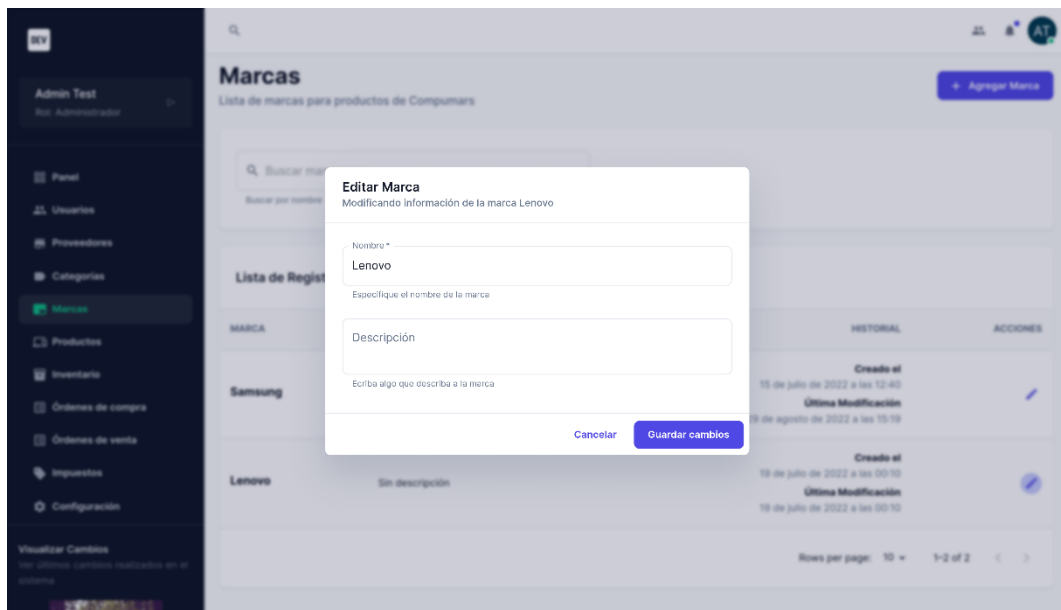


Figura 133. Interfaz gráfica de Modificar marca

4. Módulo de gestión de servicios

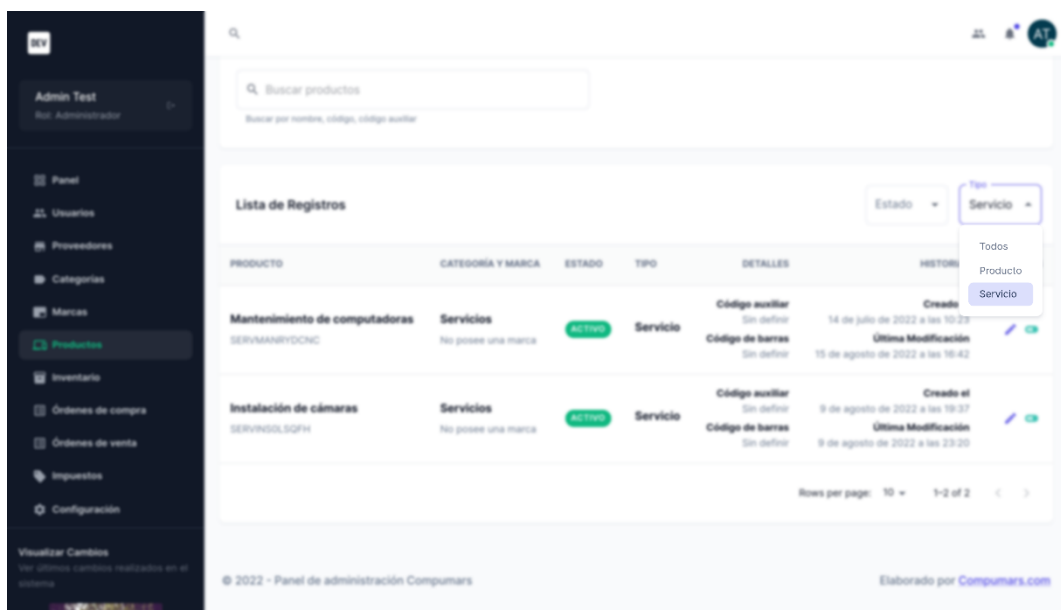


Figura 134. Interfaz gráfica de Filtrar listado de servicios

5. Módulo de gestión de inventario

Inventario
Lista de productos en el inventario de Computars

Buscar en el inventario
Buscar por código o nombre del producto

Lista de Registros

PRODUCTO	CANTIDAD FÍSICA	CANTIDAD DISPONIBLE	CANTIDAD RESERVADA	INFORMACIÓN	HISTORIAL	ACCIONES
Computadora Lenovo core i7 Código del producto: PRODCOMKINZZ	10 Artículos	6 Artículos	4 Artículos	Precio Actual \$1478.00 Precio de Venta \$1500.00	20 de julio de 2022 a las 08:55 Última Modificación 29 de agosto de 2022 a las 12:35	
Apple iPhone 13 Pro (256GB, Sierra Blue) #C94B68 Código del producto: PRODAPPVEAZ5G Código del ítem: CODIGONUEVOCOLORI	4 Artículos	0 Artículos	4 Artículos	Precio Actual \$87.50 Precio de Venta \$125.00	25 de julio de 2022 a las 16:15 Última Modificación 26 de julio de 2022 a las 16:16	
Apple iPhone 13 Pro (256GB, Sierra Blue) #Z200ZZ Código del producto: PRODAPPVEAZ5G Código del ítem: CODIGONUEVOCOLOR	13 Artículos	1 Artículo	12 Artículos	Precio Actual \$126.15 Precio de Venta \$135.00	28 de julio de 2022 a las 16:11 Última Modificación 26 de julio de 2022 a las 16:16	
				Precio Actual	Creado el	

Figura 135. Interfaz gráfica de Ver Inventario

Detalles del Inventario
Lista de movimientos del producto Apple iPhone 13 Pro (256GB, Sierra Blue)

[Ver Producto](#)

Apple iPhone 13 Pro (256GB, Sierra Blue)
#Z200ZZ
PRODAPPVEAZ5G ACTIVO

13 existentes 1 disponibles 12 reservados

Precio de venta \$135.00 **Precio actual** \$126.15 **Impuesto actual** IVA 12%

Código auxiliar CODE **Código de barras** BARRAS **Código del ítem** CODIGONUEVOCOLOR

Lista de Movimientos

PRODUCTO	TIPO	CANTIDAD	PRECIO	TOTAL	HISTORIAL	ACCIONES
Apple iPhone 13 Pro (256GB, Sierra Blue) (Color: #220022) Referencia: PRODAPPVEAZ5G	Orden de venta POR PAGAR	1 Artículo	\$135.00 (Imp. Excl.) \$151.20 (Imp. Incl.)	\$135.00 (Imp. Excl.) \$151.20 (Imp. Incl.)	30 de agosto de 2022 a las 09:17 Última Modificación 30 de agosto de 2022 a las 09:17	
Apple iPhone 13 Pro (256GB, Sierra Blue) (Color: #220022)	Orden de venta	1	\$135.00 (Imp. Excl.)	\$135.00 (Imp. Excl.)	29 de agosto de 2022 a las 15:08 Última Modificación	

Figura 136. Interfaz gráfica de Detalles de un producto del inventario

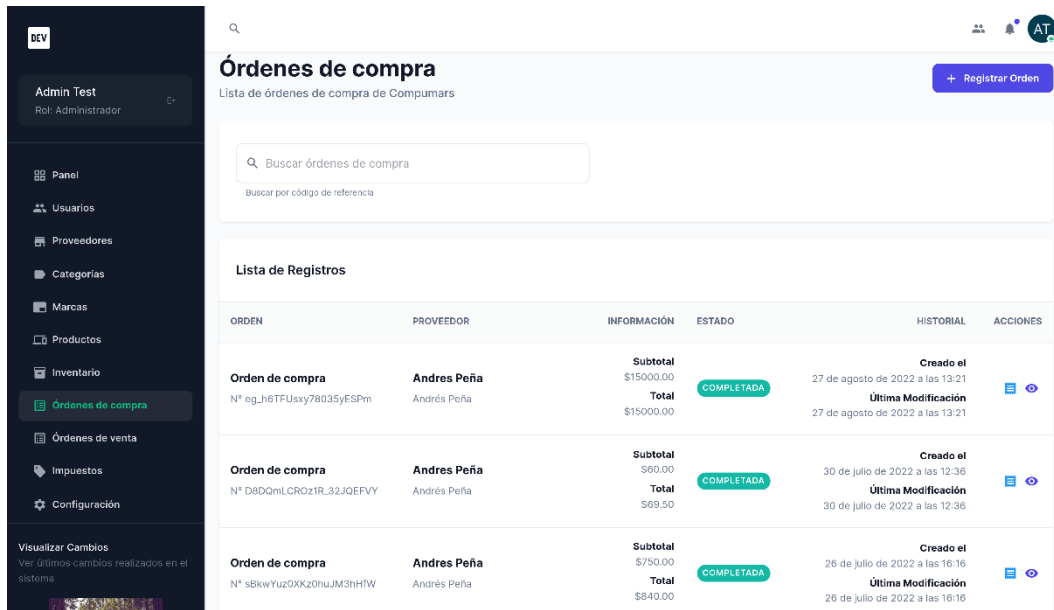


Figura 137. Interfaz gráfica de Listar órdenes de compra

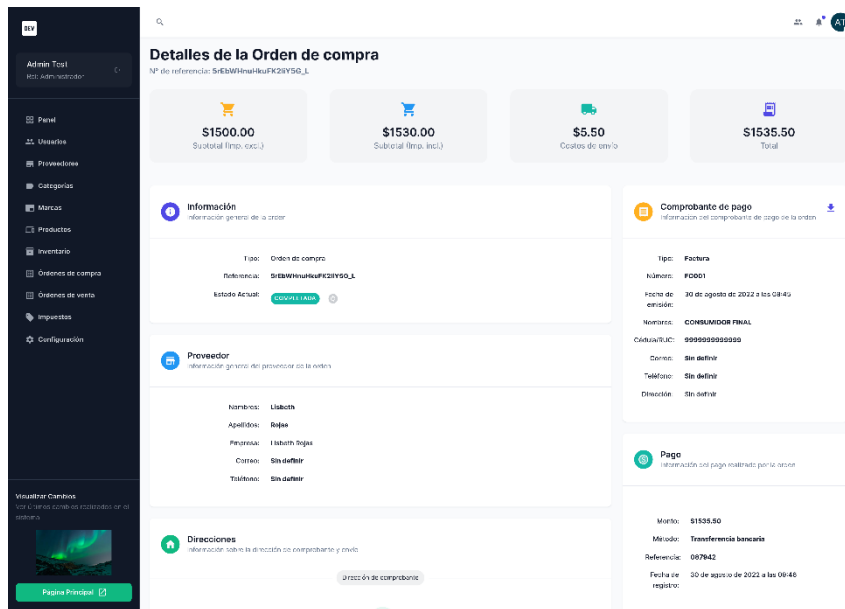


Figura 138. Interfaz gráfica de Detalles de una orden de compra

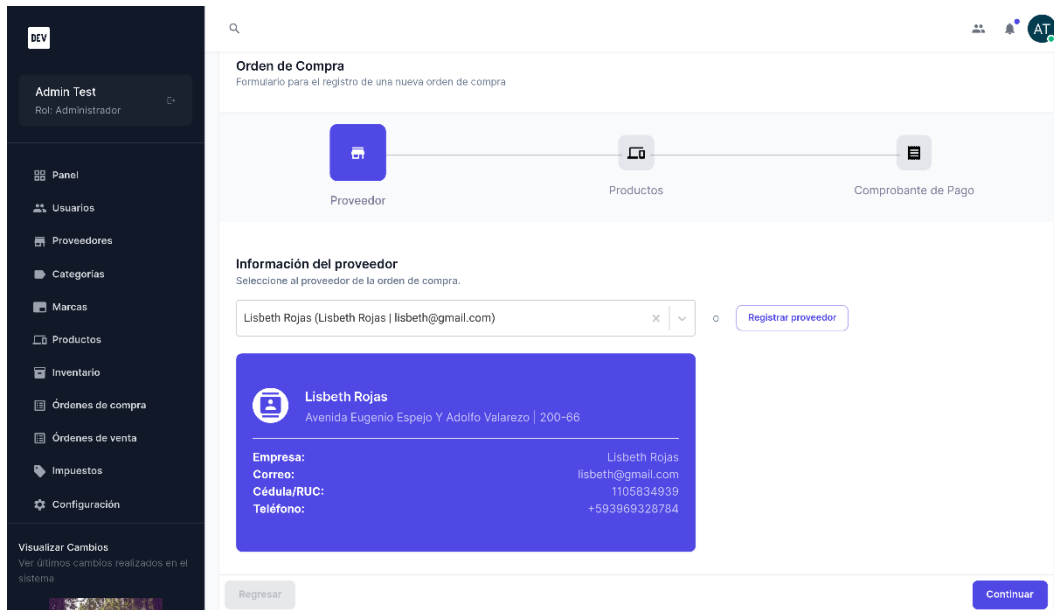


Figura 139. Interfaz gráfica de Nueva orden de compra (Etapa de seleccionar proveedor)

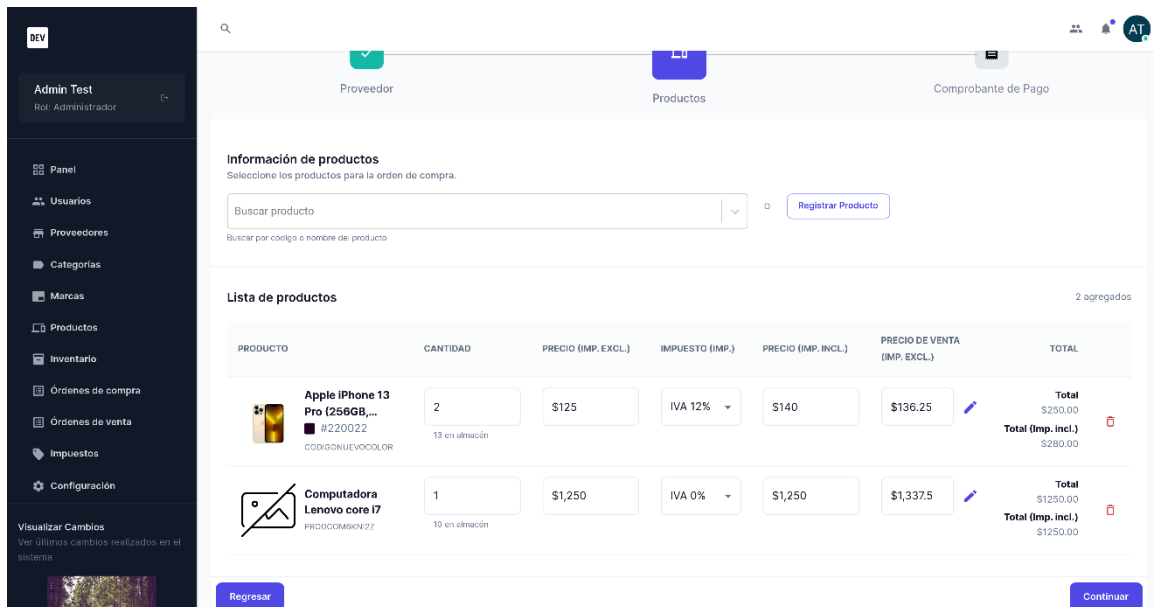


Figura 140. Interfaz gráfica de Nueva orden de compra (Etapa de seleccionar productos)

Comprobante de pago
Registro del comprobante y la forma de pago por los productos seleccionados.

\$1500.00 Subtotal (Imp. excl.)	\$1530.00 Subtotal (Imp. incl.)	\$5.50 Costos de envío	\$1535.50 Total
-------------------------------------------	-------------------------------------------	----------------------------------	---------------------------

Tipo de documento *
 Factura Nota de venta

Número del documento *

Especifique el número del documento

Costo de envío *

Tipo de pago *
 Efectivo Transferencia bancaria

Referencia del pago

Escriba una referencia que lo identifique el pago

[Regresar](#) [Continuar](#)

Figura 141. Interfaz gráfica de Nueva orden de compra (Etapa de registro del comprobante de pago)

6. Módulo de gestión de ventas

Principal > Carrito de compras

Direcciones

Victor Rojas
 1105834939 | victandres9@gmail.com
 Avenida Eugenio Espejo
 Adolfo Valarezo
 +593969328489

Dirección de envío
[+ Registrar nueva dirección](#)

Usar también esta dirección para el comprobante de venta

Método de pago

Efectivo Transferencia bancaria

Por favor, transfiera el importe total de la orden a nuestra cuenta bancaria. Los artículos asociados al pedido se reservarán por **2 hora(s)** para usted y se procesará la orden inmediatamente después de recibir y confirmar el pago.

N° de cuenta:
 087947435

Titular:
 Compumars (1105834939001 | contacto@compumars.com)

Detalles:
 Banco de Loja.

Resumen

Apple iPhone 13 Pro (256GB, Sierra Blue) (Color: #220022) #220022	\$151.20
Cantidad 1	- 1 +
Bateria 12V 3A	\$15.12
Cantidad 1	- 1 +
Subtotal	\$146.50
Impuestos	\$17.82
Costo de envío	\$0.00
Total	\$166.32

[Realizar Pedido](#)

