



UNIVERSIDAD TÉCNICA DE AMBATO

FACULTAD DE INGENIERÍA EN SISTEMAS, ELECTRÓNICA E INDUSTRIAL

CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES E INFORMÁTICOS

Tema:

SISTEMA PARA EL CONTROL DE INVENTARIO Y DESPACHO DE LISTA DE COMPRAS UTILIZANDO INTEGRACIÓN DE SISTEMAS WEB Y MÓVIL EN EL MINIMARKET “PATY” DE LA CIUDAD DE AMBATO

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

ÁREA: Software

LÍNEA DE INVESTIGACIÓN: Desarrollo de Software

AUTOR: Karen Lissette Aguaysa Carrillo

TUTOR: Ing. Franklin Oswaldo Mayorga Mayorga

Ambato – Ecuador

agosto - 2021

APROBACIÓN DEL TUTOR

En calidad de tutor del Trabajo de Titulación con el tema: SISTEMA PARA EL CONTROL DE INVENTARIO Y DESPACHO DE LISTA DE COMPRAS UTILIZANDO INTEGRACIÓN DE SISTEMAS WEB Y MÓVIL EN EL MINIMARKET “PATY” DE LA CIUDAD DE AMBATO, desarrollado bajo la modalidad Proyecto de Investigación, por la señorita Karen Lissette Aguaysa Carrillo, estudiante de la carrera de Ingeniería en Sistemas Computacionales e Informáticos, de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, de la Universidad Técnica de Ambato, me permito indicar que la estudiante ha sido tutorada durante todo el desarrollo del trabajo hasta su conclusión, de acuerdo a lo dispuesto en el Artículo 15 del Reglamento para obtener el Título de Tercer Nivel, de Grado de la Universidad Técnica de Ambato, y el numeral 7.4 del respectivo instructivo.

Ambato, agosto 2021.

Ing. Franklin Mayorga

TUTOR

AUTORÍA

El presente proyecto de investigación titulado: SISTEMA PARA EL CONTROL DE INVENTARIO Y DESPACHO DE LISTA DE COMPRAS UTILIZANDO INTEGRACIÓN DE SISTEMAS WEB Y MÓVIL EN EL MINIMARKET “PATY” DE LA CIUDAD DE AMBATO es absolutamente original, auténtico y personal. En tal virtud, el contenido, efectos legales y académicos que se desprenden del mismo son de exclusiva responsabilidad del autor.

Ambato, agosto 2021.

Karen Lissette Aguaysa Carrillo

C.C. 1804775912

AUTOR

APROBACIÓN TRIBUNAL DE GRADO

En calidad de par calificador del Informe Final del Trabajo de Titulación presentado por la señorita Karen Lissette Aguaysa Carrillo, estudiante de la Carrera de Ingeniería en Sistemas Computacionales e Informáticos, de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, bajo la Modalidad Proyecto de Investigación, titulado SISTEMA PARA EL CONTROL DE INVENTARIO Y DESPACHO DE LISTA DE COMPRAS UTILIZANDO INTEGRACIÓN DE SISTEMAS WEB Y MÓVIL EN EL MINIMARKET “PATY” DE LA CIUDAD DE AMBATO, nos permitimos informar que el trabajo ha sido revisado y calificado de acuerdo al Artículo 17 del Reglamento para obtener el Título de Tercer Nivel, de Grado de la Universidad Técnica de Ambato, y al numeral 7.6 del respectivo instructivo. Para cuya constancia suscribimos, conjuntamente con la señora Presidenta del Tribunal.

Ambato, agosto 2021.

Ing. Pilar Urrutia, Mg.
PRESIDENTE DEL TRIBUNAL

Ing. Mg. Sandra Carrillo
PROFESOR CALIFICADOR

Dr. Julio Balarezo
PROFESOR CALIFICADOR

DERECHOS DE AUTOR

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

Cedo los derechos de mi Trabajo de Titulación en favor de la Universidad Técnica de Ambato, con fines de difusión pública. Además, autorizo su reproducción total o parcial dentro de las regulaciones de la institución.

Ambato, agosto 2021.

Karen Lissette Aguaysa Carrillo

C.C. 1804775912

AUTOR

DEDICATORIA

El presente proyecto es dedicado a mis padres, Martha y Mesías por ser los pilares fundamentales y mi motor principal para llegar a alcanzar mi sueño profesional.

A mis hermanos Christian y Patricia por ser unos hermanos maravillosos y convertirse en mi ejemplo de esfuerzo y sacrificio.

A mi sobrina Damaris quien me motiva cada día a ser mejor y a conseguir cada uno de los objetivos que me planteo.

A mi enamorado Daniel Alejandro por ser mi guía y apoyo incondicional durante todo este proceso de formación profesional.

A mi abuela Emma quien me demostró a diario que los sueños se cumplen, siempre que los realicemos con amor, paciencia y perseverancia.

Karen Lissette Aguaysa Carrillo

AGRADECIMIENTO

Agradezco en primer lugar a Dios por regalarme salud y guiar mi camino durante todo este tiempo, por darme la sabiduría y fortaleza necesaria para vencer los obstáculos y alcanzar mis objetivos, así también por derramar miles de bendiciones en mi vida.

A mis padres, hermanos y sobrina por brindarme su amor incondicional, comprensión y ayuda en las distintas etapas de mi desarrollo como persona y como profesional.

A mi enamorado Daniel por todo el amor que me brinda a diario y sus palabras de motivación que siempre me inspiran a alcanzar mis metas.

Al Ing. Franklin Mayorga por ser un gran tutor y amigo, por alentarme diariamente a seguir adelante en la carrera, pero sobre todo por ser un constante apoyo y guía en la elaboración de este proyecto.

A mi mejor amigo Bryan Torres por ser un gran amigo y compañero que me alienta a seguir mis ideales y a no decaer en el camino, así también a mis mejores amigas Gabriela Iza y Karen Escobar, fieles compañeras de toda mi vida universitaria.

A la Universidad Técnica de Ambato, en especial a la Facultad de Ingeniería en Sistemas, Electrónica e Industrial por abrirme sus puertas para forjar mi vida profesional y convertirse en mi segundo hogar.

A la Sra. Patricia de Gavilema y al Sr. Enrique Gavilema propietarios del Minimarket “Paty” por brindarme su confianza y permitirme desarrollar mi trabajo de graduación.

A todos aquellos familiares, amigos y compañeros que me acompañaron durante todo este proceso estudiantil y que se vieron de una u otra manera involucrados en la elaboración de esta tesis.

Karen Lissette Aguaysa Carrillo

Índice

APROBACIÓN DEL TUTOR.....	ii
AUTORÍA.....	iii
APROBACIÓN TRIBUNAL DE GRADO	iv
DERECHOS DE AUTOR	v
DEDICATORIA	vi
AGRADECIMIENTO	vii
RESUMEN EJECUTIVO	xvii
ABSTRACT	xviii
CAPÍTULO I.....	1
MARCO TEÓRICO.....	1
1.1 Tema de investigación.....	1
1.2 Antecedentes investigativos	1
1.2.1 Contextualización del problema.....	1
1.2.2 Fundamentación teórica	2
1.2.2.1. Tecnologías de desarrollo integrado web/móvil	4
1.2.2.1.1. Angular	6
1.2.2.1.2. Ionic	6
1.2.2.2. Base de datos SQL	7
1.2.2.2.1. MySQL	7
1.2.2.3. Node.js	7
1.2.2.4. Express.js	8
1.2.2.5. Metodologías para el desarrollo de software	8
1.2.2.5.1. Metodología Extreme Programming	9
1.3 Objetivos	11
1.3.1 Objetivo general	11
1.3.2 Objetivos específicos	11
CAPÍTULO II	12
METODOLOGÍA	12
2.1 Materiales	12
2.2 Métodos	16
2.2.1 Modalidad de investigación	16
2.2.2 Población y muestra	16