Figura 142. Interfaz gráfica del Carrito de compras

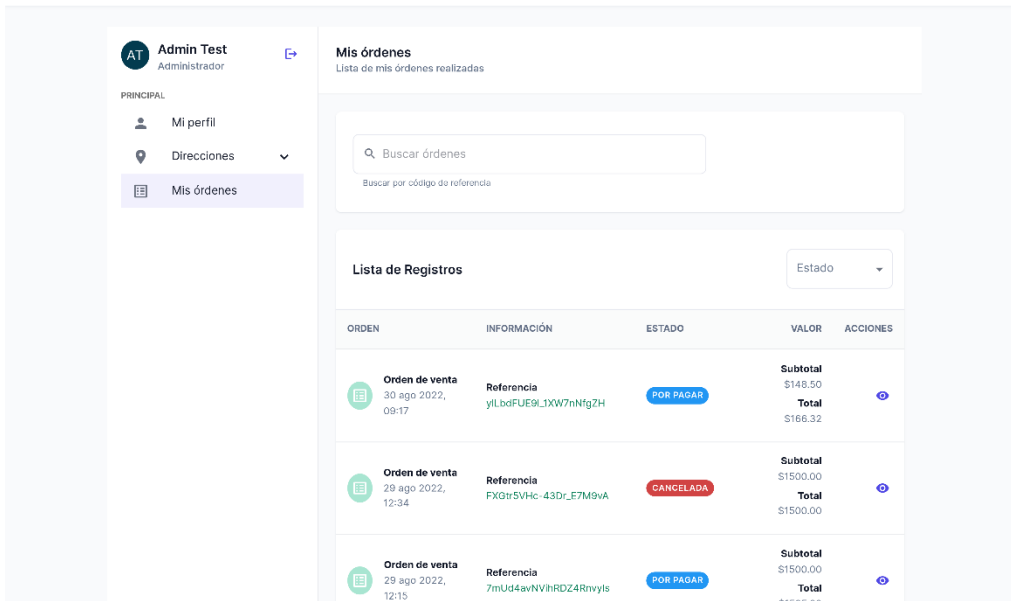


Figura 143. Interfaz gráfica de órdenes realizadas por el cliente

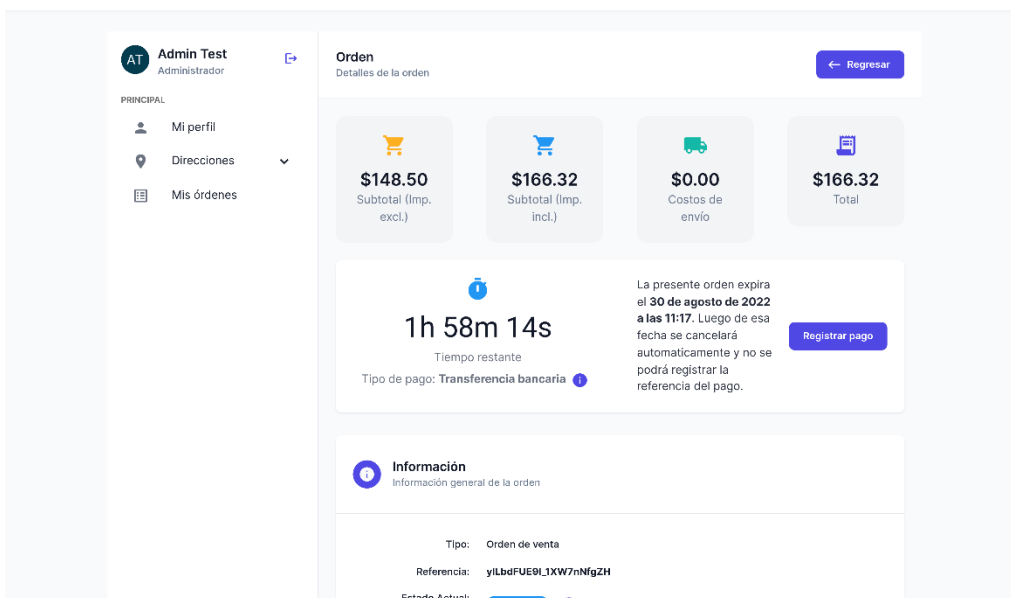


Figura 144. Interfaz gráfica de Ver detalles de una orden

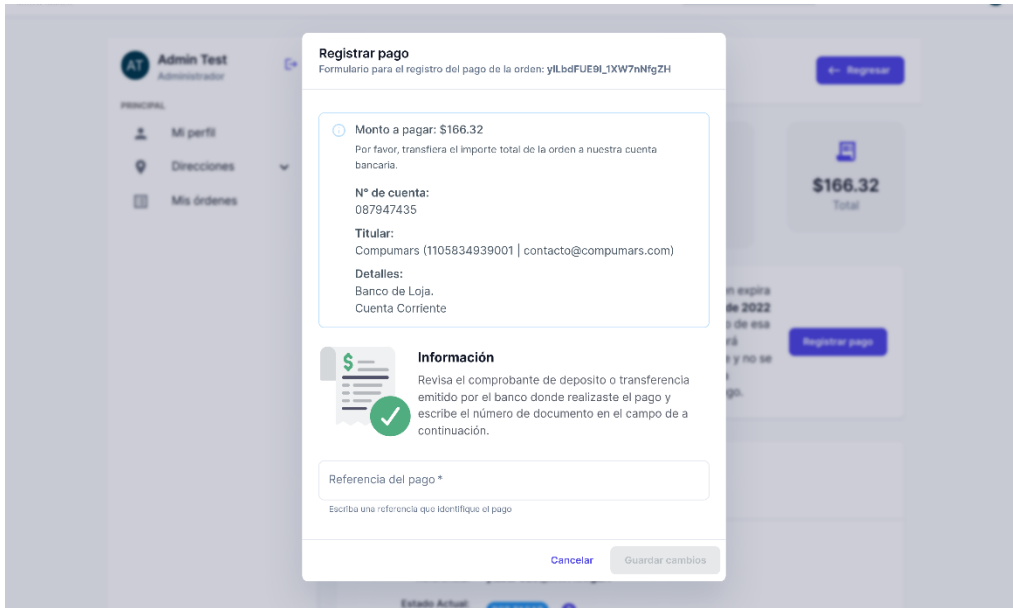


Figura 145. Interfaz gráfica de Registrar constancia del pago de una orden

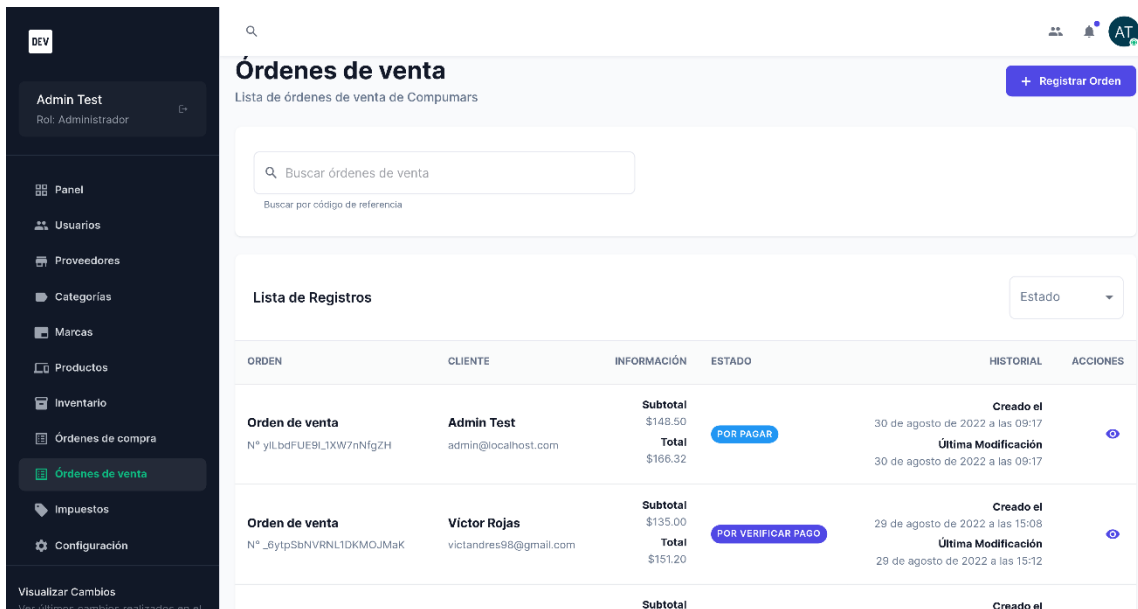


Figura 146. Interfaz gráfica de Listar órdenes de venta

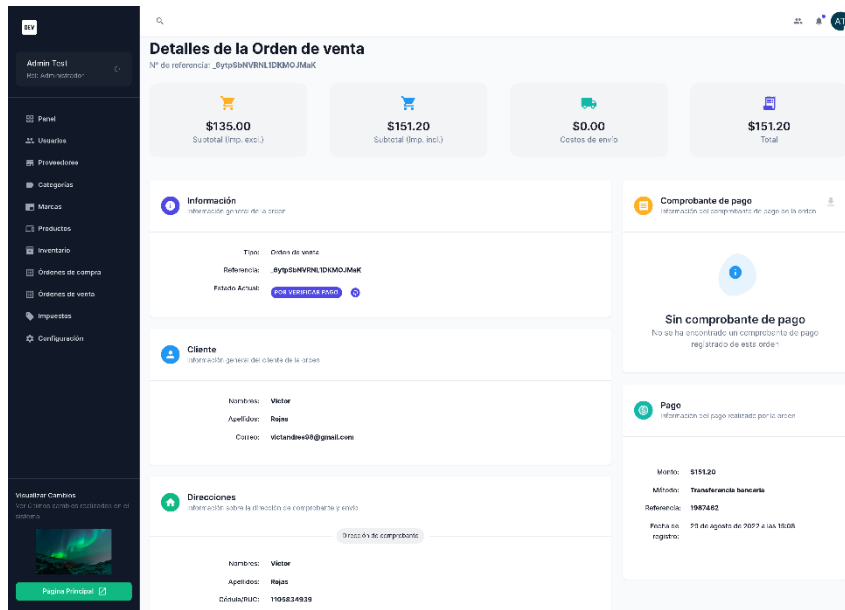


Figura 147. Interfaz gráfica de Detalles de una orden de venta

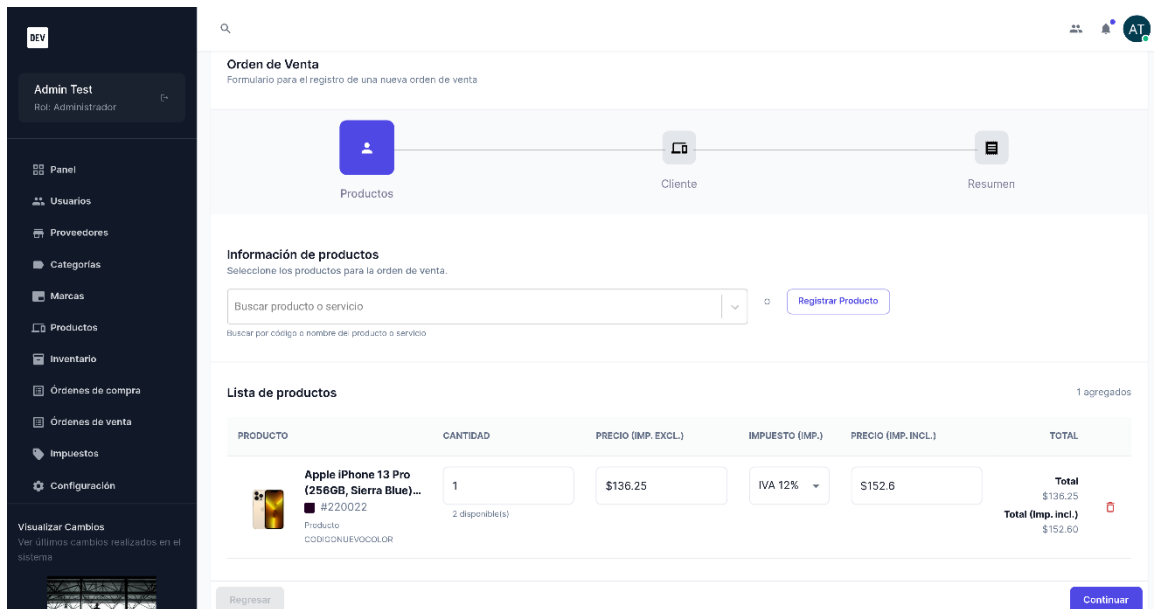


Figura 148. Interfaz gráfica de Nueva orden de venta (Etapa de seleccionar productos)

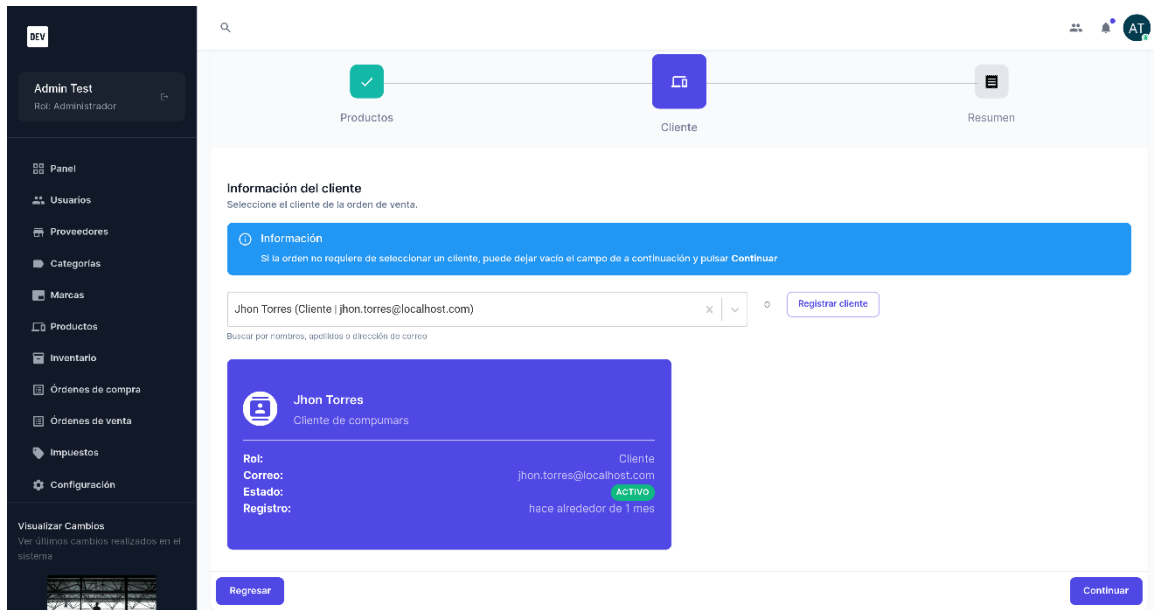


Figura 149. Interfaz gráfica de Nueva orden de venta (Etapa de seleccionar cliente)

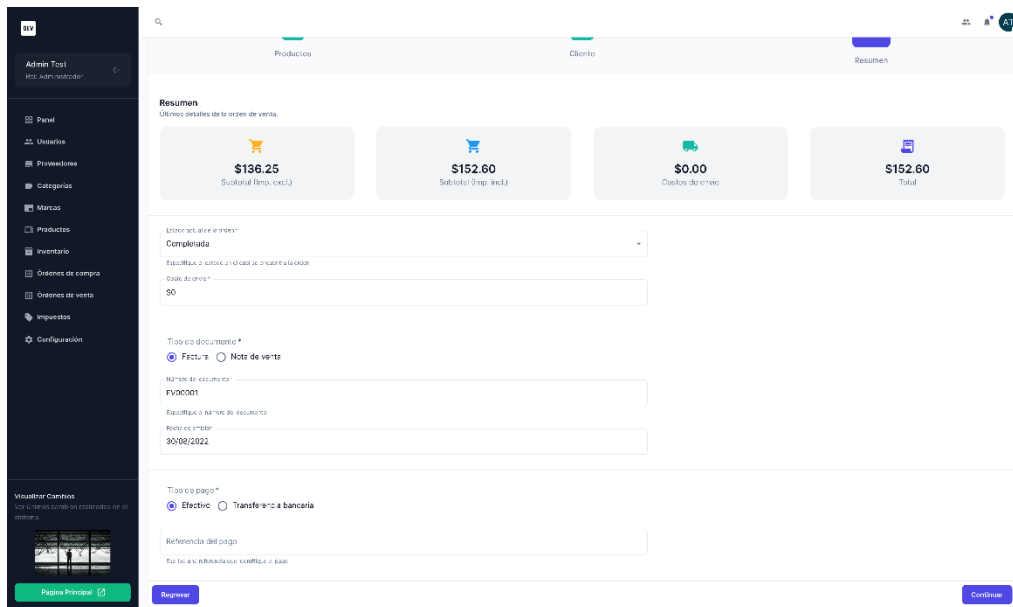


Figura 150. Interfaz gráfica de Nueva orden de venta (Etapa de resumen)

Impuestos
Lista de impuestos registrados para Compumars

+ Registrar impuesto

Buscar impuestos
Buscar por nombre del impuesto

Lista de Registros

DENOMINACIÓN	TASA DE IMPUESTO	HISTORIAL	ACCIONES
IVA 12%	12	<p>Creado el 16 de julio de 2022 a las 14:58</p> <p>Última Modificación 16 de julio de 2022 a las 14:58</p>	
IVA 0%	0	<p>Creado el 16 de julio de 2022 a las 14:58</p> <p>Última Modificación 16 de julio de 2022 a las 14:58</p>	
IVA 16%	16	<p>Creado el 28 de julio de 2022 a las 11:09</p> <p>Última Modificación 28 de julio de 2022 a las 11:30</p>	

Figura 151. Interfaz gráfica de Listar impuestos

Principal > Panel > Impuestos > Nuevo

Nuevo Impuesto
Registro de un impuesto para Compumars

Nuevo Impuesto
Formulario de registro de un nuevo impuesto

Nombre o denominación *
Especifique el nombre que identificará al impuesto

Tasa de impuesto *
0
Especifique un valor que se encuentre entre 0 y 100

Registrar Impuesto

Información
En esta sección podrás crear un nuevo Impuesto, el mismo que servirá para llevar el control del valor agregado de tus productos entrantes y salientes del inventario.

© 2022 - Panel de administración Compumars
Elaborado por Compumars.com

Figura 152. Interfaz gráfica de Nuevo impuesto

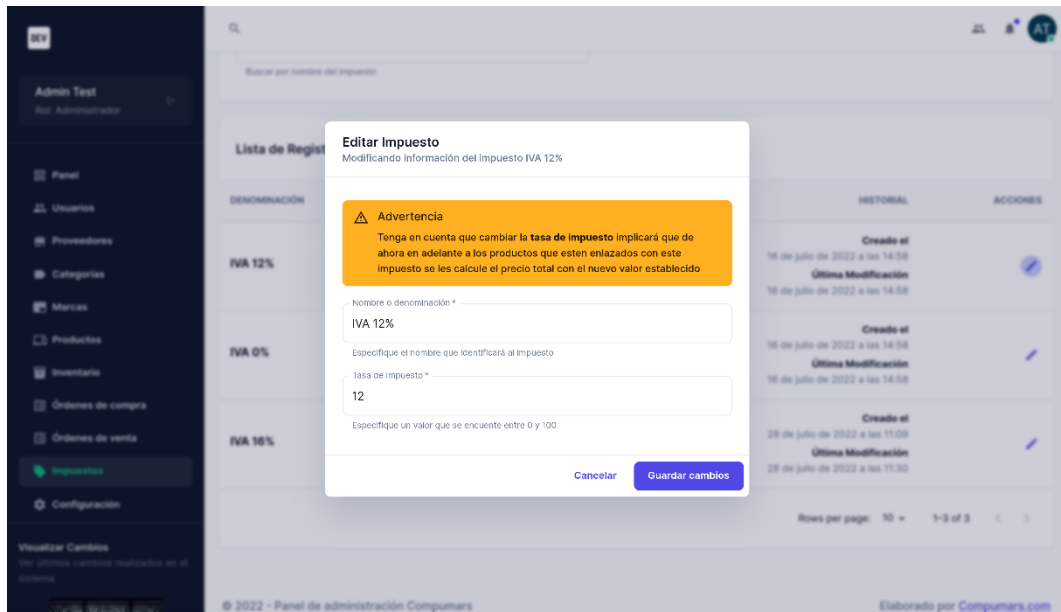


Figura 153. Interfaz gráfica de Modificar impuesto

**Desarrollo de una Aplicación Web Progresiva para la
gestión y comercialización de productos y servicios para la
Empresa “Compumars” de la ciudad de Loja**

Manual de instalación de la Aplicación Web Progresiva

1. Introducción

En el presente documento se cubren aspectos relacionados a la configuración e instalación de tecnologías y herramientas para el correcto funcionamiento de la Aplicación Web Progresiva desarrollada en el presente TT.

1.1. Objetivo

El objetivo principal de este manual es describir los pasos necesarios para la configuración e instalación de las distintas tecnologías y herramientas indispensables para la ejecución de la Aplicación Web Progresiva relacionada a la gestión y comercialización de productos y servicios para la empresa Compumars dentro de un entorno de desarrollo y producción.

1.2. Alcance

El presente manual de instalación está dirigido para todos los interesados con conocimientos informáticos que pretendan instalar y ejecutar de manera correcta la presente Aplicación Web Progresiva.

1.3. Requisitos Previos

Los requisitos necesarios para el funcionamiento de la solución informática son:

- Node.js (véase: <https://nodejs.org/en/download>)
- Docker (véase: <https://docs.docker.com/get-docker>)
- Kubernetes (minikube) (véase: <https://minikube.sigs.k8s.io/docs/start>)
- kubectl (véase: <https://kubernetes.io/docs/tasks/tools>)
- Skaffold (véase: <https://skaffold.dev/docs/install>)
- Acceso al repositorio oficial de Compumars (véase: <https://github.com/victand98/compumars>)
- Git (véase: <https://git-scm.com/downloads>)
- Editor de código (de preferencia *Visual Studio Code*) (véase: <https://code.visualstudio.com/Download>)
- Línea de comandos (CMD)

1.4. Restricciones

- Para realizar cambios en el código del proyecto y hacerlos publicarlos en el repositorio del presente proyecto se necesitará tener autorización (ser miembro del repositorio).

- Las versiones antiguas de las herramientas o tecnologías mencionadas en los requisitos previos pueden no ser compatibles con las dependencias utilizadas en el presente proyecto.

2. Instalación y Configuración del Software

En este apartado se cubre los distintos pasos necesarios a seguir para realizar la correcta instalación, configuración y puesta en marcha de la presente solución informática.

2.1. Entorno de Desarrollo

Para comenzar, se debe clonar el repositorio de Compumars desde Github:

```
git clone https://github.com/victand98/compumars.git
```

Ejecutar el clúster de kubernetes de manera local utilizando minikube:

```
minikube start
```

Asimismo, se debe crear las variables de entorno necesarias para el funcionamiento de la aplicación, para ello se debe ejecutar cada una de las siguientes líneas de comando utilizando la herramienta kubectl:

```
# MONGO_URI: Dirección de la base de datos MongoDB
kubectl create secret generic mongo-uri --from-literal=MONGO_URI=<valor>

# JWT_KEY: Clave secreta para la construcción y verificación de Json Web
Token (JWT)
kubectl create secret generic jwt-secret --from-literal=JWT_KEY=<valor>

# ACTIVATION_TOKEN_MAX_AGE: Valor que indica el número de horas que tiene
un nuevo usuario para verificar su cuenta mediante su correo electrónico
kubectl create secret generic activation-token-max-age --from-
literal=ACTIVATION_TOKEN_MAX_AGE=<valor>

# USER_EMAIL: Credencial de acceso para configuración de envío de correos
kubectl create secret generic user-email --from-
literal=USER_EMAIL=<valor>

# PASSWORD_EMAIL: Credencial de acceso para configuración de envío de
correos
kubectl create secret generic password-email --from-
literal=PASSWORD_EMAIL=<valor>
```

```
# CLOUD_NAME: Nombre de la nube del servicio de Cloudinary
kubectl create secret generic cloud-name --from-
literal=CLOUD_NAME=<valor>

# API_KEY: Clave del API de Cloudinary
kubectl create secret generic api-key --from-literal=API_KEY=<valor>

# API_SECRET: Clave secreta del API de Cloudinary
kubectl create secret generic api-secret --from-
literal=API_SECRET=<valor>
```

Además, es necesario activar el controlador NGINX Ingress, para ello ejecute el siguiente comando:

```
minikube addons enable ingress
```

Luego se debe crear un alias para acceder al sitio web desde el navegador. Para ello primeramente se debe obtener la dirección IP del clúster de Kubernetes:

```
minikube ip
```

El comando anterior mostrará la IP en la cual se encuentra ejecutándose el clúster, se debe copiar esa ip y pegarlo en el fichero *hosts* del sistema operativo:

```
sudo nano /etc/hosts
```

Agregar la siguiente línea en el fichero *hosts* creará un alias para acceder mediante <https://compumars.dev> desde el navegador cuando la aplicación esté ejecutándose.

```
# /etc/hosts
- - -
<IP copiada>    compumars.dev
- - -
```

Por último, se debe encontrar en el directorio raíz del repositorio de Compumars previamente clonado y ejecutar la siguiente instrucción para poner en marcha la aplicación:

```
skaffold dev
```

Eso ejecutará la aplicación en un entorno de desarrollo, la cual se puede acceder, como se mencionó anteriormente, desde la dirección url <https://compumars.dev>, sin embargo, es necesario que se instalen las dependencias respectivas tanto para el cliente como el servidor, esto con el fin de poder realizar cambios en el código fuente.

En el directorio raíz de Compumars, ejecutar:

```
cd client
```

Luego:

```
npm install
```

De la misma manera para el servidor, asimismo, en el directorio raíz de compumars ejecutar:

```
cd server
```

Seguidamente:

```
npm install
```

Se puede ejecutar el conjunto de pruebas elaboradas en el servidor por medio del comando:

```
npm run test
```

2.2. Entorno de Producción

La instrucción `skaffold dev` ejecuta la aplicación leyendo todos los archivos de manifiesto (yaml) ubicados tanto en el directorio `/infra/k8s/` como en `/infra/k8s-dev/`, además pone en funcionamiento los archivos *Dockerfile* ubicados en las carpetas *client* y *server*.

Para ejecutar la aplicación localmente en un entorno de producción se debe ejecutar la siguiente instrucción:

```
skaffold dev -p prod
```

Por su lado, el comando anterior ejecutará los archivos de manifiesto encontrados en las carpetas `/infra/k8s/` y `/infra/k8s-prod/` respectivamente, y en este caso pondrá en marcha los archivos *Dockerfile.prod* ubicados en las carpetas *client* y *server*.

De esta manera sencilla se puede ejecutar la aplicación en un entorno de producción gracias a la ayuda que brindan los archivos de manifiesto previamente preparados en el directorio `/infra/` del proyecto. Se puede cambiar el valor de las variables de entorno en caso de ser necesario que se apliquen otros valores, para ello se puede seguir la guía oficial de Kubernetes (<https://kubernetes.io/docs/concepts/configuration/secret/>).

Sin embargo, todos los cambios que se realicen sobre la rama ***main***, *procederán* a ejecutar una serie de flujos de trabajo (*workflows*), estos son archivos de manifiesto que se encuentran ubicados en el directorio `/.github/workflows/`. Estos flujos de trabajo se encargan de desplegar de manera automática los cambios realizados en los directorios `/client/`, `/server/` o `/infra/` en la plataforma de servicios en la nube, que para este caso es DigitalOcean. En la dirección <https://github.com/victand98/compumars/actions> se puede ver el historial de ejecución de los distintos flujos de trabajo llevados a cabo hasta el momento.

Anexo 6. Pruebas unitarias y de integración de la Aplicación Web Progresiva

Para comprobar el correcto funcionamiento de los procedimientos elaborados durante la codificación de la Aplicación Web Progresiva se realizaron un conjunto de pruebas unitarias y de integración tomando en cuenta las funcionalidades más relevantes que posee esta solución informática. A continuación, se muestran las pruebas mencionadas:

```
compumars - dni.test.ts
1 import { validateDNI } from "@compumars/common";
2
3 describe("Prueba: Cédula/RUC", () => {
4   it("La cédula ingresada es correcta", () => {
5     const dni = "1105834939";
6     expect(validateDNI(dni)).toBe(true);
7   });
8
9   it("La cédula ingresada no es correcta", () => {
10    const dni = "1105834933";
11    expect(validateDNI(dni)).toBe(false);
12  });
13
14  it("El RUC ingresado es correcto", () => {
15    const dni = "1105834939001";
16    expect(validateDNI(dni)).toBe(true);
17  });
18
19  it("El RUC ingresado no es correcto", () => {
20    const dni = "1105834939002";
21    expect(validateDNI(dni)).toBe(false);
22  });
23
24  it("El número de Cédula/RUC no es válido ya que posee menos de 10 caracteres", () => {
25    const dni = "110583493";
26    expect(validateDNI(dni)).toBe(false);
27  });
28 });
29
```

Figura 154. Pruebas unitarias para la validación de Cédula/RUC

```
compumars - password.test.ts
1 import { Password } from "../password";
2
3 describe("Prueba: Encriptar clave", () => {
4   const password = "Password-123";
5
6   it("La clave ingresada ha sido encriptada correctamente", async () => {
7     const result = await Password.toHash(password);
8     expect(result).not.toBe(password);
9   });
10
11  it("La clave ingresada es igual a la clave encriptada", async () => {
12    const storedPassword = await Password.toHash(password);
13    const passwordsMatch = await Password.compare(storedPassword, password);
14
15    expect(passwordsMatch).toBe(true);
16  });
17
18  it("La clave ingresada no es igual a la clave encriptada", async () => {
19    const anotherPassword = "Password-12";
20    const storedPassword = await Password.toHash(password);
21    const passwordsMatch = await Password.compare(
22      storedPassword,
23      anotherPassword
24    );
25
26    expect(passwordsMatch).toBe(false);
27  });
28 });
29
```

Figura 155. Pruebas unitarias para la encriptación y validación de contraseñas

```

compumars - auth.test.ts

1 describe("Prueba: Iniciar Sesión", () => {
2   it("Recibir un error cuando el usuario se autentica con un correo no registrado", async () => {
3     await api.post(`${AUTH_ENDPOINT}signin`).send(signInData).expect(400);
4   });
5
6   it("Recibir un error cuando el usuario se autentica con una contraseña inválida", async () => {
7     await api.post(`${AUTH_ENDPOINT}signup`).send(signUpData).expect(201);
8
9     await api
10      .post(`${AUTH_ENDPOINT}signin`)
11      .send({ ...signInData, password: "wrongPassword" })
12      .expect(400);
13   });
14
15   it("Recibir una cookie cuando el inicio de sesión es válido", async () => {
16     const signUpResponse = await api
17       .post(`${AUTH_ENDPOINT}signup`)
18       .send(signUpData)
19       .expect(201);
20
21     // Activate account
22     await UserModel.findByIdAndUpdate(signUpResponse.body.id, { status: true });
23
24     const signInResponse = await api
25       .post(`${AUTH_ENDPOINT}signin`)
26       .send(signInData)
27       .expect(200);
28
29     expect(signInResponse.get("Set-Cookie")).toBeDefined();
30   });
31 });

```

Figura 156. Pruebas de integración de Iniciar Sesión

```

compumars - user.test.ts

1 it("Recibir una lista de usuarios", async () => {
2   const cookie = await global.signin();
3   const { body } = await api.get(USER_ENDPOINT).set("Cookie", cookie);
4   expect(body.docs).toBeDefined();
5 });
6
7 it("Recibir un error al obtener la lista de usuarios sin autenticarse", async () => {
8   await api.get(USER_ENDPOINT).expect(401);
9 });

```

Figura 157. Pruebas de integración de listar usuarios

```

compumars - user.test.ts

1 it("Registrar un usuario", async () => {
2   const userData = await getUserData();
3   const cookie = await global.signin();
4   await api
5     .post(USER_ENDPOINT)
6     .set("Cookie", cookie)
7     .send(userData)
8     .expect(201);
9 });
10
11 it("Registrar un usuario con datos incompletos", async () => {
12   const { role, ...userData } = await getUserData();
13   const cookie = await global.signin();
14   await api
15     .post(USER_ENDPOINT)
16     .set("Cookie", cookie)
17     .send(userData)
18     .expect(400);
19 });

```

Figura 158. Pruebas de integración de Registrar usuarios

```
compumars - user.test.ts

1  it("Modificar rol de un usuario registrado", async () => {
2    const userData = await getUserData();
3    const cookie = await global.signin();
4    const { body } = await api
5      .post(USER_ENDPOINT)
6      .set("Cookie", cookie)
7      .send(userData)
8      .expect(201);
9
10   const role = await RoleModel.findOne({ name: Roles.cashier });
11
12   const { body: updateBody } = await api
13     .put(`${USER_ENDPOINT}${body.id}`)
14     .set("Cookie", cookie)
15     .send({ role: role?.id })
16     .expect(200);
17
18   expect(updateBody.role.id).toBe(role?.id);
19 });
```

Figura 159. Prueba de integración de Modificar rol de un usuario

```
compumars - user.test.ts

1  it("Dar de baja un usuario", async () => {
2    const userData = await getUserData();
3    const cookie = await global.signin();
4    const { body } = await api
5      .post(USER_ENDPOINT)
6      .set("Cookie", cookie)
7      .send(userData)
8      .expect(201);
9
10   const { body: updateBody } = await api
11     .put(`${USER_ENDPOINT}${body.id}`)
12     .set("Cookie", cookie)
13     .send({ status: !body.status })
14     .expect(200);
15
16   expect(updateBody.status).not.toBe(body.status);
17 });
```

Figura 160. Prueba de integración de Dar de baja un usuario

```
compumars - product.test.ts

1  it("Recibir una lista de productos", async () => {
2    const cookie = await global.signin(Roles.customer);
3    const { body } = await api.get(PRODUCT_ENDPOINT).set("Cookie", cookie);
4    expect(body.docs).toBeDefined();
5  });
```

Figura 161. Prueba de integración de Listar productos


```
compumars - product.test.ts
1 it("Registrar un nuevo producto", async () => {
2   const cookie = await global.signin(Roles.warehouseManager);
3   await api
4     .post(PRODUCT_ENDPOINT)
5     .set("Cookie", cookie)
6     .send(randomProduct)
7     .expect(201);
8 });
9
10 it("Recibir un error al registrar un nuevo producto con información faltante", async () => {
11   const cookie = await global.signin(Roles.warehouseManager);
12   const { name, code, ...productData } = randomProduct;
13   await api
14     .post(PRODUCT_ENDPOINT)
15     .set("Cookie", cookie)
16     .send(productData)
17     .expect(400);
18 });
19
20 it("Recibir un error al registrar dos productos con el mismo código", async () => {
21   const cookie = await global.signin(Roles.warehouseManager);
22   await api
23     .post(PRODUCT_ENDPOINT)
24     .set("Cookie", cookie)
25     .send(randomProduct)
26     .expect(201);
27
28   await api
29     .post(PRODUCT_ENDPOINT)
30     .set("Cookie", cookie)
31     .send(randomProduct)
32     .expect(400);
33 });
```

Figura 162. Pruebas de integración de Registrar productos

```
compumars - product.test.ts
1 it("Modificar un producto", async () => {
2   const cookie = await global.signin(Roles.warehouseManager);
3   const { body } = await api
4     .post(PRODUCT_ENDPOINT)
5     .set("Cookie", cookie)
6     .send(randomProduct)
7     .expect(201);
8
9   const { body: updateBody } = await api
10    .put(`${PRODUCT_ENDPOINT}${body.id}`)
11    .set("Cookie", cookie)
12    .send({ name: "Product new name" })
13    .expect(200);
14
15   expect(randomProduct.name).not.toBe(updateBody.name);
16 });
```

Figura 163. Prueba de integración de Modificar producto

```
compumars - product.test.ts

1  it("Dar de baja un producto", async () => {
2    const cookie = await global.signin(Roles.warehouseManager);
3    const { body } = await api
4      .post(PRODUCT_ENDPOINT)
5      .set("Cookie", cookie)
6      .send(randomProduct)
7      .expect(201);
8
9    const { body: updateBody } = await api
10     .put(`${PRODUCT_ENDPOINT}${body.id}`)
11     .set("Cookie", cookie)
12     .send({ status: !randomProduct.status })
13     .expect(200);
14
15     expect(randomProduct.status).not.toBe(updateBody.status);
16   });
```

Figura 164. Prueba de integración de Dar de baja producto

```
compumars - order.test.ts

1  beforeEach(async () => {
2    await Promise.all([
3      new TaxModel(tax).save(),
4      ProductModel.insertMany(randomProducts),
5      StockModel.insertMany(stocks),
6      KardexModel.insertMany(kardexes),
7    ]);
8  });
9
10 it("Recibir una lista de ordenes", async () => {
11   const cookie = await global.signin(Roles.cashier);
12   const { body } = await api.get(ORDER_ENDPOINT).set("Cookie", cookie);
13   expect(body.docs).toBeDefined();
14 });
15
16 it("Recibir un error al obtener la lista de ordenes sin autenticarse", async () => {
17   await api.get(ORDER_ENDPOINT).expect(401);
18 });
```

Figura 165. Pruebas de integración de Listar órdenes

```
compumars - order.test.ts

1  it("Realizar una orden", async () => {
2    const cookie = await global.signin(Roles.customer);
3    const orderData = getPlaceOrder(randomProducts, kardexes);
4    await api
5      .post(`${ORDER_ENDPOINT}place`)
6      .set("Cookie", cookie)
7      .send(orderData)
8      .expect(201);
9  });
10
11 it("Recibir un error al realizar una orden sin items", async () => {
12   const cookie = await global.signin(Roles.customer);
13   const { items, ...orderData } = getPlaceOrder(randomProducts, kardexes);
14   await api
15     .post(`${ORDER_ENDPOINT}place`)
16     .set("Cookie", cookie)
17     .send(orderData)
18     .expect(400);
19 });
```

Figura 166. Pruebas de integración de Realizar órdenes

```
compumars - order.test.ts

1 it("Registrar orden de venta", async () => {
2   const cookie = await global.signin(Roles.cashier);
3   const orderData = getSellOrder(randomProducts, kardexes);
4   await api
5     .post(`${ORDER_ENDPOINT}sell`)
6     .set("Cookie", cookie)
7     .send(orderData)
8     .expect(201);
9 });
10
11 it("Recibir un error al registrar una orden de venta sin items", async () => {
12   const cookie = await global.signin(Roles.cashier);
13   const { items, ...orderData } = getSellOrder(randomProducts, kardexes);
14   await api
15     .post(`${ORDER_ENDPOINT}sell`)
16     .set("Cookie", cookie)
17     .send(orderData)
18     .expect(400);
19 });
20
21 it("Recibir un error al registrar una orden de venta con un rol no autorizado", async () => {
22   const cookie = await global.signin(Roles.customer);
23   const orderData = getSellOrder(randomProducts, kardexes);
24   await api
25     .post(`${ORDER_ENDPOINT}sell`)
26     .set("Cookie", cookie)
27     .send(orderData)
28     .expect(401);
29 });
```

Figura 167. Pruebas de integración de Registrar órdenes de venta

```
compumars - order.test.ts

1 describe("Órdenes de compra", () => {
2   it("Registrar orden de compra", async () => {
3     const cookie = await global.signin(Roles.warehouseManager);
4     const orderData = getPurchaseOrder(randomProducts, kardexes);
5     await api
6       .post(`${ORDER_ENDPOINT}purchase`)
7       .set("Cookie", cookie)
8       .send(orderData)
9       .expect(201);
10    });
11
12    it("Recibir un error al registrar orden de compra sin items", async () => {
13      const cookie = await global.signin(Roles.warehouseManager);
14      const { items, ...orderData } = getPurchaseOrder(
15        randomProducts,
16        kardexes
17      );
18      await api
19        .post(`${ORDER_ENDPOINT}purchase`)
20        .set("Cookie", cookie)
21        .send(orderData)
22        .expect(400);
23    });
24
25    it("Recibir un error al registrar orden de compra con un rol no autorizado", async () => {
26      const cookie = await global.signin(Roles.cashier);
27      const { items, ...orderData } = getPurchaseOrder(
28        randomProducts,
29        kardexes
30      );
31      await api
32        .post(`${ORDER_ENDPOINT}purchase`)
33        .set("Cookie", cookie)
34        .send(orderData)
35        .expect(401);
36    });
37  });
```

Figura 168. Pruebas de integración de Registrar órdenes de compra

```
compumars - payment.test.ts

1 it("Registrar pago de una orden", async () => {
2   const cookie = await global.signin(Roles.customer);
3   const randomOrder = createRandomOrder({
4     orderStatus: OrderStatus.awaitingPayment,
5   });
6   const newOrder = await new OrderModel(randomOrder).save();
7   await new PaymentModel({ ...randomPayment, order: newOrder.id }).save();
8   await api
9     .put(`${ORDER_ENDPOINT}/payment/${newOrder.id}`)
10    .set("Cookie", cookie)
11    .send({ reference: randomPayment.reference })
12    .expect(200);
13 });
14
15 it("Recibir un error al registrar el pago de una orden con datos incompletos", async () => {
16   const cookie = await global.signin(Roles.customer);
17   const randomOrder = createRandomOrder({
18     orderStatus: OrderStatus.awaitingPayment,
19   });
20
21   await api
22     .put(`${ORDER_ENDPOINT}/payment/${randomOrder._id}`)
23     .set("Cookie", cookie)
24     .send()
25     .expect(400);
26 });
27
28 it("Recibir un error al registrar el pago de una orden que no existe", async () => {
29   const cookie = await global.signin(Roles.customer);
30   const orderId = faker.database.mongodbObjectId();
31   await api
32     .put(`${ORDER_ENDPOINT}/payment/${orderId}`)
33     .set("Cookie", cookie)
34     .send({ reference: randomPayment.reference })
35     .expect(404);
36 });
37
38 it("Recibir un error al registrar el pago de una orden en la que ya se habia registrado el pago", async () => {
39   const cookie = await global.signin(Roles.customer);
40   const randomOrder = createRandomOrder({
41     orderStatus: OrderStatus.awaitingPaymentVerification,
42   });
43   const newOrder = await new OrderModel(randomOrder).save();
44
45   await api
46     .put(`${ORDER_ENDPOINT}/payment/${newOrder.id}`)
47     .set("Cookie", cookie)
48     .send({ reference: randomPayment.reference })
49     .expect(400);
50 });
```

Figura 169. Pruebas de integración de Registrar pago de una orden

Anexo 7. Modelo de Encuesta de Satisfacción



Universidad Nacional de Loja

FACULTAD DE LA ENERGÍA, LAS INDUSTRIAS Y LOS RECURSOS NATURALES NO RENOVABLES CARRERA DE INGENIERÍA EN SISTEMAS / COMPUTACIÓN

Un saludo. La presente encuesta tiene la finalidad de conocer su criterio al probar la Aplicación Web Progresiva para la gestión y comercialización de productos y servicios para la Empresa “Compumars” de la ciudad de Loja. Por favor, responda las siguientes preguntas y muchas gracias de antemano por su colaboración.

Dirección de correo electrónico					
Pregunta	Respuesta				
¿Considera que la Aplicación Web es fácil de usar?	Totalmente en desacuerdo	En desacuerdo	Neutral	De acuerdo	Totalmente de acuerdo
¿Considera que la manera de llevar a cabo la adquisición de productos dentro de la Aplicación Web es adecuada?	Totalmente en desacuerdo	En desacuerdo	Neutral	De acuerdo	Totalmente de acuerdo
¿Considera que mediante el uso de la Aplicación puede agilizar la adquisición de productos?	Totalmente en desacuerdo	En desacuerdo	Neutral	De acuerdo	Totalmente de acuerdo
¿Considera útil que la Aplicación se pueda instalar en un dispositivo directamente?	No es útil	Poco útil	Neutral	Útil	Muy útil
¿Cómo calificaría el proceso de realizar un pedido?	Muy difícil	Difícil	Neutral	Fácil	Muy Fácil
¿Qué dificultad representó para usted el proceso de buscar productos en la aplicación?	Muy difícil	Difícil	Neutral	Fácil	Muy Fácil
¿Qué tan probable es que usted vuelva a utilizar la aplicación?	Nada probable	Poco probable	Neutral	Probable	Muy probable
¿Qué tan probable es que recomiende la Aplicación a otras personas que necesiten adquirir algún producto o servicio en línea?	Nada probable	Poco probable	Neutral	Probable	Muy probable
¿En qué podría mejorar la presente aplicación? Por favor, Indique su recomendación.	Respuesta				

La encuesta se encuentra disponible en Google Forms: <https://forms.gle/gg1gbjJKro3fpm5d9>

Encuesta de Satisfacción

Un saludo. La presente encuesta tiene la finalidad de conocer su criterio al probar la Aplicación Web Progresiva para la gestión y comercialización de productos y servicios para la Empresa "Compumars" de la ciudad de Loja. Por favor, responda las siguientes preguntas y muchas gracias de antemano por su colaboración.

victor.rojas@unl.edu.ec [Switch account](#)

Your email will be recorded when you submit this form

*** Required**

¿Considera que la Aplicación Web es fácil de usar? *

1 2 3 4 5

Totalmente en desacuerdo Totalmente de acuerdo

¿Considera que la manera de llevar a cabo la adquisición de productos dentro de la Aplicación Web es adecuada? *

1 2 3 4 5

Totalmente en desacuerdo Totalmente de acuerdo

¿Considera que mediante el uso de la Aplicación puede agilizar la adquisición de productos? *

1 2 3 4 5

Totalmente en desacuerdo Totalmente de acuerdo

¿Considera útil que la Aplicación se pueda instalar en un dispositivo directamente? *

1 2 3 4 5

No es útil Muy útil

¿Cómo calificaría el proceso de realizar un pedido? *

1 2 3 4 5

Muy difícil Muy Fácil

¿Qué dificultad representó para usted el proceso de buscar productos en la aplicación? *

1 2 3 4 5

Muy difícil Muy Fácil

¿Qué tan probable es que usted vuelva a utilizar la aplicación? *

1 2 3 4 5

Nada probable Muy probable

¿Qué tan probable es que recomiende la Aplicación a otras personas que necesiten adquirir algún producto o servicio en línea? *

1 2 3 4 5

Nada probable Muy probable

¿En qué podría mejorar la presente aplicación? Por favor, Indique su recomendación.

Your answer

Page 1 of 1
Clear form

Figura 170. Encuesta de Satisfacción en Google Forms

Anexo 8. Resultados de la Encuesta de Satisfacción

Los resultados recopilados a continuación corresponden a la encuesta de satisfacción aplicada a los estudiantes de primer ciclo paralelos A y B de la carrera de Ingeniería en Ciencias de la Computación de la Universidad Nacional de Loja. Los resultados obtenidos son los siguientes:

Tabla 49. Participantes y resultados de la Encuesta de Satisfacción

Marca temporal	Email	¿Considera que la Aplicación Web es fácil de usar?	¿Considera que la manera de llevar a cabo la adquisición de productos dentro de la Aplicación Web es adecuada?	¿Considera que mediante el uso de la Aplicación puede agilizar la adquisición de productos?	¿Considera útil que la Aplicación se pueda instalar en un dispositivo directamente?	¿Cómo calificaría el proceso de realizar un pedido?	¿Qué dificultad representó para usted el proceso de buscar productos en la aplicación?	¿Qué tan probable es que usted vuelva a utilizar la aplicación?	¿Qué tan probable es que recomiende la Aplicación a otras personas que necesiten adquirir algún producto o servicio en línea?
11/21/2022 9:59:14	joseph.balcazar@unl.edu.ec	5	5	5	5	5	5	3	3
11/21/2022 9:59:18	miguel.matute@unl.edu.ec	5	5	5	4	5	5	5	5
11/21/2022 9:59:24	mateo.silva@unl.edu.ec	5	5	4	3	5	4	1	5
11/21/2022 10:01:13	jean.encalada@unl.edu.ec	5	5	5	5	4	3	4	3
11/21/2022 10:01:14	bayron.jimenez@unl.edu.ec	5	5	5	4	5	5	4	4
11/21/2022 10:01:44	jostin.jimenez@unl.edu.ec	5	5	5	5	5	5	5	5
11/21/2022 10:02:14	nathaly.angamarca@unl.edu.ec	2	3	4	2	2	4	1	1
11/21/2022 10:02:17	juan.c.alverca@unl.edu.ec	4	5	5	5	5	5	3	4
11/21/2022 10:02:56	dany.cabrera@unl.edu.ec	5	5	5	5	5	5	5	5
11/21/2022 10:03:12	jhostin.tapia@unl.edu.ec	5	5	5	3	4	5	5	5

Marca temporal	Email	¿Considera que la Aplicación Web es fácil de usar?	¿Considera que la manera de llevar a cabo la adquisición de productos dentro de la Aplicación Web es adecuada?	¿Considera que mediante el uso de la Aplicación puede agilizar la adquisición de productos?	¿Considera útil que la Aplicación se pueda instalar en un dispositivo directamente?	¿Cómo calificaría el proceso de realizar un pedido?	¿Qué dificultad representó para usted el proceso de buscar productos en la aplicación?	¿Qué tan probable es que usted vuelva a utilizar la aplicación?	¿Qué tan probable es que recomiende la Aplicación a otras personas que necesiten adquirir algún producto o servicio en línea?
11/21/2022 10:03:41	alexis.ludena@unl.edu.ec	5	4	4	5	5	4	4	4
11/21/2022 10:04:29	jhostin.rojas@unl.edu.ec	4	4	5	3	5	4	3	2
11/21/2022 10:05:51	estefania.torres@unl.edu.ec	5	5	5	5	5	4	5	5
11/21/2022 10:06:25	paul.parra@unl.edu.ec	5	5	5	5	5	5	5	5
11/21/2022 10:07:48	geovanny.romero@unl.edu.ec	5	4	4	2	4	5	3	3
11/21/2022 10:09:04	bryan.troya@unl.edu.ec	5	5	5	4	5	5	5	5
11/21/2022 10:09:40	josselyn.cale@unl.edu.ec	5	3	3	3	4	5	3	5
11/21/2022 10:12:27	lenin.campoverde@unl.edu.ec	5	5	5	5	5	5	5	5
11/21/2022 10:13:14	mateo.rojas@unl.edu.ec	5	5	4	3	5	5	4	4
11/21/2022 10:13:41	isaac.ramon@unl.edu.ec	5	5	5	4	4	5	4	4
11/21/2022 10:13:49	luis.d.armijos.r@unl.edu.ec	5	5	5	5	5	4	4	4
11/21/2022 11:08:39	abel.mora@unl.edu.ec	5	4	4	5	4	5	3	5
11/21/2022 11:08:39	alejandro.villamagua@unl.edu.ec	5	3	3	5	5	5	5	4