2.2.3	Recolección de información.....	17
2.2.4	Procesamiento y análisis de datos	28
CAPÍTULO III.....		30
RESULTADOS Y DISCUSIÓN		30
3.1.	Análisis y discusión de resultados.....	30
3.1.1.	Determinación del framework de desarrollo web – móvil	30
3.1.2.	Determinación de la metodología de desarrollo del proyecto	31
3.2.	Desarrollo de la propuesta	32
3.2.1.	Fase I: Exploración	32
3.2.1.1.	Levantamiento de información	32
3.2.1.2.	Descripción de procesos.....	34
3.2.1.3.	Análisis de resultados.....	38
3.2.1.4.	Definición de roles	39
3.2.1.5.	Arquitectura del sistema.....	40
3.2.2.	Fase II: Planificación	41
3.2.2.1.	Historias de usuario.....	41
3.2.2.2.	Estimación de historias de usuario.....	61
3.2.2.3.	Plan de entrega	62
3.2.3.	Fase III: Iteraciones	64
3.2.4.	Fase IV: Puesta en Producción	66
3.2.4.1.	Iteración 1.....	66
3.2.4.2.	Iteración 2.....	73
3.2.4.3.	Iteración 3.....	76
3.2.4.4.	Iteración 4.....	80
3.2.4.5.	Iteración 5.....	83
3.2.4.6.	Iteración 6.....	86
3.2.5.	Codificación.....	90
3.2.5.1.	Servidor.....	90
3.2.5.2.	Aplicación web.....	95
3.2.5.3.	Aplicación móvil.....	101
3.2.6.	Fase V: Pruebas	103
3.2.6.1.	Pruebas de aceptación	103
3.2.6.2.	Pruebas de rendimiento	120
3.2.7.	Fase IV: Implantación.....	124

3.2.7.1. Instalación de la app en el servidor	124
3.2.7.2. Creación del .apk de la aplicación móvil	128
CAPÍTULO IV	131
CONCLUSIONES Y RECOMENDACIONES	131
4.1. Conclusiones	131
4.2. Recomendaciones	132
BIBLIOGRAFÍA	134

Índice de figuras

Figura 1. Rango de edad de las personas encuestadas.....	22
Figura 2. Frecuencia de compras en el Minimarket "Paty"	22
Figura 3. Tiempo que toma a las personas realizar sus compras	23
Figura 4. Modelo tradicional de gestión de compras	24
Figura 5. Personas encuestadas que realizan compras online.....	24
Figura 6. Personas encuestadas que disponen de un dispositivo móvil.....	25
Figura 7. Personas encuestadas que han realizado compras a través de una aplicación móvil	26
Figura 8. Personas encuestadas que desean una aplicación móvil para el Minimarket "Paty"	26
Figura 9. Personas encuestadas que con una app móvil optimizarán tiempo	27
Figura 10. Personas encuestadas que les gustaría el servicio a domicilio	28
Figura 11. Proceso de Inventario	34
Figura 12. Proceso de Venta	35
Figura 13. Proceso de Compra.....	37
Figura 14. Arquitectura de la aplicación.....	40
Figura 15. Interfaz principal de MySQL.....	67
Figura 16. Interfaz principal de Visual Studio Code	68
Figura 17. Interfaz inicial del proyecto en Angular	69
Figura 18. Modelo relacional de la base de datos	70
Figura 19. Ingreso a la aplicación web	71
Figura 20. Interfaz de la actualización de contraseña	71
Figura 21. Interfaz de la pantalla de inicio.....	72
Figura 22. Interfaz de registro del cliente	72
Figura 23. Interfaz registro de rol	73
Figura 24. Interfaz de visualización de roles	74
Figura 25. Interfaz de edición de roles.....	74
Figura 26. Interfaz de eliminación de roles.....	75
Figura 26. Interfaz de eliminación de roles.....	75
Figura 28. Interfaz de visualización de usuarios.....	76
Figura 29. Interfaz de edición de usuarios	77
Figura 30. Interfaz de eliminación de usuarios	77
Figura 31. Interfaz de registro de pasillo	78

Figura 32. Interfaz de visualización de pasillos	78
Figura 33. Interfaz de edición de pasillos	79
Figura 34. Interfaz de eliminación de pasillos	79
Figura 35. Interfaz de registro de percha	80
Figura 36. Interfaz de visualización de perchas	81
Figura 37. Interfaz de edición de perchas	81
Figura 38. Interfaz de eliminación de perchas	82
Figura 39. Interfaz de registro de producto	82
Figura 40. Interfaz de visualización de productos	83
Figura 41. Interfaz de edición de productos.....	84
Figura 42. Interfaz de eliminación de productos.....	84
Figura 43. Interfaz de registro de venta	85
Figura 44. Interfaz de recepción de listado de compras.....	85
Figura 45. Interfaz de ingreso a la aplicación móvil.....	86
Figura 46. Interfaz de catálogo de productos	87
Figura 47. Interfaz del carrito de productos	87
Figura 48. Interfaz de confirmación de datos	88
Figura 49. Interfaz de confirmación de la compra	88
Figura 50. Interfaz de envío del listado	89
Figura 51. Interfaz de actualización de datos.....	89
Figura 52. Método de renovación de token.....	90
Figura 53. Métodos para la gestión de roles	91
Figura 54. Método para procesamiento de imágenes.....	92
Figura 55. Método de retorno de imagen	92
Figura 56. Método de registro de ventas	93
Figura 57. Método de insertar ventas.....	94
Figura 58. Método de conexión de las interfaces con los servicios	95
Figura 59. Métodos de conexión con la interfaz en Angular	96
Figura 60. Método de registro de productos	97
Figura 61. Método de registro de detalle	98
Figura 62. Método para abrir la ventana modal	99
Figura 63. Método para guardar factura.....	99
Figura 64. Método de creación del socket	100
Figura 65. Método del llamado al socket.....	100

Figura 66. Método de visualización del catálogo de productos	101
Figura 67. Método de envío de mensajes.....	102
Figura 68. Interfaz de DigitalOcean.....	121
Figura 69. Creación del droplet.....	121
Figura 70. Comando ssh en el servidor.....	122
Figura 71. Servidor encendido	122
Figura 72. El servicio corriendo online.....	123
Figura 73. Prueba de rendimiento	123
Figura 74. Ejecución del comando build	124
Figura 75. Carpeta dist.....	125
Figura 76. Instalación de FileZilla	125
Figura 77. Configuración de FileZilla	126
Figura 78. Directorio de la carpeta dist.....	126
Figura 79. Instalación de nginx	127
Figura 80. Ingreso a la carpeta etc	127
Figura 81. Configuración del server	127
Figura 82. Aplicación web en el dominio	128
Figura 83. Ejecución del comando build	128
Figura 84. Resultado de la ejecución del comando build	129
Figura 85. Ejecución del comando capacitor	129
Figura 86. Compilación de Ionic	130
Figura 87. Repositorio del apk.....	130

Índice de tablas

Tabla 1. Comparación de frameworks para aplicación web multiplataforma.....	5
Tabla 2. Comparación de frameworks para aplicación móvil multiplataforma.....	5
Tabla 3. Modelo Entrevista Administrador.....	14
Tabla 4. Entrevista Administrador	18
Tabla 5. Comparación de metodologías ágiles	31
Tabla 6. Definición de Roles.....	40
Tabla 7. Plantilla de Historia de Usuario	41
Tabla 8. Puntos estimados de la Historia de Usuario.....	42
Tabla 9. Historia de Usuario - Estructura del sistema.....	43
Tabla 10. Historia de Usuario - Diseño de la base de datos.....	43
Tabla 11. Historia de Usuario - Ingreso a la aplicación web	44
Tabla 12. Historia de Usuario - Restablecer contraseña	44
Tabla 13. Historia de Usuario - Visualización de la pantalla de inicio.....	45
Tabla 14. Historia de Usuario - Registro de rol	45
Tabla 15. Historia de Usuario - Visualización de Roles	46
Tabla 16. Historia de Usuario - Edición de roles	46
Tabla 17. Historia de Usuario - Eliminación de roles.....	47
Tabla 18. Historia de Usuario - Registro de usuario.....	47
Tabla 19. Historia de Usuario - Visualización de usuario.....	48
Tabla 20. Historia de Usuario - Edición de usuarios	48
Tabla 21. Historia de Usuario - Eliminación de usuarios	49
Tabla 22. Historia de Usuario - Registro de pasillo	49
Tabla 23. Historia de Usuario - Visualización de pasillos	50
Tabla 24. Historia de Usuario - Edición de pasillos.....	50
Tabla 25. Historia de Usuario - Eliminación de pasillos.....	51
Tabla 26. Historia de Usuario - Registro de percha	51
Tabla 27. Historia de Usuario - Visualización de perchas	52
Tabla 28. Historia de Usuario - Edición de perchas.....	52
Tabla 29. Historia de Usuario - Eliminación de perchas.....	53
Tabla 30. Historia de Usuario - Registro de producto.....	53
Tabla 31. Historia de Usuario - Visualización de productos.....	54
Tabla 32. Historia de Usuario - Edición de productos	54