Marca temporal	Email	¿Considera que la Aplicación Web es fácil de usar?	¿Considera que la manera de llevar a cabo la adquisición de productos dentro de la Aplicación Web es adecuada?	¿Considera que mediante el uso de la Aplicación puede agilizar la adquisición de productos?	¿Considera útil que la Aplicación se pueda instalar en un dispositivo directamente?	¿Cómo calificaría el proceso de realizar un pedido?	¿Qué dificultad representó para usted el proceso de buscar productos en la aplicación?	¿Qué tan probable es que usted vuelva a utilizar la aplicación?	¿Qué tan probable es que recomiende la Aplicación a otras personas que necesiten adquirir algún producto o servicio en línea?
11/21/2022 11:09:07	kevin.f.guerrero@unl.edu.ec	5	5	5	5	5	5	5	5
11/21/2022 11:09:31	ariel.gonzalez@unl.edu.ec	5	5	5	5	5	5	4	4
11/21/2022 11:09:35	josue.asanza@unl.edu.ec	4	5	5	4	5	4	5	4
11/21/2022 11:09:36	ricardo.ochoa@unl.edu.ec	5	4	5	5	4	5	4	4
11/21/2022 11:09:39	cristian.correa@unl.edu.ec	4	4	4	4	4	5	4	4
11/21/2022 11:09:43	santiago.r.guachizaca@unl.edu.ec	5	5	5	3	4	5	4	5
11/21/2022 11:11:03	santiago.s.ordonez@unl.edu.ec	4	5	4	5	4	4	4	4
11/21/2022 11:11:04	brandon.medina@unl.edu.ec	4	4	4	4	4	4	4	4
11/21/2022 11:11:08	darwin.a.sarango@unl.edu.ec	5	5	4	4	4	5	5	4
11/21/2022 11:11:30	darwin.granda@unl.edu.ec	5	5	5	5	5	5	4	4
11/21/2022 11:11:38	brian.aguinsaca@unl.edu.ec	5	4	5	5	5	5	4	4
11/21/2022 11:11:50	betty.suquilanda@unl.edu.ec	4	4	4	4	2	4	3	4

Marca temporal	Email	¿Considera que la Aplicación Web es fácil de usar?	¿Considera que la manera de llevar a cabo la adquisición de productos dentro de la Aplicación Web es adecuada?	¿Considera que mediante el uso de la Aplicación puede agilizar la adquisición de productos?	¿Considera útil que la Aplicación se pueda instalar en un dispositivo directamente?	¿Cómo calificaría el proceso de realizar un pedido?	¿Qué dificultad representó para usted el proceso de buscar productos en la aplicación?	¿Qué tan probable es que usted vuelva a utilizar la aplicación?	¿Qué tan probable es que recomiende la Aplicación a otras personas que necesiten adquirir algún producto o servicio en línea?
11/21/2022 11:11:59	isabel.morcho@unl.edu.ec	5	4	5	4	4	5	2	3
11/21/2022 11:12:03	alvaro.salinas@unl.edu.ec	5	5	5	4	4	5	5	5
11/21/2022 11:13:19	mathias.medina@unl.edu.ec	5	4	5	4	4	4	3	3
11/21/2022 11:13:37	stiven.jimenez@unl.edu.ec	5	5	5	5	5	5	5	5
11/21/2022 11:14:57	steven.luna@unl.edu.ec	5	4	5	3	4	2	4	4
11/21/2022 11:15:14	tayron.morales@unl.edu.ec	5	5	5	5	5	5	5	5
11/21/2022 11:15:46	anthony.gutierrez@unl.edu.ec	5	5	5	5	5	5	4	4
11/21/2022 11:16:33	jhon.rojas@unl.edu.ec	5	4	5	5	5	4	5	4
11/21/2022 11:16:57	elias.poma@unl.edu.ec	4	5	5	5	5	5	3	3
11/21/2022 11:18:28	alexander.a.sanchez@unl.edu.ec	5	5	4	5	3	5	5	4
11/21/2022 11:19:40	david.a.guaman@unl.edu.ec	4	4	4	5	4	4	4	4
11/21/2022 13:10:05	erika.l.curimilma@unl.edu.ec	5	5	5	5	5	5	5	5

¿Considera que la Aplicación Web es fácil de usar?

47 responses

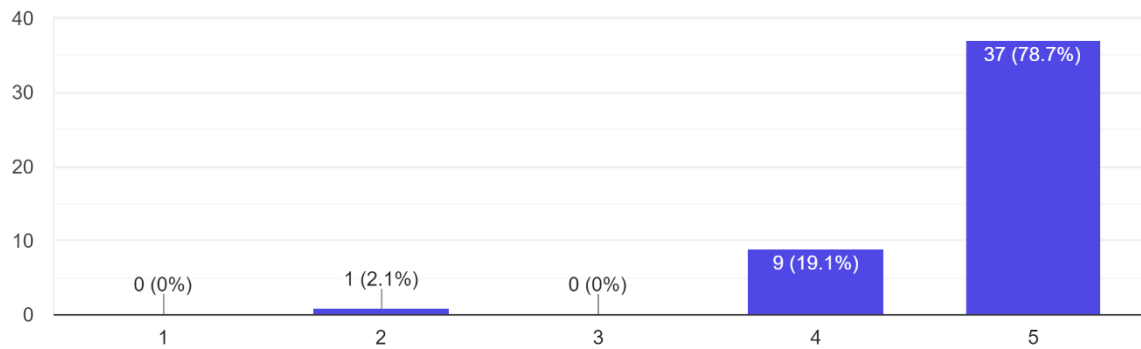


Figura 171. Resultados: ¿Considera que la Aplicación Web es fácil de usar?

¿Considera que la manera de llevar a cabo la adquisición de productos dentro de la Aplicación Web es adecuada?

47 responses

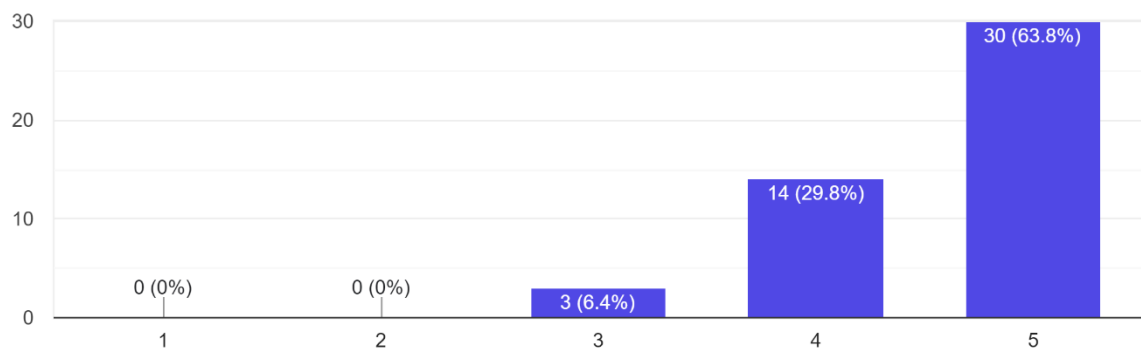


Figura 172. Resultados: ¿Considera que la manera de llevar a cabo la adquisición de productos dentro de la Aplicación Web es adecuada?

¿Considera que mediante el uso de la Aplicación puede agilizar la adquisición de productos?

47 responses

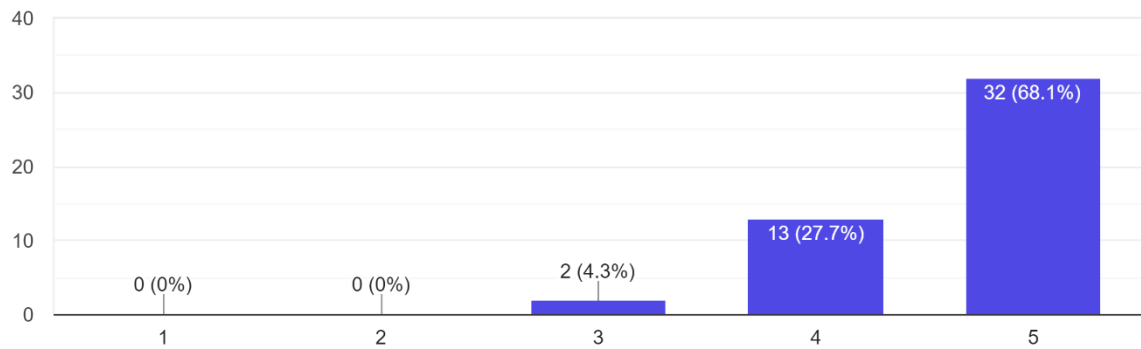


Figura 173. Resultados: ¿Considera que mediante el uso de la Aplicación puede agilizar la adquisición de productos?

¿Considera útil que la Aplicación se pueda instalar en un dispositivo directamente?

47 responses

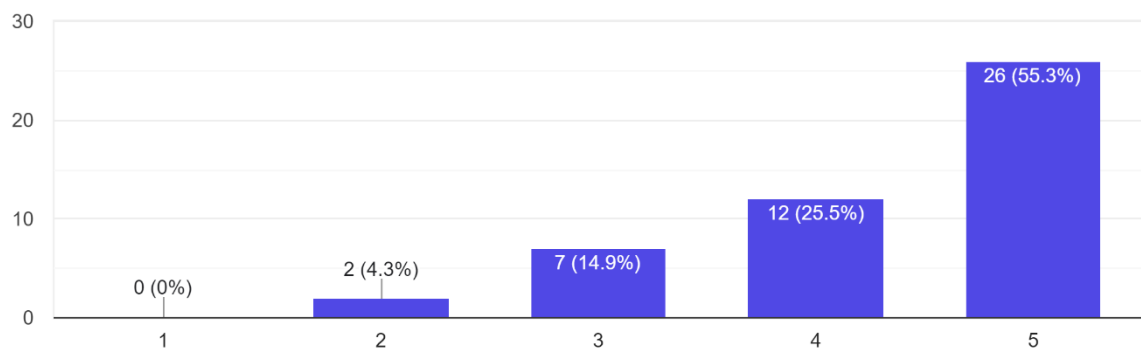


Figura 174. Resultados: ¿Considera útil que la Aplicación se pueda instalar en un dispositivo directamente?

¿Cómo calificaría el proceso de realizar un pedido?

47 responses

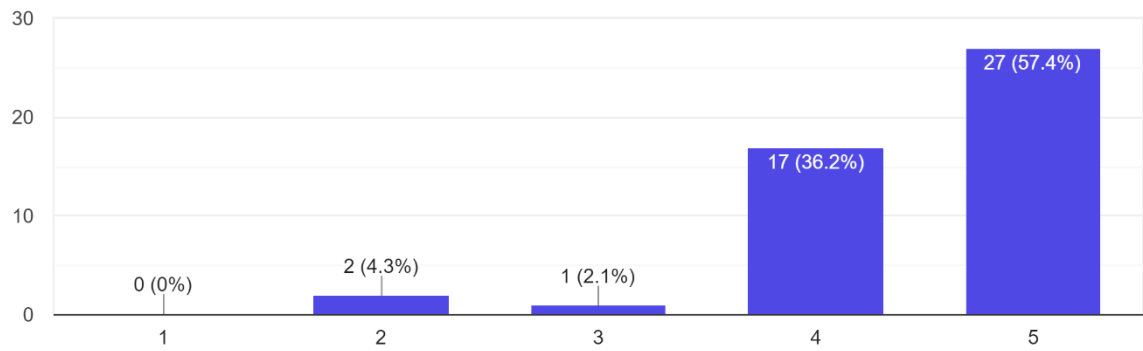


Figura 175. Resultados: ¿Cómo calificaría el proceso de realizar un pedido?

¿Qué dificultad representó para usted el proceso de buscar productos en la aplicación?

47 responses

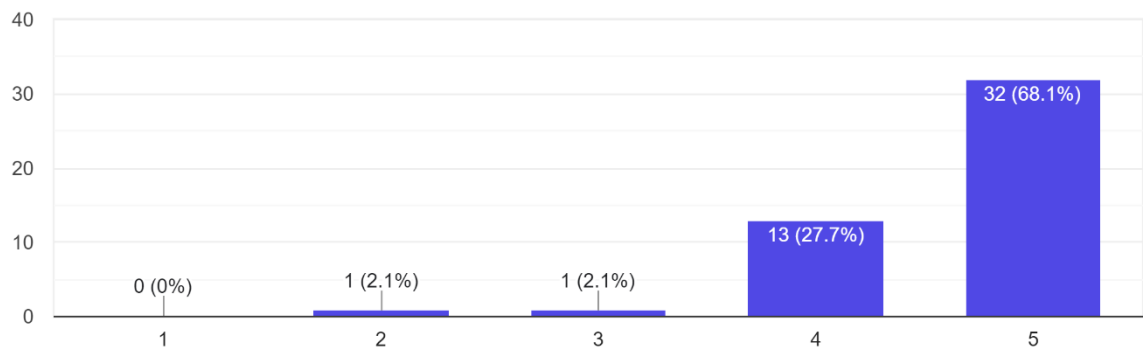


Figura 176. Resultados: ¿Qué dificultad representó para usted el proceso de buscar productos en la aplicación?

¿Qué tan probable es que usted vuelva a utilizar la aplicación?

47 responses

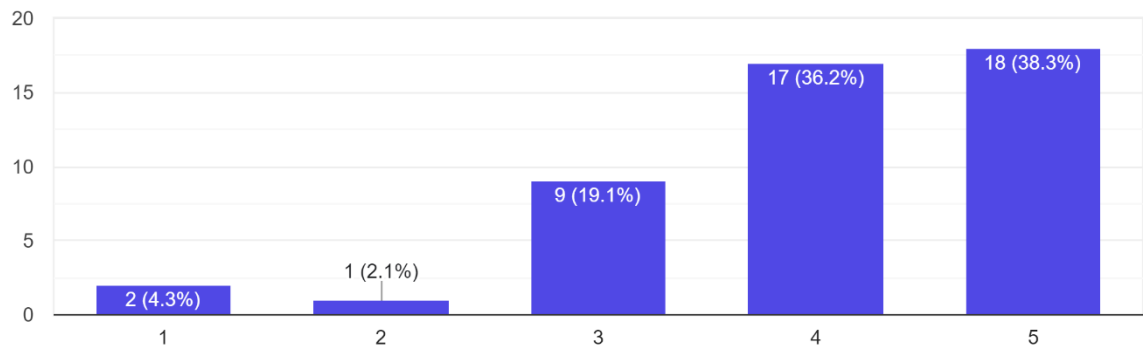


Figura 177. Resultados: ¿Qué tan probable es que usted vuelva a utilizar la aplicación?

¿Qué tan probable es que recomiende la Aplicación a otras personas que necesiten adquirir algún producto o servicio en línea?

47 responses

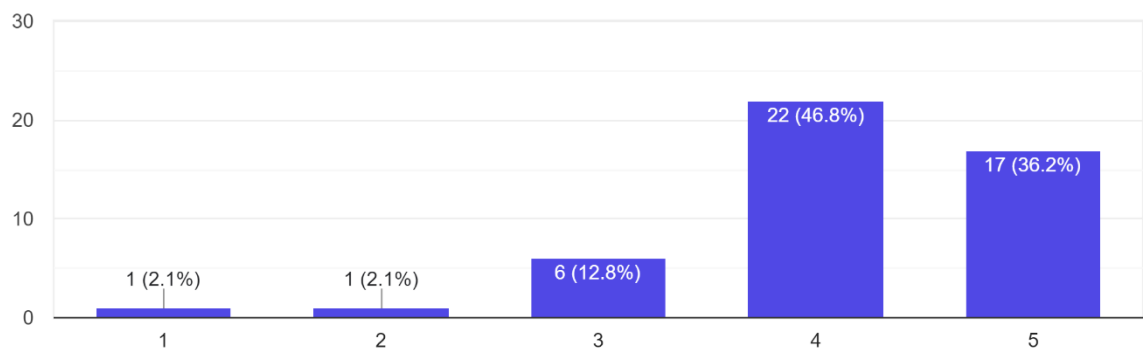


Figura 178. Resultados: ¿Qué tan probable es que recomiende la Aplicación a otras personas que necesiten adquirir algún producto o servicio en línea?

El resumen final de los resultados obtenidos en la Encuesta de Satisfacción se presenta a continuación:

Tabla 50. Resumen de resultados de la Encuesta de Satisfacción

Pregunta	1	2	3	4	5	Promedio de Aceptación
¿Considera que la Aplicación Web es fácil de usar?	0.00%	2.13%	0.00%	19.15%	78.72%	97.87%
¿Considera que la manera de llevar a cabo la adquisición de productos dentro de la Aplicación Web es adecuada?	0.00%	0.00%	6.38%	29.79%	63.83%	100.00%
¿Considera que mediante el uso de la Aplicación puede agilizar la adquisición de productos?	0.00%	0.00%	4.26%	27.66%	68.09%	100.00%
¿Considera útil que la Aplicación se pueda instalar en un dispositivo directamente?	0.00%	4.26%	14.89%	25.53%	55.32%	95.74%
¿Cómo calificaría el proceso de realizar un pedido?	0.00%	4.26%	2.13%	36.17%	57.45%	95.74%
¿Qué dificultad representó para usted el proceso de buscar productos en la aplicación?	0.00%	2.13%	2.13%	27.66%	68.09%	97.87%
¿Qué tan probable es que usted vuelva a utilizar la aplicación?	4.26%	2.13%	19.15%	36.17%	38.30%	93.62%
¿Qué tan probable es que recomiende la Aplicación a otras personas que necesiten adquirir algún producto o servicio en línea?	2.13%	2.13%	12.77%	46.81%	36.17%	95.74%
Promedio Final						97.07%

Anexo 9. Acta de validación de Pruebas de la Aplicación Web Progresiva para la gestión y comercialización de productos y servicios para la empresa Compumars

Desarrollo de una Aplicación Web Progresiva para la gestión y comercialización de productos y servicios para la Empresa “Compumars” de la ciudad de Loja

Acta de validación de Pruebas de la Aplicación Web Progresiva

1. Introducción

En la ciudad de Loja, se procedió a realizar las distintas reuniones que permitieron exponer el uso de la Aplicación Web Progresiva para la gestión y comercialización de productos y servicios para la empresa Compumars, presentada por el estudiante Víctor Andrés Rojas Curimilma al Gerente-Propietario: Ing. Mario Enrique Cueva Hurtado. Tras la exposición, seguidamente se realizó la validación de los requerimientos obtenidos en etapas iniciales de este proyecto, así como la verificación de los principales casos de uso establecidos en la etapa de Diseño de la arquitectura física y lógica desarrollada en el presente TT.

2. Agenda

1.	Ejecución de pruebas de aceptación de los escenarios en la forma automatizada (Solución informática desarrollada)
2.	Ejecución de pruebas de los escenarios en la forma manual (Procedimiento actual llevado por Compumars)
3.	Resolución de firmas de aprobación de pruebas de la Aplicación Web Progresiva para la gestión y comercialización de productos y servicios para la Empresa "Compumars" de la ciudad de Loja

3. Desarrollo

3.1. Prueba de los escenarios en la forma automatizada

En este apartado se describen los casos de prueba de aceptación construidos acorde a los casos de uso más relevantes realizados en la etapa de Diseño de la arquitectura física y lógica del presente proyecto.

Caso de Prueba de Aceptación CU09		Nro: 1
Título	Registrar órdenes de venta	
Pantalla	Órdenes de venta	
Área de prueba	Aplicación Web Progresiva	
Objetivos	El cajero registra una nueva orden de venta	
Acciones a Realizar		
1. El cajero pulsa el botón "Registrar orden de venta" de la página Órdenes de venta		
2. El sistema muestra el formulario de nueva orden de venta		
3. El cajero registra los datos de la orden de venta (datos del cliente, dirección de envío y comprobante de venta, lista de productos, información del pago)		
4. El cajero presiona el botón Guardar del formulario		
Resultado(s) Esperado(s)		
- El sistema valida la información ingresada en el formulario		
- El sistema crea una nueva orden de venta y muestra un mensaje indicando: Orden creada correctamente		

<ul style="list-style-type: none"> - El sistema envía la confirmación de la orden al correo electrónico del cliente - El sistema redirige al cajero a la página "Órdenes de venta" 			
Validado	Si	No	Parcialmente
Observación(es)			
Nombre tester	Mario Enrique Cueva Hurtado		
Tiempo empleado	3 min.		
Referencia al caso de prueba manual	Nro 2. Realizar una nueva venta		

Caso de Prueba de Aceptación CU10			Nro: 2
Título	Completar órdenes		
Pantalla	Ver detalles de una orden		
Área de prueba	Aplicación Web Progresiva		
Objetivos	El operador actualiza una orden al estado de Completado, lo que indica que la orden se ha realizado con éxito		
Acciones a Realizar			
<ol style="list-style-type: none"> 1. El operador presiona el botón Actualizar estado 2. El sistema muestra una ventana emergente solicitando que el operador confirme la operación para completar la orden 3. El operador presiona el botón Confirmar 			
Resultado(s) Esperado(s)			
<ul style="list-style-type: none"> - El sistema establece el estado de la orden en Completada y muestra un mensaje indicando: Orden modificada correctamente - El sistema actualiza el stock de los productos asociados a la orden - El sistema envía la comprobante en formato pdf al correo electrónico del cliente de la orden 			
Validado	Si	No	Parcialmente
Observación(es)			
Nombre tester	Mario Enrique Cueva Hurtado		
Tiempo empleado	15 s.		
Referencia al caso de prueba manual	Nro 3. Registrar y validar pago de una venta		

Caso de Prueba de Aceptación CU12		Nro: 3	
Título	Visualizar pagos de órdenes		
Pantalla	Órdenes de venta		
Área de prueba	Aplicación Web Progresiva		
Objetivos	El operador visualiza el pago de una orden registrada		
Acciones a Realizar			
<ol style="list-style-type: none"> 1. El operador accede a la página "Órdenes de venta" 2. El sistema muestra el listado de todas las órdenes de venta en la tabla de la página Órdenes de venta 3. El operador selecciona una orden de venta de la tabla de la página Órdenes de venta y presiona el botón Ver detalles 			
Resultado(s) Esperado(s)			
<ul style="list-style-type: none"> - El sistema redirige al operador a la página "Ver detalles" de la orden seleccionada - El sistema muestra la información principal de la orden y un apartado para visualizar el pago registrado 			
Validado	Si	No	Parcialmente
Observación(es)			
Nombre tester	Mario Enrique Cueva Hurtado		
Tiempo empleado	10 s.		
Referencia al caso de prueba manual	Nro 3. Registrar y validar pago de una venta		