Tabla 33. Historia de Usuario - Eliminación de productos	55
Tabla 34. Historia de Usuario - Registro de venta	55
Tabla 35. Historia de Usuario - Visualización de ventas	56
Tabla 36. Historia de Usuario - Recepción del listado de compras	56
Tabla 37. Historia de Usuario - Ingreso a la aplicación	57
Tabla 38. Historia de Usuario - Registro de cliente	57
Tabla 39. Historia de Usuario - Catálogo de productos	58
Tabla 40. Historia de Usuario - Carrito con productos elegidos.....	58
Tabla 41. Historia de Usuario - Confirmación de datos del comprador	59
Tabla 42. Historia de Usuario - Confirmación de la compra	59
Tabla 43. Historia de Usuario - Envío del listado	60
Tabla 44. Historia de Usuario - Actualización de datos del cliente	60
Tabla 45. Estimación de historias de Usuarios	61
Tabla 46. Plan de entrega	63
Tabla 47. Planificación de iteraciones	64
Tabla 48. Planificación de la iteración 1	66
Tabla 49. Planificación de la iteración 2	73
Tabla 50. Planificación de la iteración 3	76
Tabla 51. Planificación de la iteración 4	80
Tabla 52. Planificación de la iteración 5	83
Tabla 53. Planificación de la iteración 6	86
Tabla 54. Prueba de aceptación 001	103
Tabla 55. Prueba de aceptación 002.....	104
Tabla 56. Prueba de aceptación 003.....	104
Tabla 57. Prueba de aceptación 004.....	105
Tabla 58. Prueba de aceptación 005.....	105
Tabla 59. Prueba de aceptación 006.....	106
Tabla 60. Prueba de aceptación 007	106
Tabla 61. Prueba de aceptación 008.....	107
Tabla 62. Prueba de aceptación 009.....	107
Tabla 63. Prueba de aceptación 010.....	108
Tabla 64. Prueba de aceptación 011	108
Tabla 65. Prueba de aceptación 012.....	109
Tabla 66. Prueba de aceptación 013.....	109

Tabla 67. Prueba de aceptación 014.....	110
Tabla 68. Prueba de aceptación 015.....	110
Tabla 69. Prueba de aceptación 016.....	111
Tabla 70. Prueba de aceptación 017.....	111
Tabla 71. Prueba de aceptación 018.....	112
Tabla 72. Prueba de aceptación 019.....	112
Tabla 73. Prueba de aceptación 020.....	113
Tabla 74. Prueba de aceptación 021.....	113
Tabla 75. Prueba de aceptación 022.....	114
Tabla 76. Prueba de aceptación 023.....	114
Tabla 77. Prueba de aceptación 024.....	115
Tabla 78. Prueba de aceptación 025.....	116
Tabla 79. Prueba de aceptación 026.....	116
Tabla 80. Prueba de aceptación 027.....	117
Tabla 81. Prueba de aceptación 028.....	117
Tabla 82. Prueba de aceptación 029.....	118
Tabla 83. Prueba de aceptación 030.....	118
Tabla 84. Prueba de aceptación 031.....	119
Tabla 85. Prueba de aceptación 032.....	119
Tabla 86. Prueba de aceptación 033.....	120

RESUMEN EJECUTIVO

En la actualidad la compra - venta de productos a través de aplicaciones web y móviles ha tomado un impacto muy grande en la sociedad, puesto que lo se conoce como e-commerce o comercio electrónico ha tenido un gran auge, en especial por la emergencia sanitaria que atraviesa el mundo por el COVID-19, por esta razón ahora se ha vuelto indispensable el manejo de éste tipo de herramientas, para mantener la comunicación entre el cliente y vendedor, asegurándose cada uno de prevenir las aglomeraciones y respetar todos los protocolos de bioseguridad para el bienestar de la sociedad.

Es por lo anteriormente mencionado que el presente proyecto ha sido desarrollado con la finalidad de colaborar con el perfeccionamiento del proceso de compra-venta de productos de primera necesidad en el Minimarket “Paty”, con la finalidad de automatizar ciertos procesos, los mismos que ayudarán a optimizar varios recursos dentro de la administración, el cual permite realizar pedidos a través de una aplicación móvil, manteniendo contacto con una aplicación web que se encargará de la parte administrativa del negocio.

La aplicación web ha sido desarrollada en utilizando el framework de Angular, mientras que la aplicación móvil ha utilizado el framework de Ionic, ambas usando TypeScript como el lenguaje de programación. Los servicios han sido desarrollados en Node.js utilizando el framework Express y con una conexión a la base de datos diseñada en MySQL.

Por otra parte, y una de las más importantes es el puente de comunicación entre las dos aplicaciones en tiempo real que se ha establecido a través de la librería de socket.io, que permite mantener el contacto a través de llamadas http.

Palabras Clave: Angular, Ionic, TypeScript, administración, automatización, comunicación, e-commerce.

ABSTACT

Currently, the purchase - sale of products through web and mobile applications has had a very large impact on society, since it is known as e-commerce or electronic commerce has had a great boom, especially due to the health emergency that crosses the world due to COVID-19, for this reason it has now become essential to use this type of tools, to maintain communication between the customer and the seller, each one making sure to prevent crowds and respect all biosafety protocols for the welfare of society.

It is for the aforementioned that this project has been developed in order to collaborate with the improvement of the process of buying and selling essential products in the Minimarket "Paty", in order to automate certain processes, which will help to optimize various resources within the administration, which allows ordering through a mobile application, maintaining contact with a web application that will be in charge of the administrative part of the business.

The web application has been developed using the Angular framework, while the mobile application has used the Ionic framework, both using TypeScript as the programming language. The services have been developed in Node.Js using the Express framework and with a connection to the database designed in MySQL.

On the other hand, and one of the most important is the communication bridge between the two applications in real time that has been established through the socket.io library, which allows maintaining contact through http calls.

Keywords: Angular, Ionic, TypeScript, administration, automation, communication, e-commerce.

CAPÍTULO I

MARCO TEÓRICO

1.1 Tema de investigación

Sistema para el control de inventario y despacho de lista de compras utilizando integración de sistemas web y móvil en el Minimarket “Paty” de la ciudad de Ambato.

1.2 Antecedentes investigativos

1.2.1 Contextualización del problema

A nivel mundial las aplicaciones móviles y web han tenido un gran auge debido a que vivimos en una época en la cual la tecnología se encuentra revolucionando el mundo, y cada día presenta grandes avances para facilitar varias actividades que el hombre realiza en su vida diaria, en especial la comercialización de productos básicos, necesarios para su subsistencia, a través de dispositivos tecnológicos que tienen conectividad a internet [1].

Por otra parte, en la actualidad la mayoría de las personas desean adquirir sus productos sin tener que aglomerarse en pequeños lugares, y a la vez realizar sus compras sin generar inestabilidad del personal que labora en dicho lugar, manteniendo un ambiente de organización tanto de las personas como de los productos que ofrecen a la venta y de ésta manera los administradores llevar una adecuada gestión contable y de inventario [2].