Caso de Prueba de Aceptación CU13		Nro: 4	
Título	Validar pagos de órdenes		
Pantalla	Ver detalles de una orden		
Área de prueba	Aplicación Web Progresiva		
Objetivos	El operador valida el pago registrado de una orden		
Acciones a Realizar			
<ol style="list-style-type: none"> 1. El operador verifica que el pago registrado por el cliente es correcto 2. El operador actualiza el estado de la orden a Procesando 3. El sistema muestra el formulario de registro del comprobante de venta asociado a la orden 4. El operador registra los datos del comprobante de venta (Tipo de documento, número de comprobante, fecha de emisión, número de serie de los productos de la orden) 5. El operador presiona el botón Guardar del formulario 			
Resultado(s) Esperado(s)			
<ul style="list-style-type: none"> - El sistema valida la información ingresada en el formulario - El sistema establece el estado de la orden en Procesando y muestra un mensaje indicando: Orden modificada correctamente - El sistema crea el comprobante de venta de la orden y muestra un mensaje indicando: Comprobante de venta creado correctamente 			
Validado	Si	No	Parcialmente
Observación(es)			
Nombre tester	Mario Enrique Cueva Hurtado		

Tiempo empleado	3 min.
Referencia al caso de prueba manual	Nro 3. Registrar y validar pago de una venta

Caso de Prueba de Aceptación CU15		Nro: 5	
Título	Registrar productos		
Pantalla	Productos		
Área de prueba	Aplicación Web Progresiva		
Objetivos	El bodeguero registra un nuevo producto		
Acciones a Realizar			
<ol style="list-style-type: none"> 1. El bodeguero pulsa el botón “Registrar producto” de la página Productos 2. El sistema muestra el formulario de nuevo producto 3. El bodeguero registra los datos del producto 4. El bodeguero le asigna una categoría al producto 5. El bodeguero presiona el botón Guardar del formulario 			
Resultado(s) Esperado(s)			
<ul style="list-style-type: none"> - El sistema valida la información ingresada en el formulario - El sistema verifica que el código del producto ingresado no existe - El sistema crea el nuevo producto y muestra un mensaje indicando: Producto creado correctamente - El sistema redirige al bodeguero a la página Productos 			
Validado	Si	No	Parcialmente
Observación(es)			
Nombre tester	Mario Enrique Cueva Hurtado		
Tiempo empleado	2 min.		
Referencia al caso de prueba manual	Nro 1. Registrar productos en el inventario		

Caso de Prueba de Aceptación CU19		Nro: 6	
Título	Registrar órdenes de compra		
Pantalla	Órdenes de compra		
Área de prueba	Aplicación Web Progresiva		
Objetivos	El bodeguero registra una nueva orden de compra		
Acciones a Realizar			
<ol style="list-style-type: none"> 1. El bodeguero pulsa el botón “Registrar orden de compra” de la página Órdenes de compra 2. El sistema muestra el formulario de nueva orden de compra 3. El cajero registra los datos de la orden de compra (datos del proveedor, datos del comprobante de venta, lista de productos) 4. El cajero presiona el botón Guardar del formulario 			
Resultado(s) Esperado(s)			
<ul style="list-style-type: none"> - El sistema valida la información ingresada en el formulario - El sistema crea una nueva orden de compra y muestra un mensaje indicando: Orden creada correctamente - El sistema actualiza el stock de los productos asociados a la orden de compra - El sistema redirige al cajero a la página “Órdenes de compra” 			
Validado	Si	No	Parcialmente

Observación(es)	
Nombre tester	Mario Enrique Cueva Hurtado
Tiempo empleado	4 min.
Referencia al caso de prueba manual	Nro 1. Registrar productos en el inventario

3.2. Pruebas de escenarios de la forma manual

En esta sección se presentan las pruebas de los escenarios ejecutadas en la forma manual, es decir en la forma actual en que Compumars lleva a cabo sus procesos. Estos casos de prueba representan la realidad existente en la empresa previo a la construcción de la solución informática.

Caso de Prueba		Nro: 1
Título	Registrar productos en el inventario	
Área de prueba	Local físico de Compumars	
Objetivos	Registrar una lista de productos entrantes al almacén y actualizar el inventario.	
Acciones Realizadas	Tiempo empleado	Encargado
Iniciar aplicación de escritorio	30 s.	Mario Cueva
Registrar un nuevo producto	2 min. 30 s.	Mario Cueva
Registrar un nuevo proveedor	1 min. 30 s.	Mario Cueva
Buscar producto registrado	30 s.	Mario Cueva
Registrar una nueva entrada del producto en el inventario	2 min.	Mario Cueva
Registrar en una hoja de cálculo los datos del comprobante de venta entregado por el proveedor.	2 min.	Miguel Gonzalez
Tiempo total empleado	9 min.	

Caso de Prueba		Nro: 2
Título	Realizar una nueva venta	
Área de prueba	Local físico de Compumars	
Objetivos	Registrar una nueva venta y actualizar el inventario de los productos salientes	
Acciones Realizadas	Tiempo empleado	Encargado
Iniciar aplicación de escritorio	30 s.	Miguel Gonzalez
Buscar producto	30 s.	Miguel Gonzalez
Registrar los datos del cliente	1 min. 30 s.	Miguel Gonzalez
Registrar una nueva salida del producto en el inventario	2 min.	Miguel Gonzalez
Registrar en una hoja de cálculo una referencia de la venta, el código del producto y su número de serie	2 min.	Miguel Gonzalez
Imprimir el comprobante de venta	2 min.	Miguel Gonzalez
Tiempo total empleado	8 min. 30 s.	

Caso de Prueba		Nro: 3
Título	Registrar y validar pago de una venta	
Área de prueba	Local físico de Compumars	
Objetivos	Registrar el pago de una venta y generar el comprobante de la venta	
Acciones Realizadas	Tiempo empleado	Encargado
Buscar en una hoja de cálculo el registro de la venta por completar y los productos asociados	3 min.	Miguel Gonzalez
Iniciar aplicación de escritorio	30 s.	Mario Cueva
Registrar los datos del cliente	1 min. 30 s.	Mario Cueva
Registrar una nueva salida de los productos en el inventario	5 min.	Mario Cueva
Actualizar en la hoja de cálculo el registro de la venta, el código de los productos y sus números de serie	5 min.	Mario Cueva
Imprimir el comprobante de venta	2 min.	Miguel Gonzalez
Tiempo total empleado	17 min.	

3.3. Resolución de firmas de aprobación de pruebas de la Aplicación Web Progresiva para la gestión y comercialización de productos y servicios para la Empresa “Compumars” de la ciudad de Loja

Tras realizar las reuniones respectivas que permitieron verificar el correcto funcionamiento de la Aplicación Web Progresiva para la gestión y comercialización de productos y servicios para la empresa Compumars de la ciudad de Loja, se procedió a realizar las firmas de aprobación de pruebas con los involucrados en esta actividad.

Gerente - Propietario de Compumars

Ing. Mario Enrique Cueva Hurtado



Firmado digitalmente por
MARIO ENRIQUE
CUEVA HURTADO

Estudiante

Víctor Andrés Rojas Curimilma



Firmado electrónicamente por:
VÍCTOR ANDRES
ROJAS CURIMILMA

Anexo 10. Manual de usuario de la Aplicación Web Progresiva

Desarrollo de una Aplicación Web Progresiva para la gestión y comercialización de productos y servicios para la Empresa “Compumars” de la ciudad de Loja

Manual de usuario de la Aplicación Web Progresiva

1. Introducción

En el presente documento se describen los procesos y pasos a seguir para realizar de manera adecuada las distintas funcionalidades que posee la Aplicación Web Progresiva para la gestión y comercialización de productos y servicios para la empresa Compumars tomando en cuenta las diferentes perspectivas de los roles involucrados en el presente proyecto.

1.1. Objetivo

El objetivo principal de este documento es funcionar como una guía para que los roles involucrados en el presente proyecto puedan realizar de manera apropiada cada uno de los procedimientos que posee la Aplicación Web Progresiva para la gestión y comercialización de productos y servicios para la empresa Compumars de la ciudad de Loja.

1.2. Alcance

El presente manual de usuario está dirigido para todo el personal que forma parte de la empresa Compumars y para sus clientes, con la finalidad de que puedan realizar de manera apropiada cada procedimiento perteneciente a las funcionalidades que posee la presente Aplicación Web Progresiva.

1.3. Requisitos Previos

Los requisitos necesarios son los siguientes:

- Conexión a internet.
- Un navegador web (se recomienda Google Chrome, Apple Safari, Firefox Desktop, Firefox for Android o Microsoft Edge).

2. Funcionalidades

2.1. Funcionalidades de todos los usuarios

Las siguientes funcionalidades están dirigidas para todos los roles del sistema.

2.1.1. Iniciar sesión

Para el inicio de sesión ingresar al enlace */ingresar* en donde aparecerá un formulario como el de a continuación:

← [Página Principal](#)

Ingresar

Formulario de Ingreso al sistema

[Ingresar Ahora](#)

[¿Olvidaste tu contraseña?](#) [¿Aún no posees una cuenta? Regístrate](#)

Figura 179. Formulario de inicio de sesión

2.1.2. Registrar cuenta

Para el registro de una nueva cuenta de usuario, se puede acceder mediante la dirección */registrarse* y aparecerá el siguiente formulario:

← [Página Principal](#)

Registro

Registro de datos de un nuevo usuario

Guarde su contraseña en un lugar seguro

Al registrarse usted acepta los [Terminos y Condiciones](#) propuestos por [Compumars](#)

[Registrarme](#)

[¿Ya tienes una cuenta? Ingresa ahora](#)

Figura 180. Formulario para el registro de nueva cuenta de usuario

Se debe ingresar los datos correctamente y pulsar el botón *Registrarme*. Seguidamente aparecerá una alerta indicando que se ha enviado un correo electrónico para la confirmación de la cuenta de nuevo usuario, tal como se muestra a continuación:

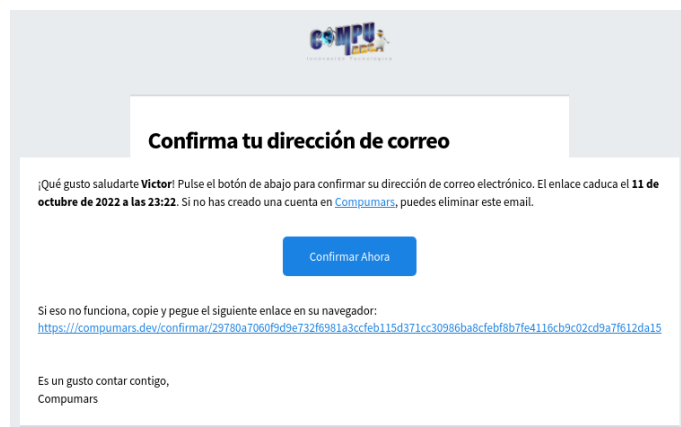


Figura 181. Correo electrónico para la confirmación de una nueva cuenta

2.1.3. Recuperar contraseña

En caso de no recordar la contraseña de su cuenta de usuario, puede acceder a la dirección */clave/recuperar*, en donde aparecerá en formulario para la recuperación de su contraseña

Figura 182. Formulario para la recuperación de contraseña

Al completar el formulario de la Figura 182 y pulsar el botón *Recuperar Contraseña*, se enviará a la dirección de correo electrónico un enlace para el cambio de contraseña, el cual tendrá un tiempo de vencimiento:

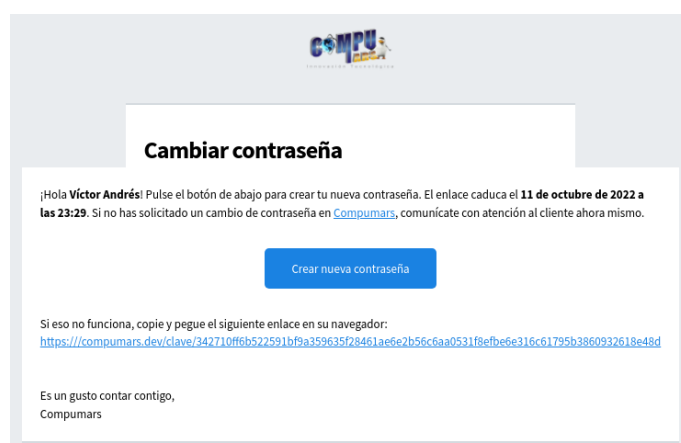


Figura 183. Correo electrónico para el cambio de contraseña

El enlace enviado en el correo electrónico tendrá un enlace que se parecerá al siguiente: */clave/[token]*, al acceder se mostrará un formulario para el cambio de contraseña.



← Iniciar sesión

Cambiar Contraseña

¡Bien, Estás a un paso de recuperar tu contraseña! Recuerda esta vez guardarla en un lugar seguro

Nueva Contraseña *

Guarde su contraseña en un lugar seguro

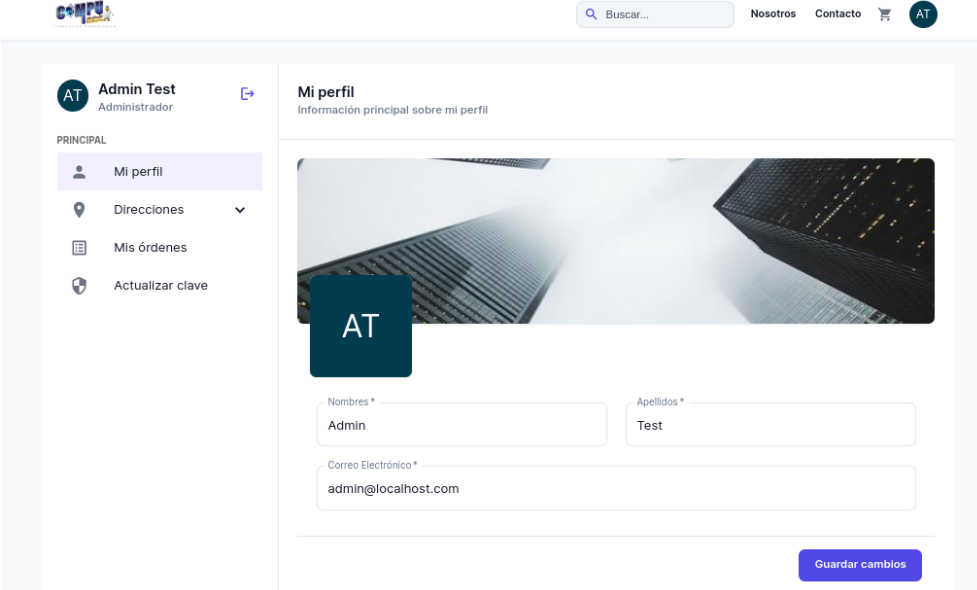
Si todo va bien, te enviaremos a la página de inicio de sesión

Cambiar Contraseña

Figura 184. Formulario para el cambio de contraseña

2.1.4. Administrar perfil

Se puede acceder al perfil del usuario mediante la dirección */perfil*, en donde se podrá actualizar la información general de la cuenta.



COMPUS

Buscar... Nosotros Contacto AT

AT Admin Test Administrador

PRINCIPAL

- Mi perfil
- Direcciones
- Mis órdenes
- Actualizar clave

Mi perfil

Información principal sobre mi perfil

AT

Nombres * Admin

Apellidos * Test

Correo Electrónico * admin@localhost.com

Guardar cambios

Figura 185. Página de perfil de usuario

En la dirección */perfil/direcciones*, aparecerá la información de las direcciones que han sido registradas por el usuario, tal como se muestra a continuación:

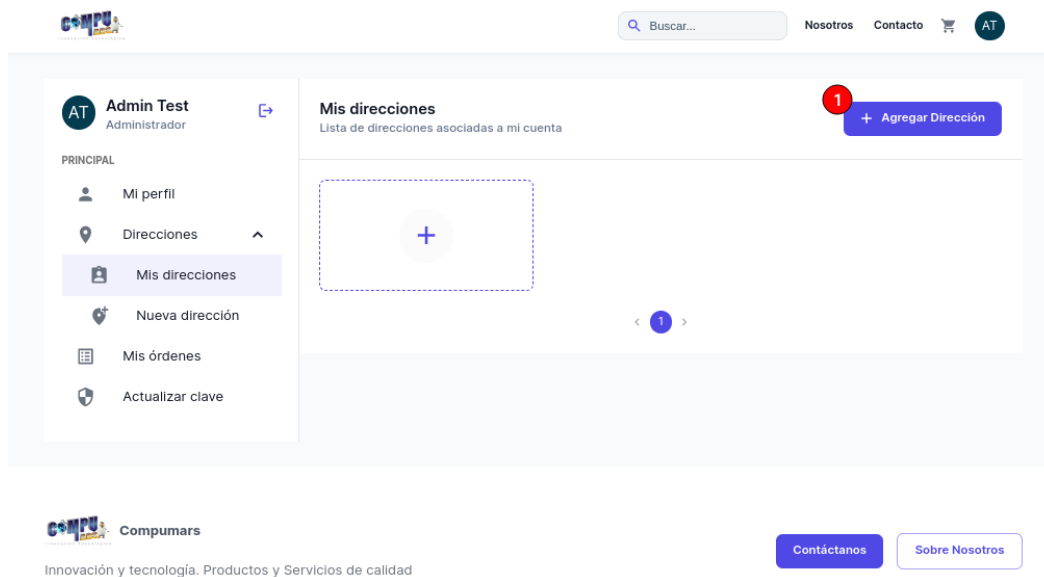


Figura 186. Página de direcciones registradas del usuario

En esta página se tienen los siguientes componentes:

Tabla 51. Componentes de la página de direcciones del usuario

Componente	Descripción
1	Botón Agregar Dirección. Enlace para dirigir al usuario a la página de registro de una nueva dirección (ver Figura 187).

Figura 187. Formulario para el registro de una nueva dirección

Continuando con la administración del perfil de usuario, en la dirección */perfil/mis-ordenes* se puede obtener la lista de ordenes realizadas por el usuario.

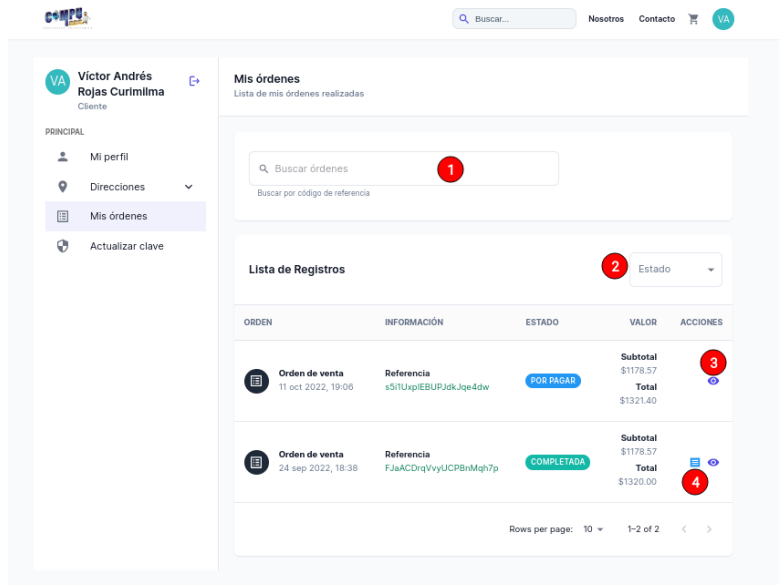


Figura 188. Página de ordenes realizadas por el usuario

En esta página se tienen los siguientes componentes:

Tabla 52. Componentes de la página de ordenes realizadas por el usuario

Componente	Descripción
1	Buscador. Se puede realizar una búsqueda mediante: código de referencia de la orden.
2	Menú desplegable. Permite filtrar los registros mediante el estado de la orden.
3	Botón Ver Detalles. Enlace para dirigir al usuario a la página de ver detalles de la orden (ver Figura 203).
4	Botón Ver Comprobante de Venta. Permite descargar el comprobante de venta asociado a la orden en formato PDF (ver Figura 189).

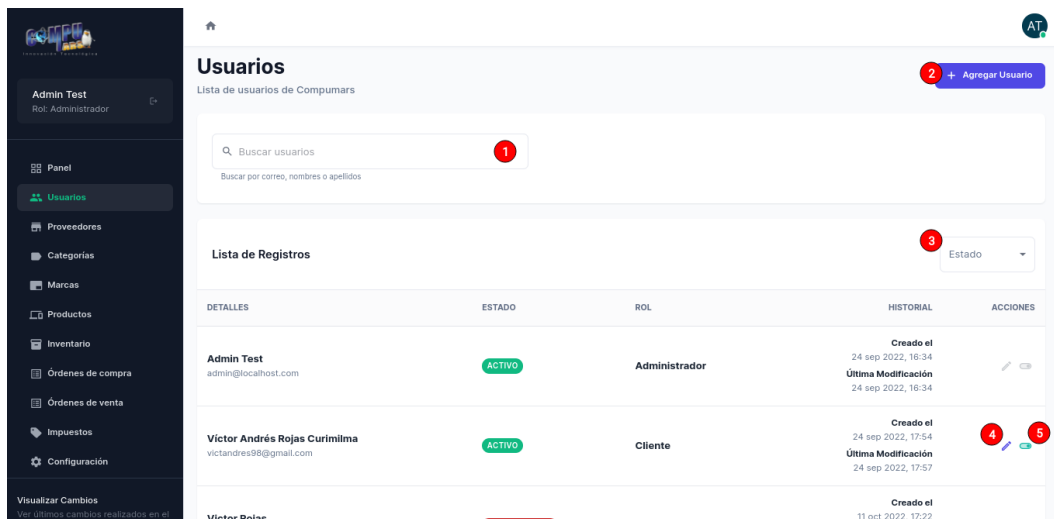


Figura 191. Página de usuarios del sistema

En esta página se tienen los siguientes componentes:

Tabla 53. Componentes de la página de ordenes realizadas por el usuario

Componente	Descripción
1	Buscador. Se puede realizar una búsqueda de usuarios mediante: correo, nombres o apellidos.
2	Botón Agregar Usuario. Enlace para dirigir al usuario a la página de registrar nuevo usuario (ver Figura 194).
3	Menú desplegable. Permite filtrar los registros mediante el estado de la cuenta de usuario.
4	Botón Modificar Usuario. Permite modificar datos de la cuenta de usuario (ver Figura 192).
5	Botón Actualizar estado. Permite habilitar o deshabilitar la cuenta de un usuario (ver Figura 193).