En la provincia de Tungurahua la mayoría de pequeñas empresas pese a que tienen una gran afluencia de clientes no cuentan con un sistema que les permita llevar un control de sus ventas y a su vez promocionar sus productos a través de una aplicación móvil que permita mejorar el proceso de compra-venta de productos, debido a que consideran un gasto económico innecesario para promocionarse en ciertas aplicaciones populares[3].

En el Minimarket “Paty”, ubicado actualmente en la ciudad de Ambato, demanda de un sistema que permita llevar el control de las ventas que realizan a diario, así como la facilidad de promocionar sus productos a los clientes y realizar ventas a través de una aplicación móvil que puedan disponer en sus dispositivos, la cual es muy necesaria, ya que cuenta con una considerable cantidad de clientes que requieren una mejor y sofisticada atención, así como también los administradores de dicho negocio requieren una mejor organización contable y estructural del mismo [4].

1.2.2 Fundamentación teórica

- **Gestión de la Información**

La función de la Gestión de la Información es vital en el desarrollo y desempeño de la empresa, puesto que ofrece una serie de mecanismos que permite emitir datos exactos para la toma de decisiones.

“En términos perfectamente entendibles sería conseguir la información adecuada, para la persona que lo necesita, en el momento que lo necesita, al mejor precio posible para toma la mejor de las decisiones”[5].

- **Desarrollo Web/móvil**

“Los sitios web móvil son aquellos que están directamente diseñados y desarrollados para la navegación desde un smartphone, pudiendo acceder directamente desde el navegador del teléfono a una versión optimizada de su sitio web, sin necesidad de instalar ninguna aplicación adicional”[6].

En la actualidad la mayoría de negocios se encuentra innovando en la venta de sus productos a través de ésta tecnología, dando paso a lo que se conoce como e-commerce.

- **Integración de Aplicaciones**

“La integración de aplicaciones es un proceso que permite la consolidación y el intercambio de datos, recursos y procesos entre las diferentes organizaciones. El objetivo final es mejorar la eficiencia del negocio.

Las funcionalidades basadas en eventos y en la integración de aplicaciones orientada a servicios de informática abarcan el procesamiento de eventos, la orquestación de servicios y la gestión de procesos”[7].

- **Sistemas Integrados Web/móvil**

Lo importante de los sistemas integrados es el gran valor que le da a una aplicación desarrollada, ya que permite adaptarse a distintos medios tecnológicos de comunicación y permitiendo la interactividad de quienes los usan.

“Los sistemas web móvil puede llegar a lugares donde la web clásica no, lugares que antes eran impensables, trayendo así nuevas opciones a las organizaciones para hacer negocios, contactar con clientes, dar a conocer la marca o producto, trabajar, gestionar tiempo de ocio, así como para otras muchas cosas”[8].

- **Gestión de Calidad**

Se define a gestión de calidad, como:

“Es una serie de procesos sistemáticos que le permiten a cualquier organización planear, ejecutar y controlar las distintas actividades que lleva a cabo. Esto garantiza estabilidad y consistencia en el desempeño para cumplir con las expectativas de los clientes”[9].

- **Logística de Venta de Productos**

“La logística en el e-commerce juega un rol primordial en el proceso de venta online, más aún en las empresas que se embarcan en la internacionalización. La logística es el punto de contacto entre el cliente y la venta realizada”[10].

A través de la logística d venta, se puede asegurar una venta exitosa, en la que todos los actores que forman parte dela misma logran cubrir sus necesidades, de la manera más óptima y garantizando el bienestar común.

- **Control Interno de Inventario**

“El control interno, hace referencia al conjunto de procedimientos de verificación automática que se producen por la coincidencia de los datos reportados por diversos departamentos o centros operativos.

El control operativo aconseja mantener las existencias a un nivel apropiado, tanto en términos cuantitativos como cualitativos”[11].

- **Control de Inventario y Despacho de Productos**

El control de inventario es uno de los procesos más importantes dentro de una empresa, puesto que a través del mismo se puede mantener funcional la misma, ya que es de gran importancia mantener siempre el control de todos los productos que se ofrece al público y de ésta manera evitar ciertos problemas.

“La gestión de inventarios es un punto determinante en el manejo estratégico de toda organización. Las tareas correspondientes a la gestión de un inventario se relacionan con la determinación de los métodos de registro, los puntos de rotación, las formas de clasificación y los modelos de reinventario, determinados por los métodos de control”[12].

1.2.2.1. Tecnologías de desarrollo integrado web/móvil

“El día de hoy ya absolutamente nadie duda de qué forma los dispositivos móviles se han impuesto en muchas áreas respecto a los tradicionales ordenadores personales, sobre todo como medios para navegar.

Es indispensable elegir y usar adecuadamente las distintas tecnologías y opciones existentes en la actualidad pues, en dependencia de cada caso y finalidad. Los sitios móviles son aquellos que están de forma directa diseñados y desarrollados a propósito para la navegación desde un móvil inteligente, pudiendo acceder de manera directa desde el navegador del teléfono.

Las aplicaciones nativas son aquellas aplicaciones que se instalan en el dispositivo mediante los diferentes repositorios de aplicaciones libres para cada sistema operativo

móvil (Apple Store para dispositivos iOS como iPhone o bien iPad; Market para dispositivos Android, entre otros). Su desarrollo es individual y también independiente para cada sistema, siendo preciso desarrollar una aplicación diferente para dispositivos Apple, Android, Windows, entre otros”[13].

Tabla 1. Comparación de frameworks para aplicación web multiplataforma
Elaborado por: Karen Aguaysa

CARACTERÍSTICAS	ANGULAR	REACT	.NET
Lenguaje	Javascript - Typescript	Javascript	C#
Rendimiento	Casi nativo	Casi nativo	Moderado
Interfaz	HTML, CSS	Componentes nativos	Componentes nativos
Reusabilidad	97% de código	90% de código	96% de código
Facilidad de uso	Muy fácil	Medio	Muy fácil

Tabla 2. Comparación de frameworks para aplicación móvil multiplataforma
Elaborado por: Karen Aguaysa

CARACTERÍSTICAS	IONIC	REACTNATIVE	XAMARIN
Lenguaje	HTML, CSS, JS	Javascript (+Java +ObjectiveC)	C#
Rendimiento	Moderado	Casi nativo	Casi nativo
Interfaz	HTML, CSS	Componentes nativos	Componentes nativos
Reusabilidad	98% de código	90% de código	96% de código
Facilidad de uso	Muy fácil	Medio	Medio
Actualizaciones	Mensuales	Cada dos semanas	Mensuales

1.2.2.1.1. Angular

“AngularJS es un potente framework de JavaScript creado para el desarrollo de aplicaciones web dinámicas. Nos permite extender la sintaxis de HTML por medio de atributos propios del framework, para expresar componentes de nuestra aplicación de manera dinámica.

Angular separa completamente el front-end y el back-end en la aplicación, evita escribir código repetitivo y mantiene todo más ordenado gracias a su patrón MVC (Modelo-Vista-Controlador) asegurando los desarrollos con rapidez, a la vez que posibilita modificaciones y actualizaciones.

En una web SPA aunque la velocidad de carga puede resultar un poco lenta la primera vez que se abre, navegar después es totalmente instantáneo, ya que se ha cargado toda la página de golpe”[14].

1.2.2.1.2. Ionic

“Ionic es bastante popular entre los framework de desarrollo de aplicaciones, probablemente el que más. Ionic es básicamente una red front-end que ayuda a construir aplicaciones móviles nativas con HTML, CSS3 y Javascript.

Funciona de manera optimizada en los últimos dispositivos móviles disponibles y proporciona el mejor entorno posible para que la aplicación llegue al mercado de la manera deseada.

La ventaja significativa de Ionic es que se pueden utilizar cientos de elementos de interfaz de usuario predeterminados como formularios, filtros, hojas de acción, vistas de lista, barras de pestañas y menú de navegación en su diseño.

Debido a su capacidad para construir aplicaciones tanto nativas como multiplataforma, Ionic es un gran marco para el desarrollo de las Progressive Web Apps; y permite combinarse junto a los proyectos de PhoneGap, Trigger o Cordova, lo que ofrece una mayor accesibilidad a los gadgets de la arquitectura física de los dispositivos móviles”[15].

1.2.2.2. Base de datos SQL

“Una base de datos de SQL consta de una colección de tablas en las que se almacena un conjunto específico de datos estructurados. Una tabla contiene una colección de filas, también denominadas tuplas o registros, y columnas, también denominadas atributos. Cada columna de la tabla se ha diseñado para almacenar un determinado tipo de información”[16].

1.2.2.2.1. MySQL

“MySQL es un sistema de gestión de base de datos relacional (RDBMS) de código abierto, basado en lenguaje de consulta estructurado (SQL).

MySQL se ejecuta en prácticamente todas las plataformas, incluyendo Linux, UNIX y Windows. A pesar de que se puede utilizar en una amplia gama de aplicaciones, MySQL se asocia más con las aplicaciones basadas en la web y la publicación en línea y es un componente importante de una pila empresarial de código abierto llamado LAMP. LAMP es una plataforma de desarrollo web que utiliza Linux como sistema operativo, Apache como servidor web, MySQL como sistema de gestión de base de datos relacional y PHP como lenguaje de programación orientado a objetos (a veces, Perl o Python se utiliza en lugar de PHP)”[17].

1.2.2.3. Node.js

“Node.js es un entorno de tiempo de ejecución de JavaScript (de ahí su terminación en .js haciendo alusión al lenguaje JavaScript). Este entorno de tiempo de ejecución en tiempo real incluye todo lo que se necesita para ejecutar un programa escrito en JavaScript.”

“Node.js es similar en diseño y está influenciado por sistemas como Event Machine de Ruby y Twisted de Python. Pero Node.js lleva el modelo de eventos un poco más allá. Incluye un bucle de eventos como runtime de ejecución en lugar de una biblioteca. En otros sistemas siempre existe una llamada de bloqueo para iniciar el bucle de eventos. Por lo general, el comportamiento se define mediante devoluciones callbacks de llamada al iniciarse un script y al final se inicia un servidor a través de una llamada de bloqueo”[18].

1.2.2.4. Express.js

“Express es el framework web más popular de Node, y es la librería subyacente para un gran número de otros frameworks web de Node populares. Proporciona mecanismos para:

- Escritura de manejadores de peticiones con diferentes verbos HTTP en diferentes caminos URL (rutas).
- Integración con motores de renderización de "vistas" para generar respuestas mediante la introducción de datos en plantillas.
- Establecer ajustes de aplicaciones web como qué puerto usar para conectar, y la localización de las plantillas que se utilizan para renderizar la respuesta.
- Añadir procesamiento de peticiones "middleware" adicional en cualquier punto dentro de la tubería de manejo de la petición.

A pesar de que Express es en sí mismo bastante minimalista, los desarrolladores han creado paquetes de middleware compatibles para abordar casi cualquier problema de desarrollo web. Hay librerías para trabajar con cookies, sesiones, inicios de sesión de usuario, parámetros URL, datos POST, cabeceras de seguridad y muchos más. Puedes encontrar una lista de paquetes middleware mantenida por el equipo de Express en Express Middleware (junto con una lista de algunos de los paquetes más populares de terceros)”[19].

1.2.2.5. Metodologías para el desarrollo de software

“Actualmente las metodologías de ingeniería de software pueden considerarse como una base necesaria para la ejecución de cualquier proyecto de desarrollo de software que se considere serio, y que necesite sustentarse en algo más que la experiencia y capacidades de sus programadores y equipo. Estas metodologías son necesarias para poder realizar un proyecto profesional, tanto para poder desarrollar efectiva y eficientemente el software, como para que sirvan de documentación y se puedan rendir cuentas de los resultados obtenidos”[20].

1.2.2.5.1. Metodología Extreme Programming

El Extreme (o XP) Programming es una metodología de desarrollo que pertenece a las conocidas como metodologías ágiles, cuyo objetivo es el desarrollo y gestión de proyectos con eficacia, flexibilidad y control.

Ambos conceptos, relacionados estrechamente, son distintos. Agile es el marco de trabajo para el desarrollo del software, se hace mediante un proceso iterativo y define las prácticas y roles del equipo. Por su lado, el XP programming es una metodología basada en la comunicación, la reutilización del código desarrollado y la realimentación.

El equipo de un proyecto XP

Los equipos de un proyecto de esta tipología y magnitud tienen normalmente las siguientes figuras y roles:

- Clientes: Establecen las prioridades y marca el proyecto. Suelen ser los usuarios finales del producto y quiénes marcan las necesidades.
- Programadores: Serán los que se encargarán de desarrollar el Extreme Programming.
- Testers: se encargan de ayudar al cliente sobre los requisitos del producto.
- Coach: Asesoran al resto de componentes del equipo y marcan el rumbo del proyecto.
- Manager: Ofrece recursos, es el responsable de la comunicación externa y quien coordina las actividades.

En general, no obstante, los participantes en este tipo de equipos no siempre toman un rol fijo y contribuyen con los conocimientos de cada uno en aras del beneficio colectivo.

Las planificaciones

Por una parte, se deben planificar los plazos temporales del proyecto basándose en las exigencias del cliente. En base a las estimaciones de coste y la dificultad del proyecto se marcan las prioridades y las fechas, no siempre de forma precisa, pero sí orientativa.

Con la entrega de la planificación efectuada, se desarrolla la de la iteración en el que cada dos semanas se marca el rumbo y se entrega el software útil después de cada uno de estos periodos bisemanales. Con esto se consigue que el nivel de precisión sea mucho mayor, las estimaciones sobre los costes sean más exactas y la información mucho más transparente.

Pruebas

Continuamente se han de efectuar una serie de pruebas automatizadas en base a los requisitos del cliente para comprobar que todo funcione correctamente. Éstas han de hacerse de forma periódica y automática.

Con las planificaciones comentadas anteriormente se incluyen las entregas al final de cada iteración, éstas serán siempre con el software probado y funcionando correctamente y será facilitado al cliente, que puede utilizarlo para cualquier propósito, incluso para el usuario final. Los equipos XP también pueden hacer entregas a otros usuarios finales.

Diseño y programación

El diseño del programa suele ser simple y basado en la funcionalidad del sistema y se lleva a cabo durante todo el proyecto, tanto durante la planificación de la entrega como en el de la iteración.

La programación del software se hace siempre en pareja, lo que se llama programar a dos manos. Se asegura con este método que al menos un programador conoce y controla la labor de otro y queda revisado. La ventaja es que se produce mejor código que en base a un programador, aunque la dificultad de la misma sea mayor.

El código es de todos, con el desarrollo de las pruebas automáticas y la programación a dos manos se incluye también la posibilidad de que cualquiera pueda añadir y retocar parte del código, aunque eso sí, deba ser un estilo común y cuyo resultado sea como si sólo lo hubiera hecho una persona.

Uso de metáforas y otras ventajas

Se buscan frases o nombres que definan parte del programa para que todos sepan a qué se refieren y cuál es su funcionalidad.

El Extreme Programming tiene como gran ventaja el de la programación organizada y planificada para que no haya errores durante todo el proceso. Los programadores suelen estar satisfechos con esta metodología. Es muy recomendable efectuarlo en proyectos a corto plazo.

1.3 Objetivos

1.3.1 Objetivo general

Desarrollar un sistema para el control de inventario y despacho de lista de compras utilizando integración de sistemas web y móvil en el minimarket “Paty” de la ciudad de Ambato.

1.3.2 Objetivos específicos

- Identificar los diferentes procesos contables que se manejan actualmente para el control de inventario y despacho de los productos dentro del minimarket.
- Determinar las herramientas para el desarrollo del sistema, que permitan integrar sistemas web/móvil.
- Diseñar la aplicación web y móvil que permita mejorar los procesos de control de inventario y despacho de lista de compra para satisfacer las necesidades de los usuarios.
- Implantar un sistema web y móvil para el control de inventario y despacho de lista de compras dentro del minimarket “Paty”.