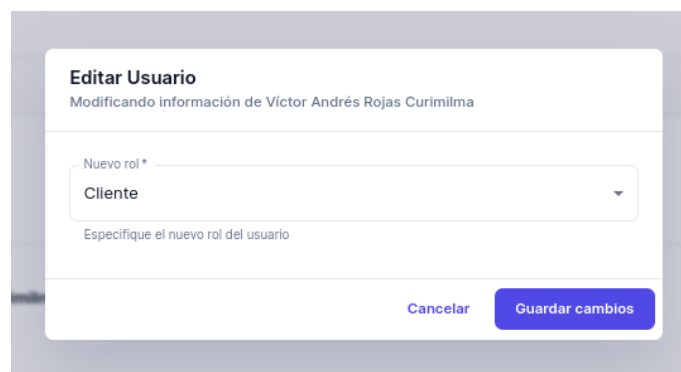


Figura 192. Formulario para modificar usuario

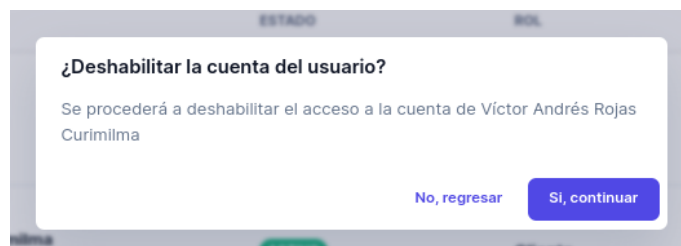


Figura 193. Cuadro de diálogo para habilitar/deshabilitar cuenta de usuario

Para registrar un nuevo usuario, acceder a la dirección </panel/usuarios/nuevo>, en donde se mostrará un formulario como el siguiente:

Figura 194. Página para el registro de un nuevo usuario

2.2.2. Administrar configuraciones del sistema

El administrador del sistema tiene la facultad de visualizar y actualizar las configuraciones generales del sistema, para ello se debe acceder a </panel/configuraciones>.

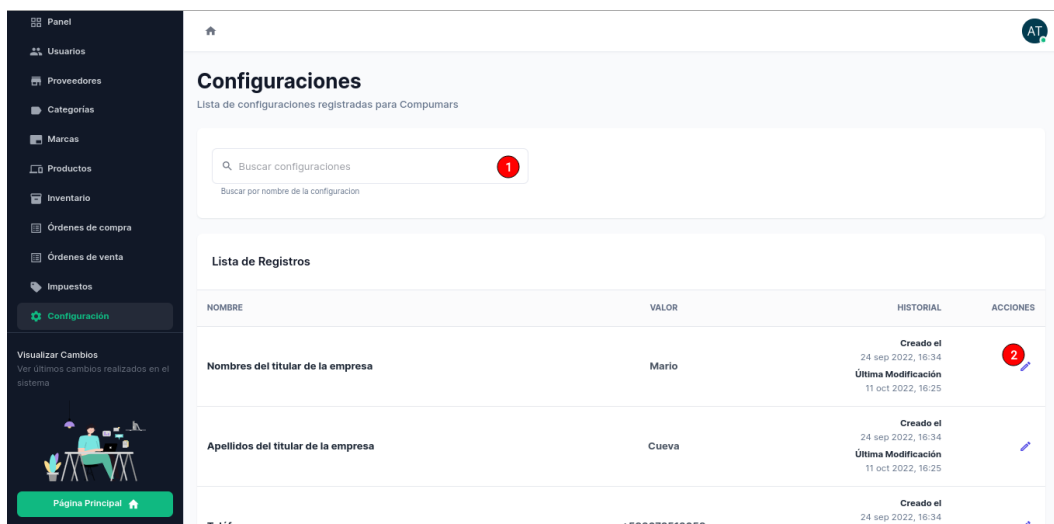


Figura 195. Página de configuraciones generales del sistema

En esta página se tienen los siguientes componentes:

Tabla 54. Componentes de la página de configuraciones generales del sistema

Componente	Descripción
1	Buscador. Se puede realizar una búsqueda de configuraciones mediante: nombre de la configuración.
2	Botón Modificar Configuración. Permite modificar el valor de la configuración. (ver Figura 196).

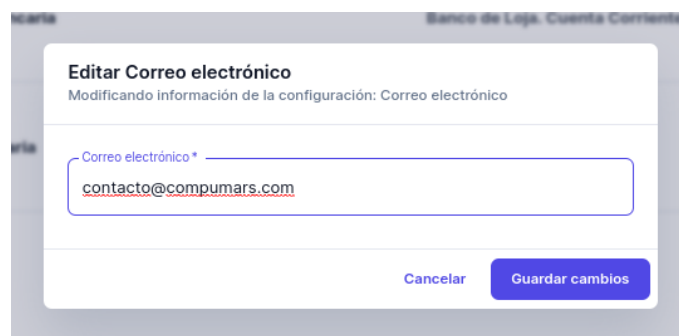


Figura 196. Formulario para modificar el valor de una configuración

2.3. Funcionalidades del rol Cliente

Las siguientes funcionalidades corresponden únicamente al Cliente del sistema:

2.3.1. Página principal

La página principal de la aplicación muestra las principales categorías y los más recientes productos registrados en el sistema, tal como se muestra a continuación.

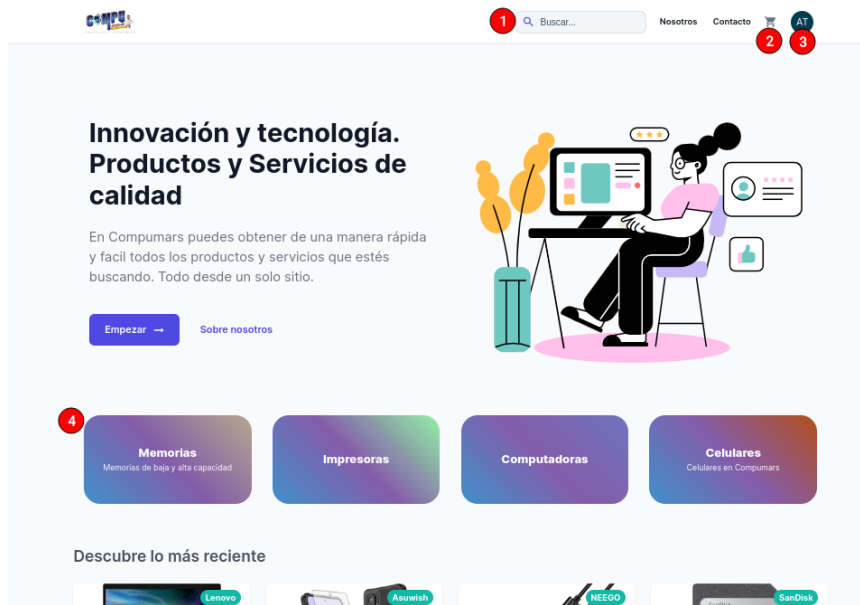


Figura 197. Página principal

En esta página se tienen los siguientes componentes:

Tabla 55. Componentes de la página de ordenes realizadas por el usuario

Componente	Descripción
1	Buscador. Se puede realizar una búsqueda de productos o servicios mediante: nombre del producto o servicio o código único (ver Figura 198).
2	Botón de Ver carrito de compras. Permite abrir un menú lateral que muestra todos los productos que han sido agregados al carrito de compras (ver Figura 201).
3	Menú desplegable. Permite ver información previa del usuario en autenticado, así como acceder a otras páginas del perfil de usuario (ver sección 2.1.4. Administrar perfil).
4	Sección Categorías principales. Permite ver una lista de las categorías principales de la aplicación, se puede pulsar alguna para llevar al usuario a la página de buscar productos por categorías (ver Figura 199).

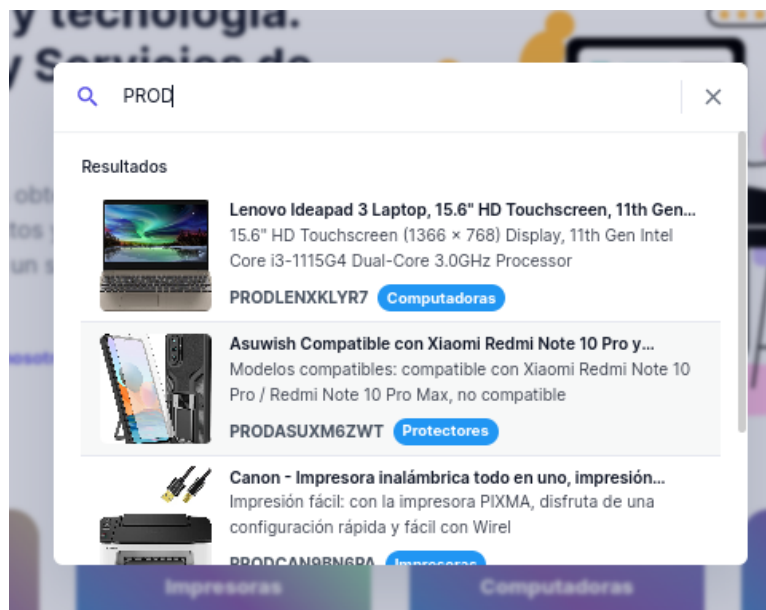


Figura 198. Buscador de productos o servicios

Al pulsar sobre alguna categoría, esta acción llevará al usuario a la dirección */categoria/[nombre de la categoría]*, aquí se mostrarán todos los productos pertenecientes a esa categoría, tal como se puede visualizar a continuación.

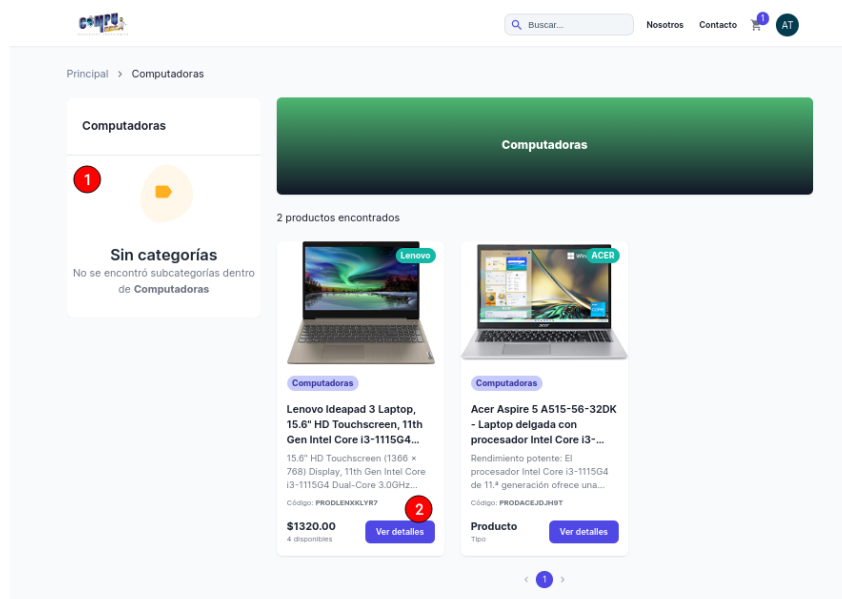


Figura 199. Página de buscar productos por categorías

En esta página se tienen los siguientes componentes:

Tabla 56. Componentes de la página de ordenes realizadas por el usuario

Componente	Descripción
1	Sección subcategorías. Esta sección abarca todas las categorías pertenecientes a la categoría actual. Cada elemento puede ser pulsado para mostrar una página similar al resultado obtenido en la Figura 199.

2	Botón de Ver detalles del producto. Enlace para dirigir al usuario a la página de ver detalles de un producto en específico (ver Figura 200).
---	-----------------------------------------------------------------------------------------------------------------------------------------------

2.3.2. Realizar orden

El cliente puede ver los detalles de cualquier producto o servicio por medio de la dirección */producto/[slug del producto]*. Aquí se mostrarán especificaciones, precios y disponibilidad.

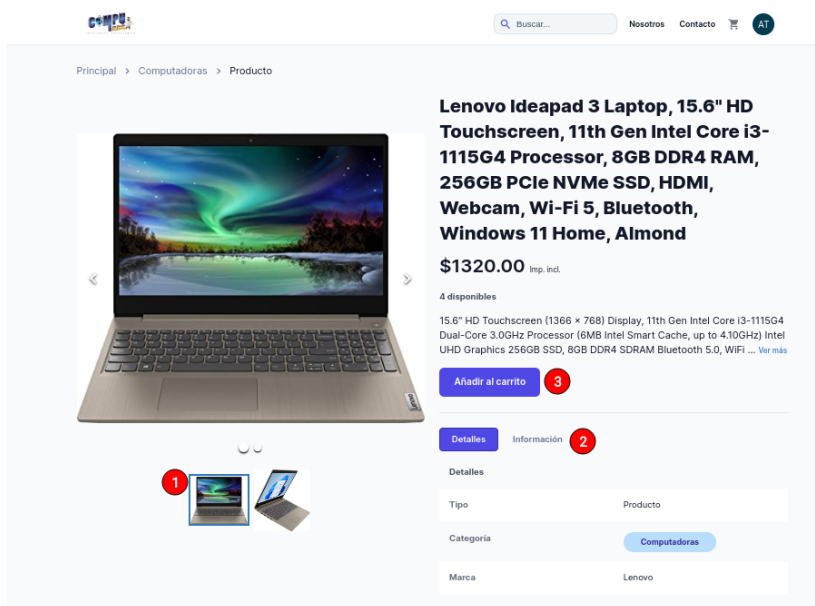


Figura 200. Página para ver detalles de un producto

En esta página se tienen los siguientes componentes:

Tabla 57. Componentes de la página de ver detalles de un producto

Componente	Descripción
1	Sección imágenes del producto. Esta sección contiene una lista seleccionable de todas las imágenes que posee el producto o servicio.
2	Sección de detalles en información. Estas dos pestañas muestran especificaciones del producto (Por ejemplo: Categoría, tamaño, modelo, etc...).
3	Botón Agregar al carrito. En caso de estar disponible, permite añadir el producto al carrito de compras.

Una vez se añada el producto al carrito de compras, este puede ser visto fácilmente desde un menú lateral como el que se presenta a continuación:

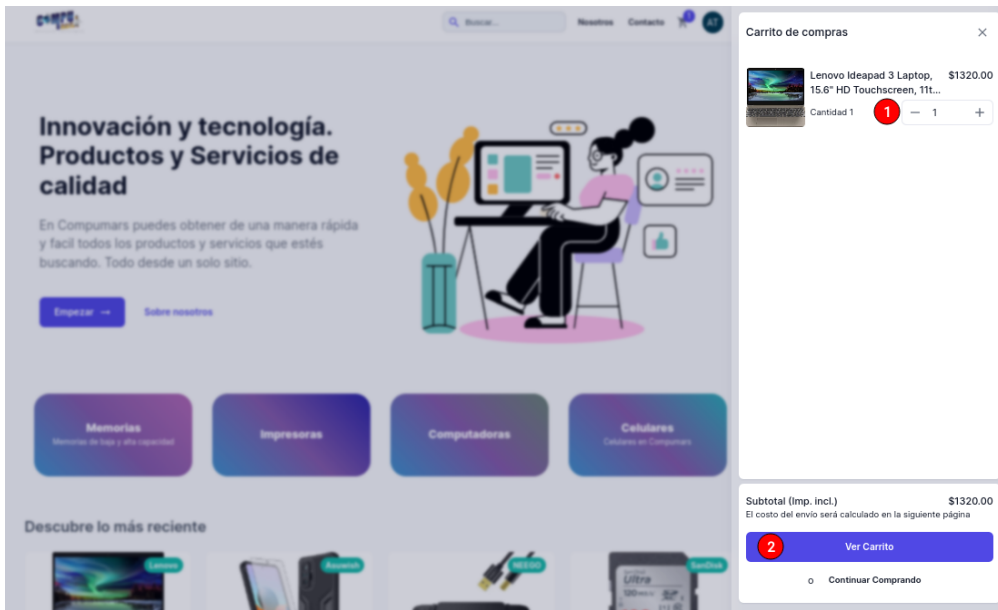


Figura 201. Menú lateral de carrito de compras

En este menú lateral se tienen los siguientes componentes:

Tabla 58. Componentes del menú lateral del carrito de compras

Componente	Descripción
1	Botones para modificar la cantidad de un producto en el carrito de compras.
2	Botón Ver carrito. Enlace para dirigir al usuario a la página de carrito de compras (ver Figura 202).

También se puede acceder a una página mucho más completa del carrito de compras por medio de `/carrito`:

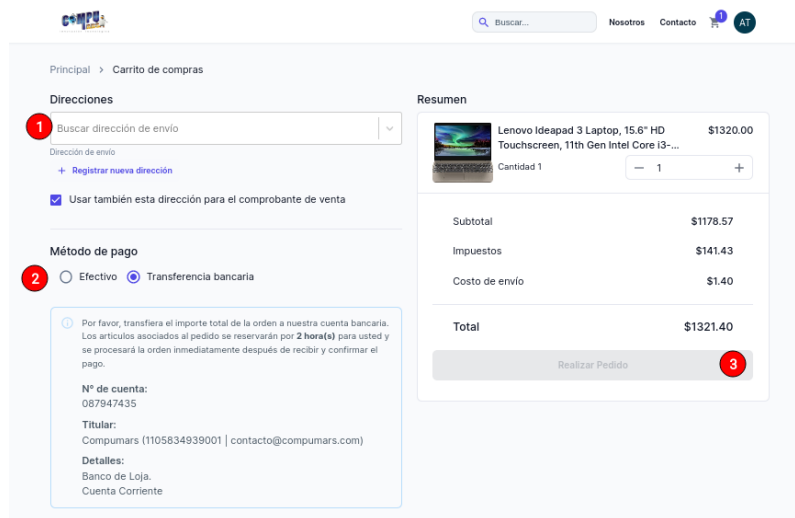


Figura 202. Página de carrito de compras

En esta página se tienen los siguientes componentes:

Tabla 59. Componentes de la página de carrito de compras

Componente	Descripción
1	Sección de direcciones. En esta sección se debe seleccionar una dirección para envío y otra dirección para la generación del comprobante de venta. En caso de no tener una dirección registrada, se puede pulsar el botón <i>Registrar nueva dirección</i> , el cual mostrará un cuadro de diálogo con el formulario de registro de una nueva dirección (ver Figura 187).
2	Sección de método de pago. En esta sección existen dos posibilidades, ambas tienen una alerta con información importante en la parte inferior.
3	Sección de resumen. Aquí se muestra todos los detalles de la orden, tales como precios e información de los productos que han sido añadidos.

Una vez que en la Figura 202 se pulse el botón Realizar Pedido se creará una nueva orden en el sistema y seguidamente se redirigirá al usuario a la página de detalles de la orden (*/perfil/mis-ordenes/[id]*):

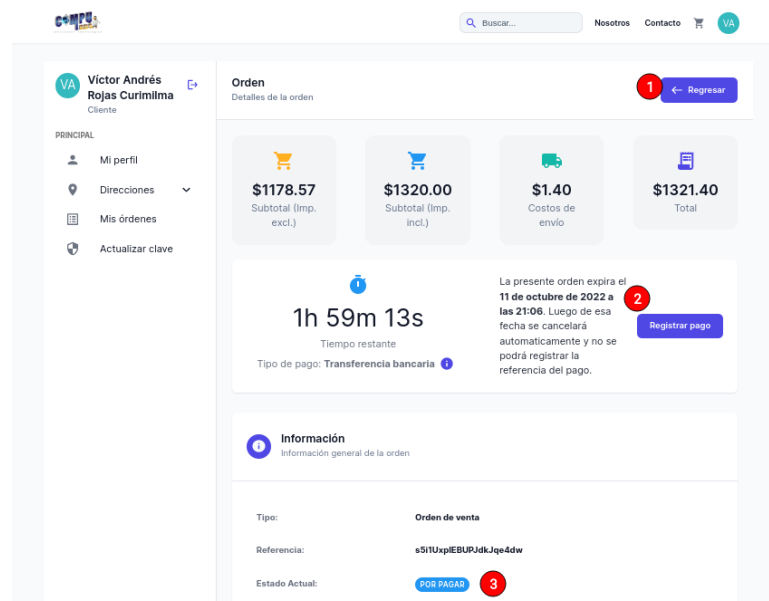


Figura 203. Página de ver detalles de una orden

En esta página se tienen los siguientes componentes:

Tabla 60. Componentes de la página de ver detalles de una orden

Componente	Descripción
1	Botón Regresar. Permite regresar a la página de todas las ordenes realizas por el usuario (ver Figura 188).
2	Sección de registro de la referencia del pago. Esta sección únicamente es visible cuando el cliente tiene la tarea pendiente de realizar el pago por los productos de la orden realizada. Se mostrará un botón <i>Registrar Pago</i> el cual mostrará un formulario para registrar la referencia del pago (ver Figura 204) (este botón está habilitado cuando se seleccionó el método de pago: Transferencia Bancaria). Además, aparece el tiempo disponible que tiene el cliente para realizar el respectivo pago, caso contrario la orden se cancelará automáticamente una vez transcurrido dicho tiempo.
3	Estado de la orden. Se puede visualizar el estado en el cual se encuentra actualmente la orden.

Figura 204. Formulario para el registro de la referencia de pago

2.4. Funcionalidades del rol Cajero

Las siguientes funcionalidades corresponden únicamente al Cajero del sistema:

2.4.1. Ver inventario

El cajero puede ver todos los productos que se encuentran en el inventario de la empresa accediendo a la dirección */panel/inventario*.

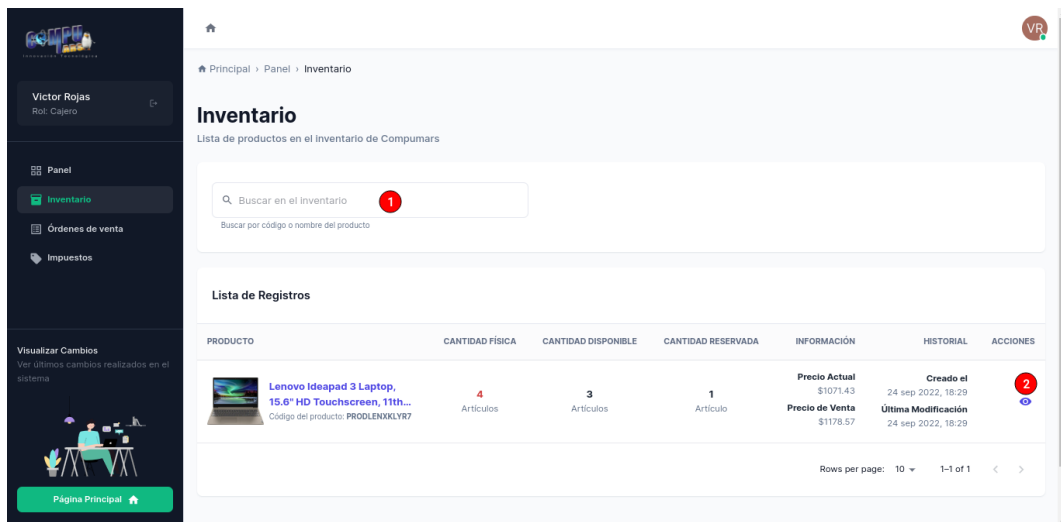


Figura 205. Página de Inventario de productos

En esta página se tienen los siguientes componentes:

Tabla 61. Componentes de la página de inventario de productos

Componente	Descripción
1	Buscador. Se puede realizar una búsqueda de productos en el inventario mediante: código único o nombre del producto.
2	Botón Ver detalles del inventario. Enlace para dirigir al usuario a la página de ver detalles del inventario (ver Figura 206).