CAPÍTULO II

METODOLOGÍA

2.1 Materiales

Para el presente proyecto, se aplicó un cuestionario a través de una entrevista con los administradores del minimarket “Paty”, con la finalidad de comprender el funcionamiento de ciertos procesos que se realizan para la venta de sus productos.

Así también se realizó una encuesta a varios de sus clientes frecuentes, para consultar datos relevantes para el desarrollo de la aplicación y conocer más a fondo las necesidades de los mismos.

ENCUESTA A CLIENTES DEL MINIMARKET “PATY”

1. ¿En qué rango se encuentra su edad?

- | | |
|----------------|--------------------------|
| 18 – 25 años | <input type="checkbox"/> |
| 26 – 35 años | <input type="checkbox"/> |
| 36 – 45 años | <input type="checkbox"/> |
| 46 – 55 años | <input type="checkbox"/> |
| 56 – 65 años | <input type="checkbox"/> |
| Más de 65 años | <input type="checkbox"/> |

2. ¿Realiza frecuentemente compras en el Minimarket “Paty”?

- Sí No

3. ¿Cuánto tiempo le toma realizar sus compras actualmente en el Minimarket “Paty”?

- | | |
|--------------------------|--------------------------|
| 0:15 – 0:30 minutos | <input type="checkbox"/> |
| 0:30 minutos – 1:00 hora | <input type="checkbox"/> |
| Más de 1:00 hora | <input type="checkbox"/> |

4. ¿Qué le parece el modelo tradicional de gestión de compras en el Minimarket “Paty”?

Excelente

Regular

Malo

5. ¿Consideraría realizar compras online para conservar el distanciamiento social y evitar aglomeraciones de personas?

Sí No

6. ¿Cuenta usted actualmente con un dispositivo móvil en su hogar?

Sí No

7. ¿Ha realizado compras de diferentes productos a través de una aplicación móvil?

Sí No

8. ¿Le gustaría que existiera una aplicación móvil que gestione su lista de compras en el Minimarket “Paty”?

Sí No

9. ¿Cree que al contar con una aplicación móvil que permita enviar la lista de compras y que le notifique cuando esté listo su pedido, optimizaría tiempo en las compras diarias y garantizará su bioseguridad?

Sí No

10. ¿Le gustaría que la entrega de los productos fuera a domicilio?

Sí No

Tabla 3. Modelo Entrevista Administrador
Elaborado por: Karen Aguaysa

ENTREVISTA

PROCESO	ACTOR	PREGUNTA	DESCRIPCIÓN	COMENTARIO	CONCLUSIÓN
Proceso de Venta de productos	Administrador	1. ¿Cómo organiza sus ventas diarias?			
		2. ¿Cómo contabiliza las ganancias de las ventas del día?			
		3. ¿Cuál es el proceso de venta de productos a sus clientes?			
		4. ¿Qué tiempo aproximadamente se demora en atender a un cliente que trae una lista de compras?			
		5. ¿Considera que con ala emergencia sanitaria actual por el covid 19, debería salvaguardar la seguridad de sus clientes, buscando un mecanismo para que no exista aglomeraciones?			
		6. ¿Le gustaría que su negocio utilice nuevas tecnologías para brindar un mejor servicio a sus clientes?			
		7. ¿Le gustaría brindar sus servicio de venta a domicilio?			
		8. ¿Qué método de pago de lo que adquieren sus clientes, le parece el más conveniente?			

		9. ¿Considera que el sistema debería contar con un módulo de facturación?			
Proceso de control de inventario	Administrador	10. ¿Cómo controla el stock de sus productos?			
		11. ¿Cuándo considera que debe abastecerse de más productos?			
		12. ¿Ha perdido clientes por no contar a tiempo con los productos?			
		13. ¿Cómo es el proceso cuando un producto cambia de precio?			
Proceso de organización de productos	Administrador	14. ¿Cómo organiza los productos en su minimarket?			
		15. ¿La manera actual en que organiza los productos le permite encontrar con facilidad los mismos?			

2.2 Métodos

2.2.1 Modalidad de investigación

Investigación de Campo

La investigación ha sido realizada de campo debido a que se ha tomado como fuente de información el mismo objeto de estudio, interactuando directamente con los involucrados, de donde se obtendrán datos reales que aporten a la búsqueda de una solución factible al problema.

Investigación Bibliográfica – Documental

Se ha realizado una investigación bibliográfica – documental ya que se desea obtener información relacionada con problemas similares y de esta manera recopilar documentación valiosa que servirá como sustento científico del proyecto, ampliando el conocimiento a través de los diferentes pensamientos críticos de los autores y visualizando diferentes perspectivas de solución.

2.2.2 Población y muestra

Población: Para la presente investigación la población estará conformada por todo el personal del negocio, tomando en cuenta las personas que laboran en el mismo, así como los clientes que realizan compras diariamente en el minimarket “Paty”, debido a la emergencia sanitaria que se vive en la actualidad y por la cual se desarrollará la aplicación que permitirá que los clientes adquieran sus productos de una manera segura y confiable.

Se tomará a consideración como población al número de clientes que frecuentemente visitan dicho minimarket, que en este caso se encuentran en un rango de 20 a 30 personas.

Muestra: Para la presente investigación se trabajará bajo la técnica de muestreo no probabilístico, donde la elección de los elementos se hará en base a procedimientos al azar debido a que no es posible determinar el grado de representatividad de la población.

Tomando en cuenta la población anterior, la muestra para una población de este tipo con el 95% de confiabilidad y el 5% de margen de error, es de 20 personas.

2.2.3 Recolección de información

Al aplicar la entrevista al administrador del minimarket “Paty” se obtuvo los siguientes resultados.

Tabla 4. Entrevista Administrador
Elaborado por: Karen Aguaysa

PROCESO	ACTOR	PREGUNTA	DESCRIPCIÓN	COMENTARIO	CONCLUSIÓN
Proceso de Venta de productos	Administrador	1. ¿Cómo organiza sus ventas diarias?	Actualmente la organización de las ventas, es manual.		Se concluye que el actual proceso de venta de productos tiene un cierto déficit dentro del mismo ya que no se mantiene una organización de las ventas que realiza diariamente, lo cual no permite optimizar tiempo y puede representar la pérdida de clientes por no mantenerse actualizado en ámbito tecnológico para brindar un mejor servicio.
		2. ¿Cómo contabiliza las ganancias de las ventas del día?	No se contabilizan las ganancias hasta cerrar diariamente el minimarket.	Lo que se encuentra en caja, se asume que es la ganancia del día.	
		3. ¿Cuál es el proceso de venta de productos a sus clientes?	Los clientes llegan a la tienda, formando una fila y piden sus productos, posteriormente se hace la cuenta y ellos proceden a pagar por sus productos.	Actualmente por le emergencia sanitaria por el covid 19, se organizan filas manteniendo el distanciamiento entre los clientes.	
		4. ¿Qué tiempo aproximadamente se demora en atender a un cliente que trae una lista de compras?	Cuando traen una lista considerable, el tiempo en atenderlo es de 20 a 30 minutos aproximadamente.		

		5. ¿Considera que con la emergencia sanitaria actual por el covid 19, debería salvaguardar la seguridad de sus clientes, buscando un mecanismo para que no exista aglomeraciones?	Sí, sería una manera de conservar la fidelidad de cada uno de los clientes, que son la prioridad del negocio.		
		6. ¿Le gustaría que su negocio utilice nuevas tecnologías para brindar un mejor servicio a sus clientes?	Sí, puesto que en la actualidad se cuenta con una variedad de tecnología para comunicarse y el negocio quiere innovar.		
		7. ¿Le gustaría brindar sus servicio de venta a domicilio?	Sí, ya que en la actualidad las personas prefieren recibir sus productos en la puerta de su casa y optimizar tiempo.	En la actualidad debido a la emergencia sanitaria, el negocio ha proporcionado la facilidad de llevar las compras a sus clientes.	
		8. ¿Qué método de pago de lo que adquieren sus clientes, le parece el más conveniente?	El método más conveniente es en efectivo, ya que no se cuenta con otros medios de pago.	Es importante mantener dinero en efectivo puesto que con el mismo se solventan pago a proveedores.	

		9. ¿Considera que el sistema debería contar con un módulo de facturación?	No, ya que actualmente en el negocio son pocas personas quienes piden facturas.	Después de un tiempo se considera implementar un sistema de facturación electrónica.	
Proceso de control de inventario	Administrador	10. ¿Cómo controla el stock de sus productos?	Actualmente se controla a través de la inspección de los productos en las perchas.	No existe ningún mecanismo automatizado para realizar ésta tarea.	En cuanto al proceso de control de inventario, se encuentran varias inconsistencias en dicho proceso ya que por no existir un eficiente control del mismo se ha perdido varias ventas al no contar con la disponibilidad de los productos.
		11. ¿Cuándo considera que debe abastecerse de más productos?	Se considera que se debería abastecer de más productos cuando ya solo exista una media docena del producto.		
		12. ¿Ha perdido clientes por no contar a tiempo con los productos?	Sí, ya que en algunos momentos la inspección es irregular y falla.		
		13. ¿Cómo es el proceso cuando un producto cambia de precio?	Generalmente cuando un producto cambia de precio, los productos restantes toman el precio actual.		

Proceso de organización de productos	Administrador	14. ¿Cómo organiza los productos en su minimarket?	Los productos en el minimarket se encuentran organizados en diferentes secciones del negocio y en perchas.		El proceso de organización actual de los productos dentro del minimarket no permite encontrar con facilidad los productos, lo cual representa el
		15. ¿La manera actual en que organiza los productos le permite encontrar con facilidad los mismos?	Muchas veces, es confuso ya que no se recuerda en los lugares exactos en que se encuentran los productos	Esto sucede con más frecuencia cuando se reorganiza la ubicación de los productos.	incremento de tiempo que los clientes deben permanecer en el minimarket y también lo que genera que exista aglomeración de personas en el mismo.

Al aplicar la encuesta a 20 de los clientes más frecuentes del minimarket “Paty” se obtuvieron las siguientes respuestas, a cada una de las preguntas realizadas en la misma.

Pregunta N° 1: ¿En qué rango se encuentra su edad?

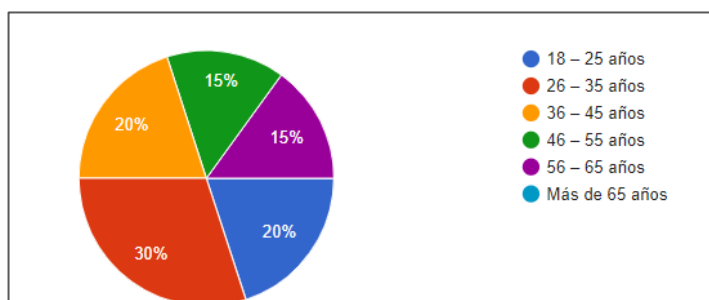


Figura 1. Rango de edad de las personas encuestadas.

Elaborado por: Karen Aguaysa

Fuente: Encuesta aplicada

Análisis e interpretación de resultados:

De acuerdo con los resultados representados en la Figura 1, es posible observar que el 20% de las personas encuestadas, se encuentran en un rango de edad entre los 18 a 25 años, el 30% de 26 a 35 años, el 20% de 36 a 45 años, el 15% de 46 a 55 años, el 15% de 55 a 65 años y que no existen personas mayor a 65 años de edad, lo que demuestra que la mayoría de personas que visitan recurrentemente el minimarket “Paty” son jóvenes.

Pregunta N° 2: ¿Realiza frecuentemente compras en el Minimarket “Paty”?

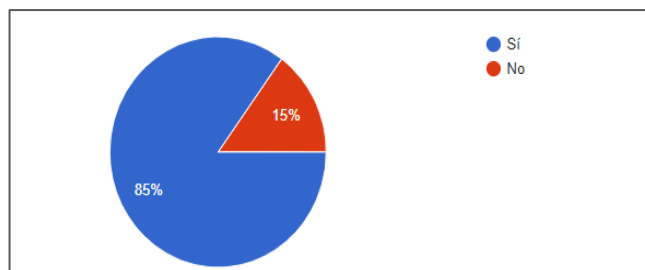


Figura 2. Frecuencia de compras en el Minimarket "Paty"

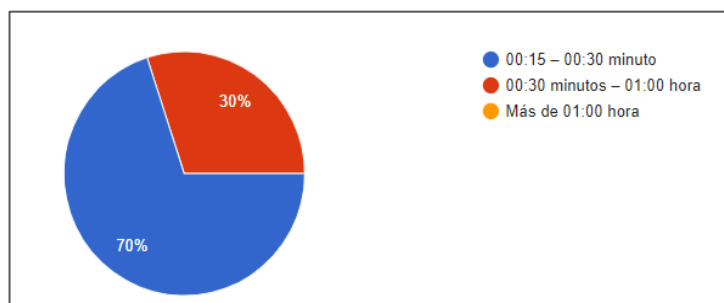
Elaborado por: Karen Aguaysa

Fuente: Encuesta aplicada

Análisis e interpretación de resultados:

De acuerdo con los resultados representados en la Figura 2, es posible observar que el 85% de las personas encuestadas, realizan frecuentemente compras en el minimarket y el 15% no son clientes muy frecuentes, lo que demuestra que la mayoría de personas que conforman el vecindario, consideran siempre realizar compras en el minimarket “Paty”.

Pregunta N° 3: ¿Cuánto tiempo le toma realizar sus compras actualmente en el Minimarket “Paty”?



*Figura 3. Tiempo que toma a las personas realizar sus compras
Elaborado por: Karen Aguaysa
Fuente: Encuesta aplicada*

Análisis e interpretación de resultados:

De acuerdo con los resultados representados en la Figura 3, es posible observar que el 70% de las personas encuestadas que realizan sus compras en el minimarket “Paty”, se toman de 00:15 a 00:30 minutos en realizar sus compras diarias, mientras que el otro 30% se toman entre 00:30 minutos a 1:00 hora en realizar sus compras, por lo que se puede identificar que la mayoría de clientes dedican un tiempo considerable a diario para realizar todas sus compras.

Pregunta N° 4: ¿Qué le parece el modelo tradicional de gestión de compras en el Minimarket “Paty”?

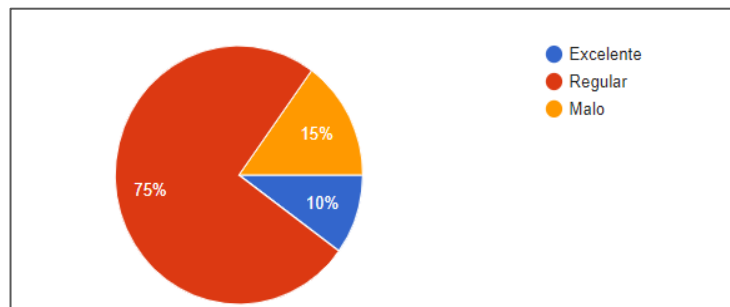


Figura 4. Modelo tradicional de gestión de compras
Elaborado por: Karen Aguaysa
Fuente: Encuesta aplicada

Análisis e interpretación de resultados:

De acuerdo con los resultados representados en la Figura 4, es posible observar que el 75% de las personas encuestadas cree que el modelo tradicional de gestión de compras dentro del minimarket “Paty” es regular, el 15% cree que es malo y el 10% cree que es excelente, por lo que se puede identificar que algunos de los procesos que se realizan durante la compra de los productos, no satisface, ni cumple con las expectativas del cliente.

Pregunta N° 5: ¿Consideraría realizar compras online para conservar el distanciamiento social y evitar aglomeraciones de personas?