Se puede ver los detalles del inventario de un producto en `/panel/inventario/[id]`:

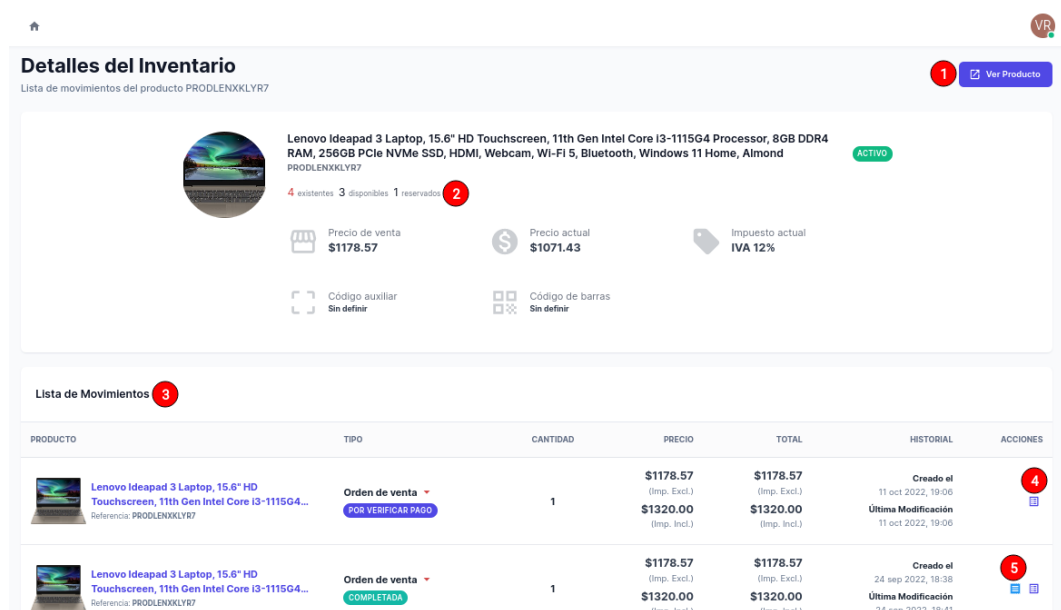


Figura 206. Página de detalles del inventario

En esta página se tienen los siguientes componentes:

Tabla 62. Componentes de la página de detalles del inventario

Componente	Descripción
1	Botón Ver producto. Enlace que en una nueva pestaña lleva al usuario a la página de ver detalles de un producto (ver Figura 200).
2	Sección de disponibilidad del producto. Se muestra la cantidad física, la cantidad reservada en alguna orden y la cantidad disponible o visible al público del producto.
3	Sección de Lista de movimientos. Muestra la lista de ordenes (compra o venta) en las que ha participado el producto.
4	Botón Ver orden. Enlace que lleva al usuario a la página de ver detalles de la orden.
5	Botón Ver Comprobante de Venta. Permite descargar el comprobante de venta asociado a la orden en formato PDF (ver Figura 189).

2.4.2. Administrar órdenes de venta

El cajero puede ver la lista de todas las órdenes de venta realizadas accediendo al enlace */panel/ordenes-venta*, aquí se podrá evidenciar el siguiente resultado:

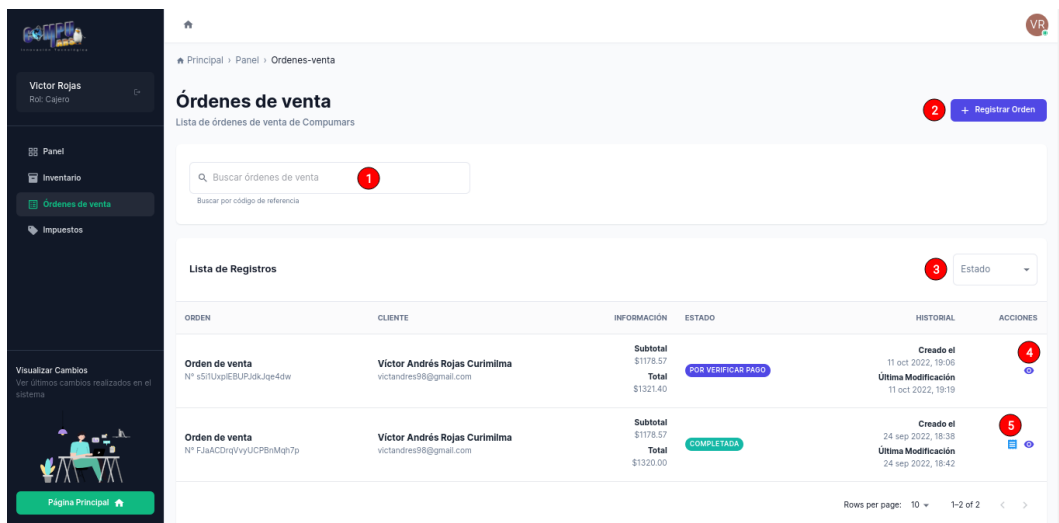


Figura 207. Página de Órdenes de venta

En esta página se tienen los siguientes componentes:

Tabla 63. Componentes de la página de órdenes de venta

Componente	Descripción
1	Buscador. Se puede realizar una búsqueda de órdenes de venta mediante: código de referencia de la orden.

2	Botón Registrar orden. Enlace que lleva al usuario a la página registrar orden de venta (ver Figura 209, Figura 210, Figura 211, Figura 212 y Figura 213).
3	Menú desplegable. Permite filtrar los registros mediante el estado de la orden.
4	Botón Ver orden de venta. Enlace que lleva al usuario a la página de ver detalles de la orden (ver Figura 208).
5	Botón Ver Comprobante de Venta. Permite descargar el comprobante de venta asociado a la orden en formato PDF (ver Figura 189).

Se puede ver el detalle de una orden de venta mediante `/panel/ordenes-venta/[id]`:

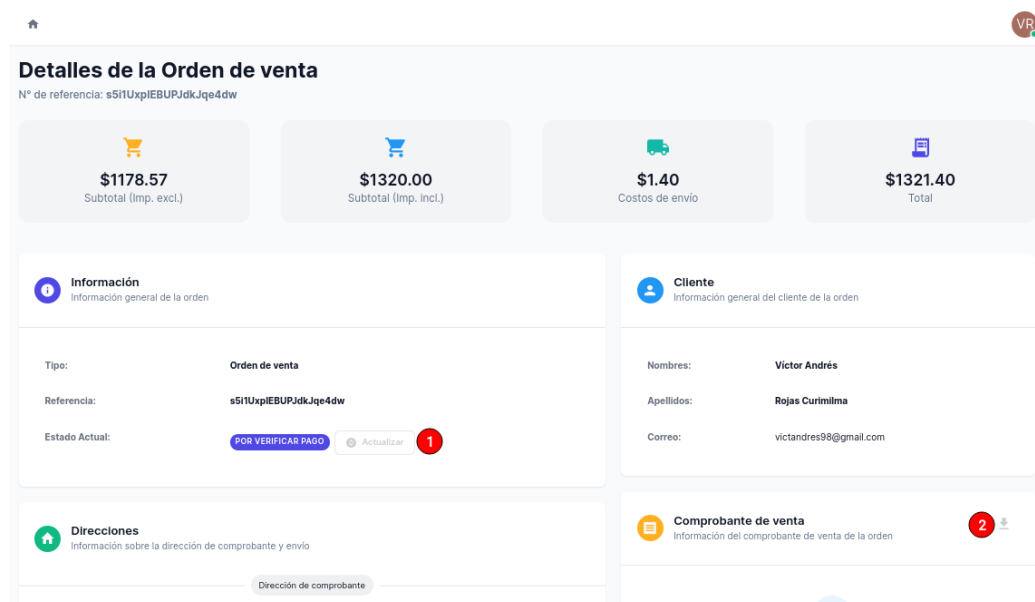


Figura 208. Página de Ver detalles de la orden de venta

En esta página se tienen los siguientes componentes:

Tabla 64. Componentes de la página de órdenes de venta

Componente	Descripción
1	Botón Actualizar estado de la orden. Permite actualizar el estado de la orden hacia un nuevo estado.
2	Botón Descargar comprobante de venta. Permite descargar el comprobante de venta asociado a la orden en formato PDF (ver Figura 189).

Por otro lado, el cajero también puede registrar órdenes de venta para cuando un cliente se encuentra en el local físico de Compumars. Mediante la dirección */panel/ordenes-venta/nueva* accederá a un formulario de 3 pasos, los cuales se describen a continuación:

Figura 209. Formulario de nueva orden de venta - Paso 1

En este primer paso del formulario se tienen los siguientes componentes:

Tabla 65. Componentes del formulario de nueva orden de venta – Paso 1

Componente	Descripción
1	Buscador de productos. Permite buscar entre todos los productos o servicios que se encuentran registrados.
2	Botón Agregar producto. Permite agregar un producto seleccionado desde la búsqueda realizada en el componente 1 a la lista de la orden de venta.
3	Botón Quitar producto. Permite remover el producto en cuestión de la lista de la orden de venta.

Al pulsar el botón *Continuar*, se presentará el siguiente paso del formulario, el mismo que luce tal como se muestra a continuación:

Figura 210. Formulario de nueva orden de venta - Paso 2

En este segundo paso del formulario se tienen los siguientes componentes:

Tabla 66. Componentes del formulario de nueva orden de venta – Paso 2

Componente	Descripción
1	Buscador de clientes. Permite buscar entre todos los clientes que se encuentran registrados.
2	Botón Registrar cliente. Permite mostrar un formulario para el registro de un nuevo cliente.
3	Casilla de verificación. Permite seleccionar un cliente o utilizar el Consumidor Final como cliente. Si esta casilla se encuentra marcada, no será posible seleccionar un cliente desde el Buscador de clientes.

Al pulsar el botón *Continuar*, se mostrará el tercer y último paso del formulario para el registro de una nueva orden de venta. En la Figura 211 y Figura 212 los datos que son requeridos marcados con un asterisco (*). Una vez se complete el formulario y pulsar nuevamente en *Continuar* se mostrará una última vista (ver Figura 213) la cual solicitará una confirmación para la creación de la nueva orden de venta.

Resumen
Últimos detalles de la orden de venta.

\$1155.00 Subtotal (Imp. excl.)	\$1293.60 Subtotal (Imp. incl.)	\$0.00 Costos de envío	\$1293.60 Total
-------------------------------------------	-------------------------------------------	----------------------------------	---------------------------

Estado actual de la orden *
Completada

Especifique el estado en el cual se encuentra la orden

Costo de envío *
\$0

Buscar desde la lista de direcciones de Víctor Andrés

Dirección de envío
[+ Registrar nueva dirección](#)

Buscar desde la lista de direcciones de Víctor Andrés *

Dirección de comprobante de venta *
[+ Registrar nueva dirección](#)

Figura 211. Formulario de nueva orden de venta - Paso 3 (primera parte)

Tipo de documento *
 Factura Nota de venta

Número del documento *
Especifique el número del documento

Números de serie

Fecha de emisión
12/10/2022

Tipo de pago *
 Efectivo Transferencia bancaria

Referencia del pago
Escriba una referencia que identifique el pago

[Regresar](#) [Continuar](#)

Figura 212. Formulario de nueva orden de venta - Paso 3 (segunda parte)

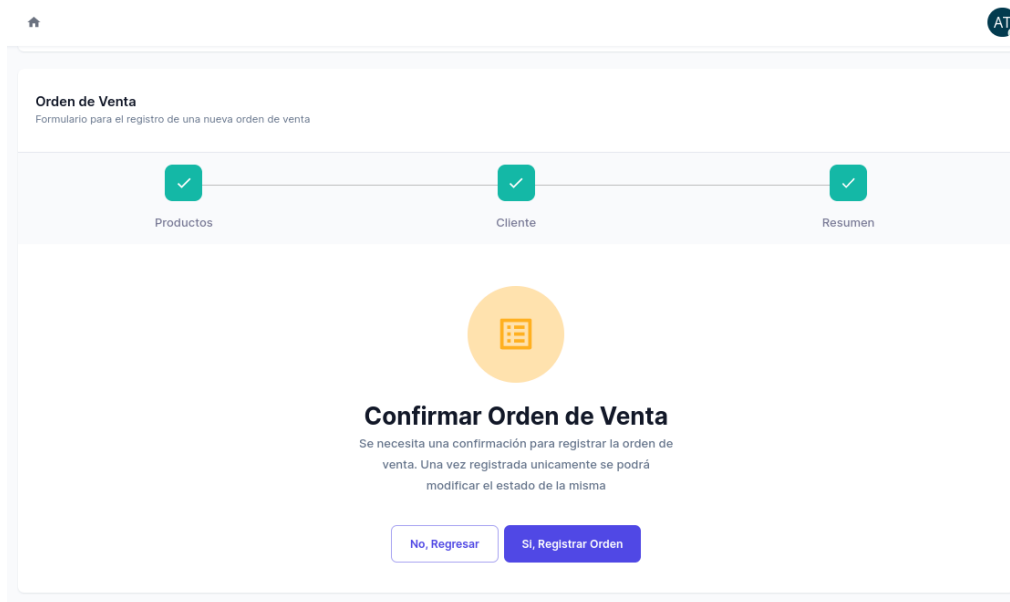


Figura 213. Confirmar registro de la orden de venta

2.4.3. Registrar pago de una orden de venta

El rol cajero también tiene la capacidad de registrar el pago de una orden de venta, siempre y cuando esta orden se encuentre en el estado: *Por Pagar*. Para ello, en la dirección `/panel/ordenes-venta/[id]` se puede ver este estado (ver Figura 214) y el botón Actualizar estará habilitado.

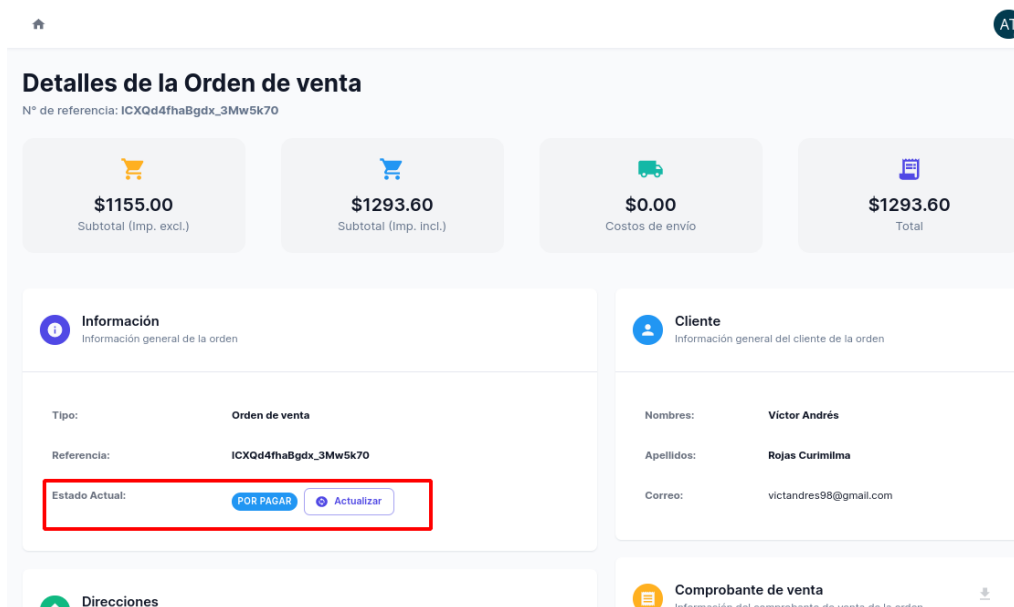


Figura 214. Orden de venta en estado: Por pagar

Luego de pulsar el botón *Actualizar* aparecerá un cuadro de dialogo con el formulario para el registro del pago de la orden, tal como se presenta a enseguida:

Figura 215. Formulario para actualizar el estado de la orden a Por verificar pago

2.5. Funcionalidades del rol Bodeguero

Las siguientes funcionalidades corresponden únicamente al Bodeguero del sistema:

2.5.1. Administrar proveedores

El bodeguero puede ver el registro de todos los proveedores registrados en el sistema accediendo a */panel/proveedores*:

Figura 216. Página de Proveedores

En esta página se tienen los siguientes componentes:

Tabla 67. Componentes de la página de proveedores

Componente	Descripción
1	Buscador. Se puede realizar una búsqueda de proveedores mediante: razón social, correo, nombres o apellidos del proveedor.

2	Botón Agregar proveedor. Enlace que lleva al usuario a la página registrar nuevo proveedor (ver Figura 218). Asimismo, se puede acceder mediante el enlace <i>/panel/proveedores/nuevo</i> .
3	Botón Modificar proveedor. Permite mostrar un cuadro de dialogo con el formulario para modificar la información del proveedor (ver Figura 217).

Figura 217. Formulario para modificar proveedor

Figura 218. Página para el registro de un nuevo proveedor

2.5.2. Administrar categorías

Por otro lado, el bodeguero puede ver el registro de todas las categorías registradas en el sistema accediendo a */panel/categorias*:

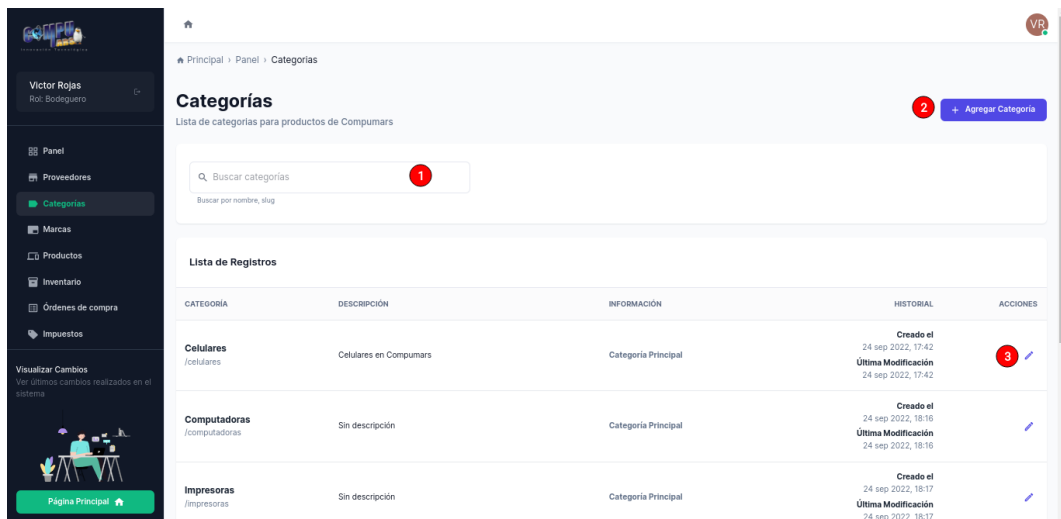


Figura 219. Página de categorías

En esta página se tienen los siguientes componentes:

Tabla 68. Componentes de la página de categorías

Componente	Descripción
1	Buscador. Se puede realizar una búsqueda de categorías mediante: nombre o slug de la categoría.
2	Botón Agregar categoría. Enlace que lleva al usuario a la página registrar una nueva categoría (ver Figura 221). Asimismo, se puede acceder mediante el enlace <i>/panel/categorias/nueva</i> .
3	Botón Modificar categoría. Permite mostrar un cuadro de dialogo con el formulario para modificar la información de la categoría (ver Figura 220).

Figura 220. Formulario para modificar categoría

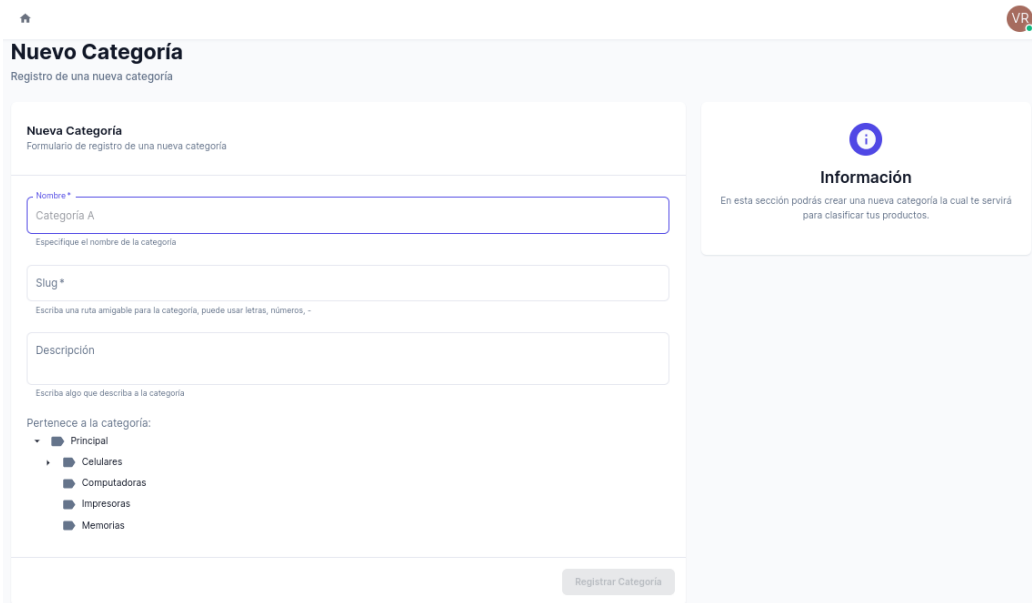


Figura 221. Página para el registro de una nueva categoría

2.5.3. Administrar marcas

Este rol también puede ver el registro de todas las marcas registradas en el sistema accediendo a `/panel/marcas`:

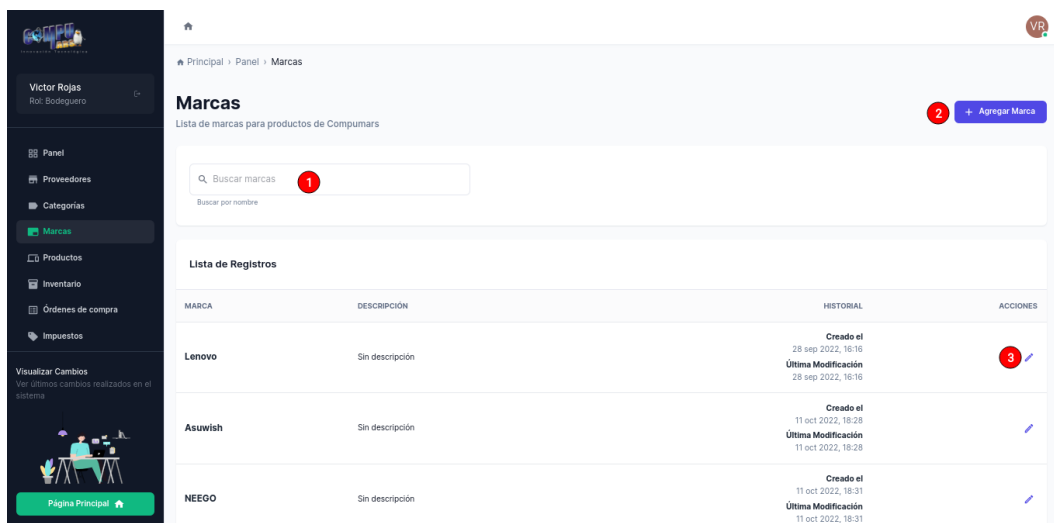


Figura 222. Página de marcas

En esta página se tienen los siguientes componentes:

Tabla 69. Componentes de la página de marcas

Componente	Descripción
1	Buscador. Se puede realizar una búsqueda de marcas mediante: nombre de la marca.

2	Botón Agregar marca. Enlace que lleva al usuario a la página registrar una nueva marca (ver Figura 224). Asimismo, se puede acceder mediante el enlace <i>/panel/marcas/nueva</i> .
3	Botón Modificar marca. Permite mostrar un cuadro de dialogo con el formulario para modificar la información de la marca (ver Figura 223).

Figura 223. Formulario para modificar marca

Figura 224. Página para el registro de una nueva marca

2.5.4. Administrar productos

Una de las facultades más importantes de este rol es el de administrar productos. Para ello, en la dirección */panel/productos*, el bodeguero puede obtener la lista de todos los productos y servicios registrados en el sistema, a continuación, se muestra el resultado obtenido:

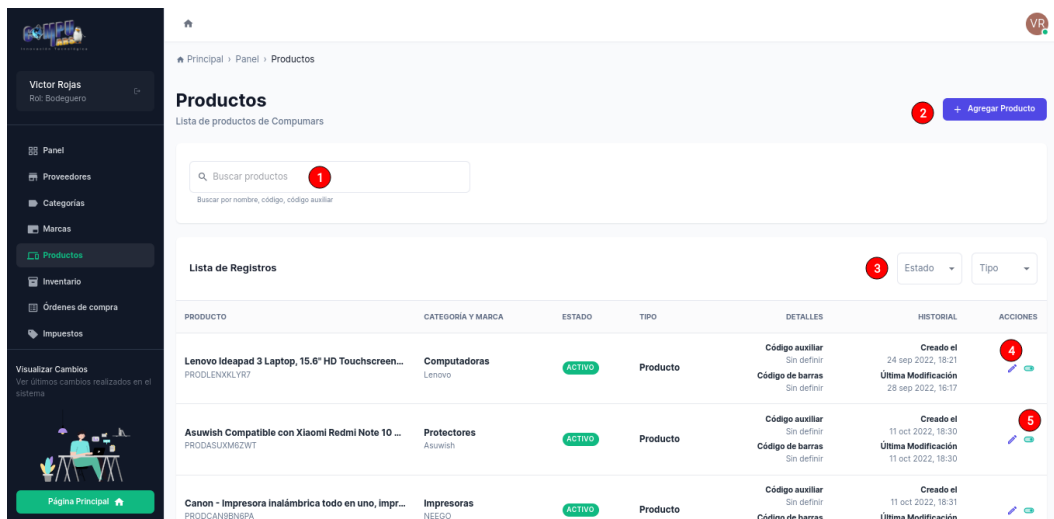


Figura 225. Página de productos

En esta página se tienen los siguientes componentes:

Tabla 70. Componentes de la página de productos

Componente	Descripción
1	Buscador. Se puede realizar una búsqueda de productos mediante: nombre, código o código auxiliar del producto.
2	Botón Agregar producto. Enlace que lleva al usuario a la página registrar un nuevo producto (ver Figura 227). Asimismo, se puede acceder mediante el enlace <i>/panel/productos/nuevo</i> .
3	Menús desplegables. Permiten filtrar el registro de los productos mediante el estado y tipo de productos.
4	Botón Modificar producto. Enlace que lleva al usuario a la página modificar producto (ver Figura 228). Asimismo, se puede acceder mediante el enlace <i>/panel/productos/[id]</i> .
5	Botón Actualizar estado. Permite habilitar o deshabilitar un producto hacia el público (ver Figura 226).



Figura 226. Cuadro de dialogo para habilitar/deshabilitar producto

Figura 227. Página para el registro de un nuevo producto

Es necesario recalcar que se debe completar todos los campos marcados con asterisco (*) para poder realizar el registro de un nuevo producto. Por otro lado, cuando se modifica un producto, a este no se le podrá cambiar el campo de tipo.

Figura 228. Página para modificar un producto

2.5.5. Administrar órdenes de compra

Las órdenes de compra difieren de las órdenes de venta, ya que estas se encargan de actualizar el inventario de Compumars. En la dirección `/panel/ordenes-compra` se puede obtener la lista de las órdenes de compra registradas:

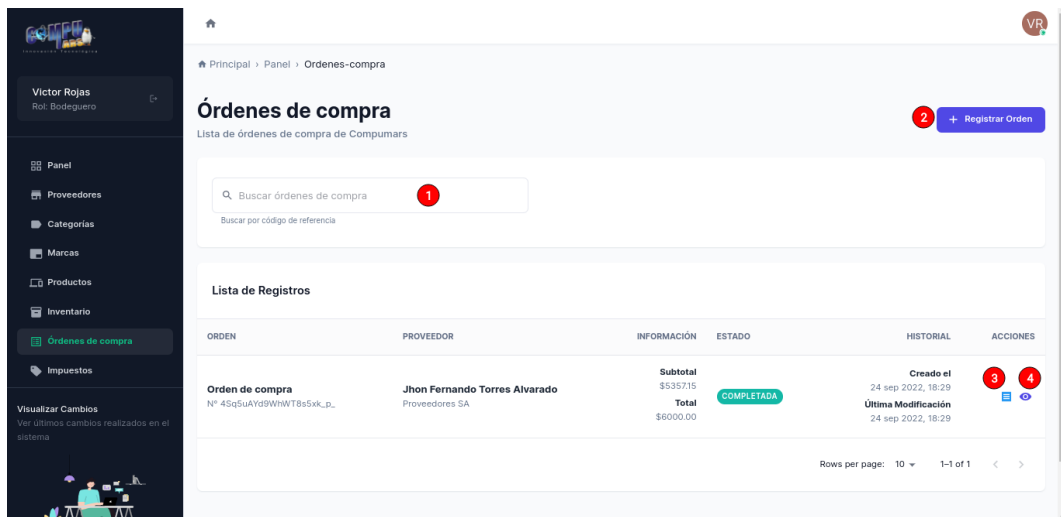


Figura 229. Página de órdenes de compra

En esta página se tienen los siguientes componentes:

Tabla 71. Componentes de la página de órdenes de compra

Componente	Descripción
1	Buscador. Se puede realizar una búsqueda de órdenes de compra mediante: código de referencia de la orden.
2	Botón Registrar orden. Enlace que lleva al usuario a la página registrar orden de compra (ver Figura 230, Figura 231, Figura 233 y Figura 234).
3	Botón Ver Comprobante de Venta. Permite descargar el comprobante de venta asociado a la orden en formato PDF (ver Figura 189).
4	Botón Ver orden de compra. Enlace que lleva al usuario a la página de ver detalles de la orden (ver Figura 208).

El formulario para el registro de una nueva orden de compra consta de tres pasos. A este formulario también se puede acceder mediante */panel/ordenes-compra/nueva*. El primer paso de este formulario solicitará información relacionada al proveedor (ver Figura 230). En la segunda etapa del formulario se encuentra toda la información de los productos que están entrando al almacén (ver Figura 231). El tercer paso posee la información del comprobante de venta y del pago realizado (ver Figura 233). Para culminar, se solicitará una confirmación para realizar el registro de la orden (ver Figura 234). Es importante completar todos los campos marcados con asterisco (*) en el transcurso del formulario.

Información

En esta sección podrás registrar una nueva orden de compra, estas órdenes permiten actualizar el inventario de los productos que selecciones.

Orden de Compra
Formulario para el registro de una nueva orden de compra

Proveedor Productos Comprobante de Venta

Información del proveedor
Seleccione al proveedor de la orden de compra.

Buscar proveedor

Figura 230. Formulario de nueva orden de compra - Paso 1

Información de productos
Seleccione los productos para la orden de compra.

Canon - Impresora inalámbrica todo en uno, impresión inalámbrica de inyección de tinta, escaneo

Buscar por código o nombre del producto

Canon - Impresora inalámbrica todo en uno, impresión inalámbrica de inyección de tinta, escan...
Producto | PRODCAN9BN6PA
ACTIVO

Cantidad *
Especifique la cantidad entrante de este producto (0 en almacén)

Lista de productos 1 agregados

PRODUCTO	CANTIDAD	PRECIO (IMP. EXCL.)	IMPUESTO (IMP.)	PRECIO (IMP. INCL.)	PRECIO DE VENTA (IMP. EXCL.)	TOTAL
Lenovo Ideapad 3 Laptop, 15.6" HD... PRODLENKKLYR7	<input type="text" value="1"/> 4 en almacén	\$0	IVA 12%	\$0	\$1,178.57	Total \$0.00 Total (Imp. Incl.) \$0.00

Figura 231. Formulario de nueva orden de compra - Paso 2

En este segundo paso del formulario se tienen los siguientes componentes:

Tabla 72. Componentes del formulario de nueva orden de compra – Paso 2

Componente	Descripción
1	Buscador de productos. Permite buscar entre todos los productos que se encuentran registrados.
2	Botón Registrar producto. Redirige al usuario a la página de registro de un nuevo producto (Figura 227).
3	Botón Agregar producto. Permite añadir al producto seleccionado en el componente 1 a la lista de productos de la orden de compra.

4	Producto añadido. En cada fila se podrá, por producto, modificar aspectos como: cantidad, precio al que se adquirió el producto (con y sin impuestos incluidos), el impuesto y precio de venta.
5	Botón Modificar precio de venta. Permitirá mostrar un cuadro de diálogo para modificar el precio de venta en base a un rango que representa el porcentaje de ganancia por la venta del producto (ver Figura 232).
6	Botón Quitar producto. Permite remover el producto en cuestión de la lista de la orden de compra.

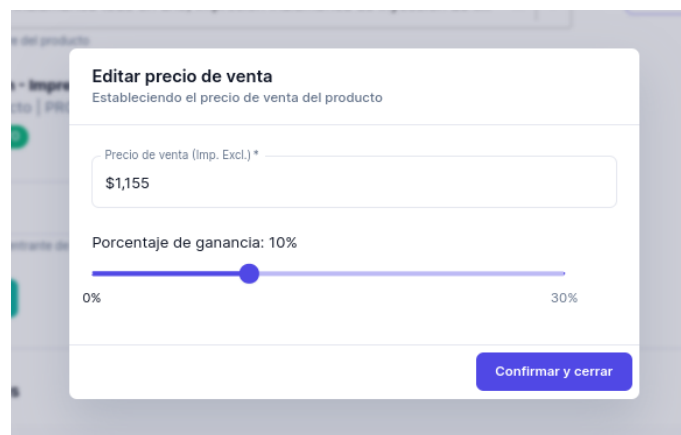


Figura 232. Cuadro de diálogo para establecer el precio de venta

Figura 233. Formulario de nueva orden de compra - Paso 3

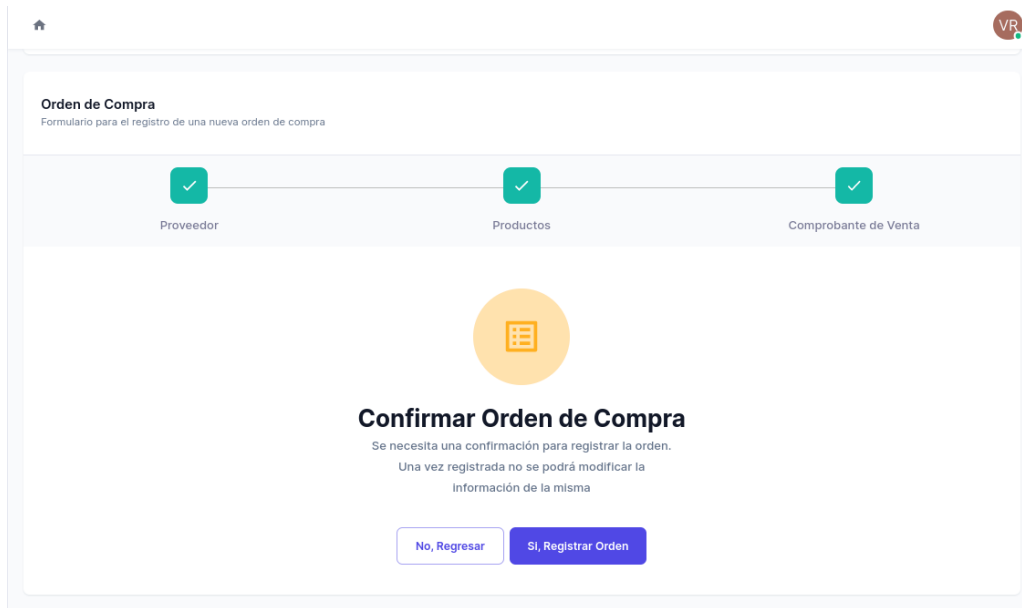


Figura 234. Confirmar registro de la orden de compra

2.5.6. Administrar impuestos

De igual manera, el bodeguero tiene la capacidad de administrar los impuestos que se utilizarán en el sistema. Esto es posible al acceder a `/panel/impuestos`, en donde se presenta lo siguiente:

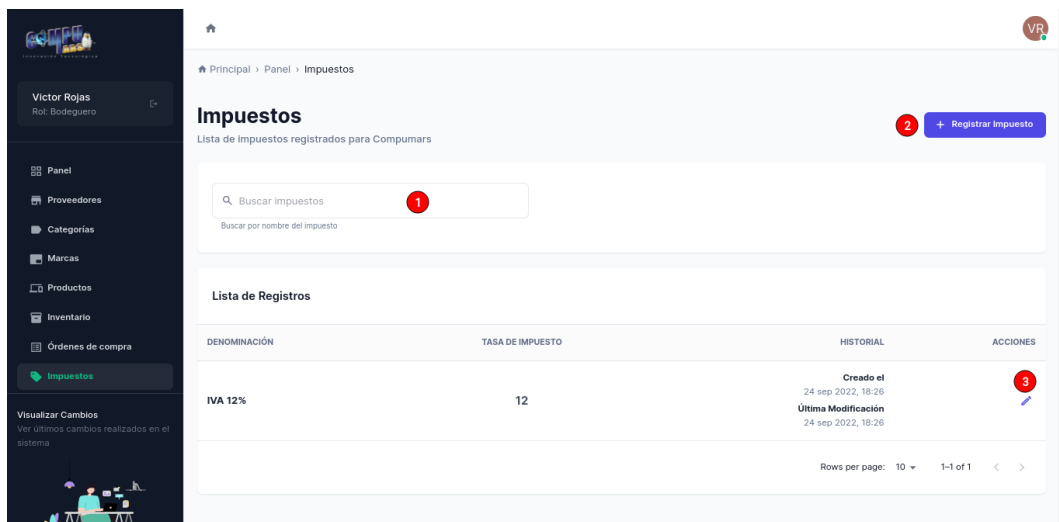


Figura 235. Página de impuestos

En esta página se tienen los siguientes componentes:

Tabla 73. Componentes de la página de impuestos

Componente	Descripción
1	Buscador. Se puede realizar una búsqueda de impuestos mediante: nombre del impuesto.

2	Botón Registrar impuesto. Enlace que lleva al usuario a la página registrar un nuevo impuesto (ver Figura 237). Asimismo, se puede acceder mediante el enlace <i>/panel/impuestos/nuevo</i> .
3	Botón Modificar impuesto. Permite mostrar un cuadro de dialogo con el formulario para modificar la información del impuesto (ver Figura 236).

Figura 236. Formulario para modificar impuesto

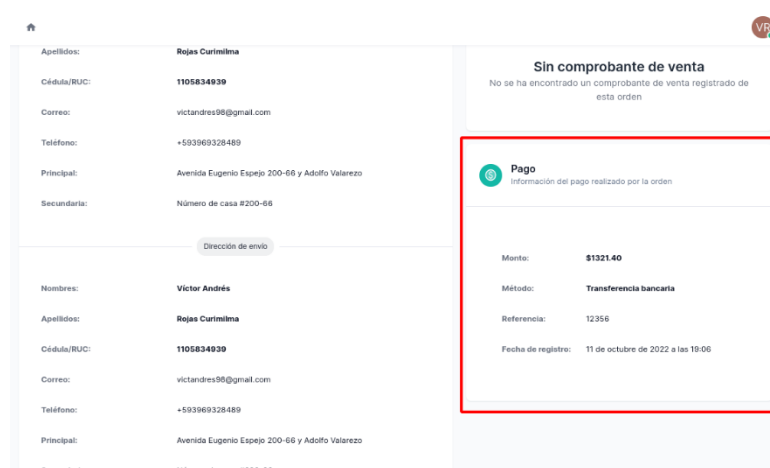
Figura 237. Página para el registro de un nuevo impuesto

2.6. Funcionalidades del rol Operador

Las siguientes funcionalidades corresponden únicamente al Operador del sistema:

2.6.1. Verificar pagos de órdenes de venta

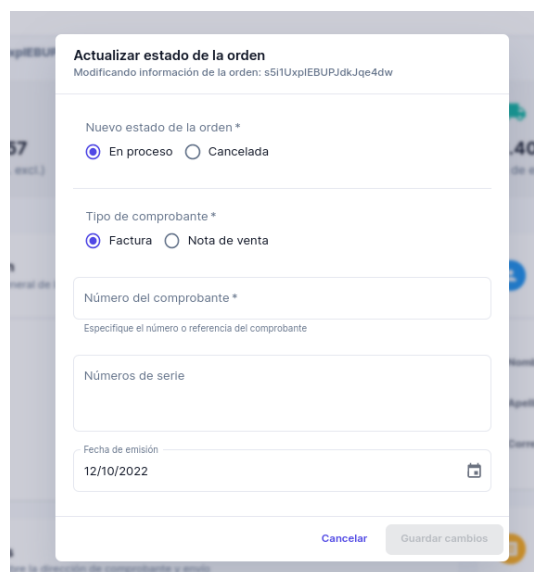
Los clientes son los que realizan las ordenes tal como se lo describió en la sección 2.3.2. Realizar, en otras ocasiones el rol cajero es quien realiza el registro estas órdenes de venta (ver sección 2.4.2. Administrar órdenes de venta). Cuando esta orden se encuentra en el estado: Por verificar pago, es cuando el Operador entra en acción, este rol puede ver el detalle de cada orden de venta accediendo a `orden/panel/ordenes-venta/[id]`, y en la sección pago (ver Figura 238) puede observar toda la información relacionada al pago.



The screenshot shows a user interface for viewing order payment details. On the left, there are two sections for customer information. The top section is for 'Rojas Curmilma' with ID '1105834939' and contact details. The bottom section is for 'Victor Andrés' with ID '1105834930' and similar contact details. On the right, a box titled 'Sin comprobante de venta' (No sales receipt found) contains a 'Pago' (Payment) section. This section is highlighted with a red border and shows the following details: Amount: \$1321.40, Method: Transferencia bancaria, Reference: 12356, and Registration Date: 11 de octubre de 2022 a las 19:06.

Figura 238. Sección de información del pago de la orden

Al constatar que el pago de la orden es válido, puede actualizar el estado de la orden desde la sección Información. Al pulsar el botón Actualizar aparecerá un formulario en donde se le solicitará la información precisa para el comprobante de venta:



The screenshot shows a modal form titled 'Actualizar estado de la orden' (Update order status). The form is for updating the status of order 's5i1UxplEBUPJdkJqe4dw'. It includes the following fields and options: 'Nuevo estado de la orden *' with radio buttons for 'En proceso' (selected) and 'Cancelada'; 'Tipo de comprobante *' with radio buttons for 'Factura' (selected) and 'Nota de venta'; 'Número del comprobante *' with a text input field and a note to specify the number or reference; 'Números de serie' with a text input field; and 'Fecha de emisión' with a date picker set to '12/10/2022'. At the bottom, there are 'Cancelar' and 'Guardar cambios' buttons.

Figura 239. Formulario para actualizar el estado de la orden a En proceso

2.6.2. Marcar órdenes de venta como completadas

En la sección anterior, si se actualizó la orden al nuevo estado de: En proceso, el Operador podrá percatarse que en la en la dirección `/panel/ordenes-venta/[id]` - sección de Información, el botón *Actualizar* estará habilitado (ver Figura 240), al pulsarlo aparecerá el formulario para actualizar el estado de la orden a Completada (ver Figura 241), esto quiere decir que se reducirá el *stock* de los productos que forman parte de esta orden.

The screenshot shows the 'Detalles de la Orden de venta' page. At the top, there are four summary cards: Subtotal (imp. excl.) at \$1178.57, Subtotal (imp. incl.) at \$1320.00, Costos de envío at \$1.40, and Total at \$1321.40. Below these are two main sections: 'Información' and 'Cliente'. The 'Información' section shows the order type as 'Orden de venta', the reference as 's5i1UxpIEBUPJdkJqe4dw', and the current status as 'EN PROCESO' with an 'Actualizar' button. The 'Cliente' section shows the name 'Victor Andrés', surnames 'Rojas Curimilma', and email 'victandres98@gmail.com'. At the bottom, there are sections for 'Direcciones' and 'Comprobante de venta'.

Figura 240. Orden de venta en estado: En proceso

The screenshot shows a modal titled 'Actualizar estado de la orden' for order 's5i1UxpIEBUPJdkJqe4dw'. It asks for the 'Nuevo estado de la orden *' with radio buttons for 'Completada' (selected) and 'Cancelada'. A green notification box states: 'Al completarse la orden se le enviará un correo electrónico a victandres98@gmail.com con el comprobante de pago respectivo.' At the bottom, there are 'Cancelar' and 'Guardar cambios' buttons.

Figura 241. Formulario para actualizar el estado de la orden a Completada

Loja, 01 de marzo de 2023

KARLA CECILIA REINOSO OCHOA, Licenciada en Ciencias de la Educación, Mención Idioma Inglés por la Universidad Técnica Particular de Loja (UTPL), y Magíster en Enseñanza de Inglés como Lengua Extranjera por la Universidad de Alcalá – España.

CERTIFICA. –

Que el resumen de tesis titulada “**Desarrollo de una Aplicación Web Progresiva para la gestión y comercialización de productos y servicio para la Empresa “Compumars” de la ciudad de Loja ”** a cargo del estudiante **Víctor Andrés Rojas Curimilma** con cédula de identidad **1105834939** ha sido debidamente traducido y revisado por mi persona.



Mgs. Karla Cecilia Reinoso Ochoa

C.I. **1104659790**

REGISTRO SENESCYT N°: 1031-14-1284657