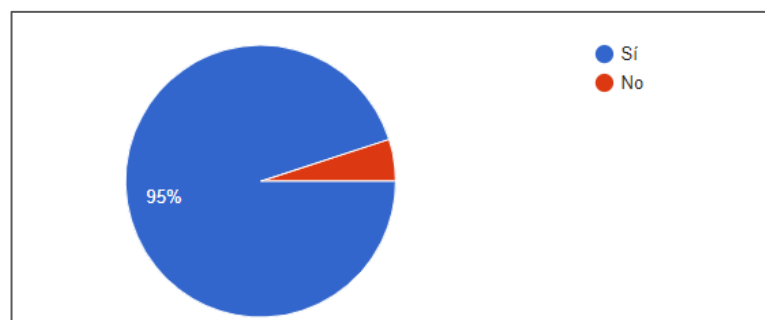


Figura 5. Personas encuestadas que realizan compras online
Elaborado por: Karen Aguaysa
Fuente: Encuesta aplicada

Análisis e interpretación de resultados:

De acuerdo con los resultados representados en la Figura 5, es posible observar que el 95% de las personas encuestadas, considerarían realizar compras en línea para conservar el distanciamiento social y evitar las aglomeraciones, el 5% restante, no considera tan importante el realizar compras en línea, lo que demuestra que la mayoría de personas estarían dispuestas a innovar en la tecnología para poder preservar su salud por la pandemia del covid 19 que actualmente atraviesa el país.

Pregunta N° 6: ¿Cuenta usted actualmente con un dispositivo móvil en su hogar?

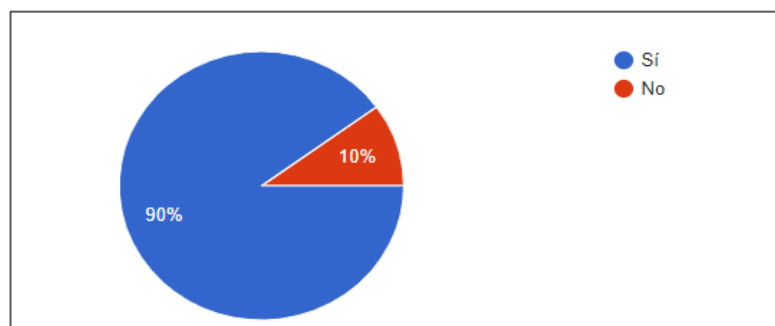


Figura 6. Personas encuestadas que disponen de un dispositivo móvil

Elaborado por: Karen Aguaysa

Fuente: Encuesta aplicada

Análisis e interpretación de resultados:

De acuerdo con los resultados representados en la Figura 6, es posible observar que el 90% de las personas encuestadas cuentan con al menos un dispositivo móvil dentro de su hogar, el 10% no cuenta con ningún dispositivo móvil, por lo que se demuestra que la mayoría de los clientes del minimarket tienen la posibilidad de comprar productos a través de un dispositivo inteligente.

Pregunta N° 7: ¿Ha realizado compras de diferentes productos a través de una aplicación móvil?

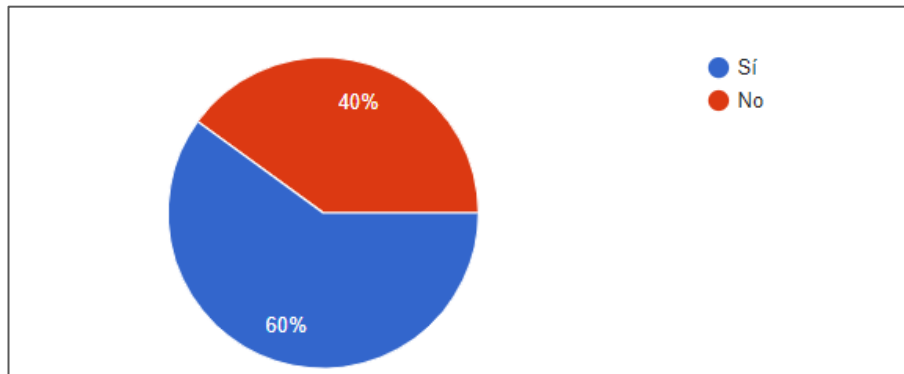


Figura 7. Personas encuestadas que han realizado compras a través de una aplicación móvil

Elaborado por: Karen Aguaysa

Fuente: Encuesta aplicada

Análisis e interpretación de resultados:

De acuerdo con los resultados representados en la Figura 7, es posible observar que el 60% de las personas encuestadas han realizado compras a través de una aplicación móvil y el 40% nunca ha realizado compras a través de éste mecanismo, por lo que se demuestra que la mayoría de clientes podrían interactuar con facilidad con una aplicación móvil en la que pueda adquirir sus productos.

Pregunta N° 8: ¿Le gustaría que existiera una aplicación móvil que gestione su lista de compras en el Minimarket “Paty”?

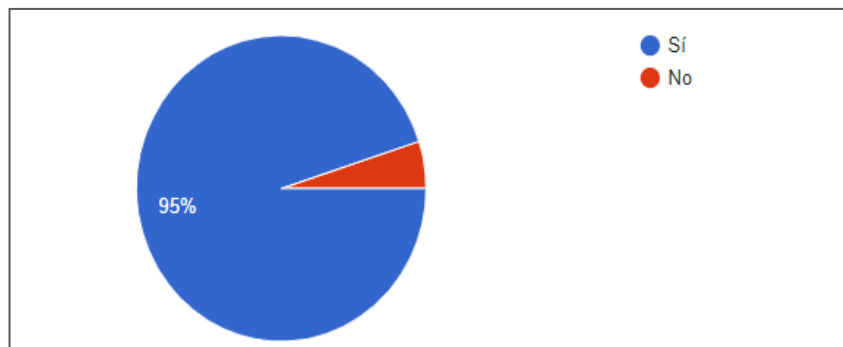


Figura 8. Personas encuestadas que desean una aplicación móvil para el Minimarket "Paty"

Elaborado por: Karen Aguaysa

Fuente: Encuesta aplicada

Análisis e interpretación de resultados:

De acuerdo con los resultados representados en la Figura 8, es posible observar que el 95% de las personas encuestadas, le gustaría que existiera una aplicación móvil que gestione sus compras en el minimarket “Paty”, el otro 5% no tiene conocimiento necesario para manipular una aplicación móvil, por lo que se demuestra que los clientes estarían dispuestos a innovar en una nueva manera de adquirir sus productos.

Pregunta N° 9: ¿Cree que, al contar con una aplicación móvil, que permita enviar la lista de compras y que le notifique cuando esté listo su pedido, optimizaría tiempo en las compras diarias y garantizará su bioseguridad?

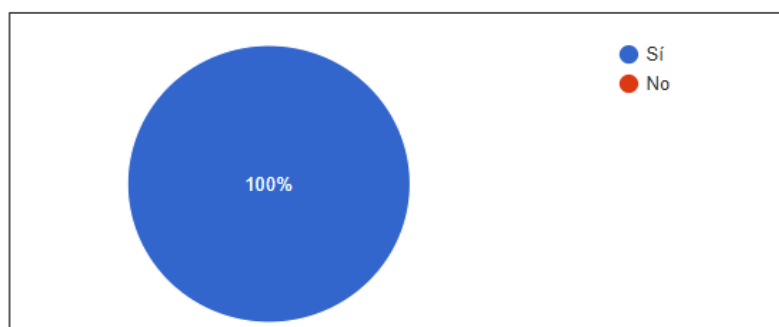


Figura 9. Personas encuestadas que con una app móvil optimizarán tiempo
Elaborado por: Karen Aguaysa
Fuente: Encuesta aplicada

Análisis e interpretación de resultados:

De acuerdo con los resultados representados en la Figura 9, es posible observar que el 100% de las personas encuestadas creen que al contar con una aplicación móvil que permita gestionar las compras para en el minimarket, permitiendo optimizar su tiempo para realizar otras actividades y sobre todo salvaguardar su salud, evitando aglomeraciones.

Pregunta N° 10: ¿Le gustaría que la entrega de los productos fuera a domicilio?

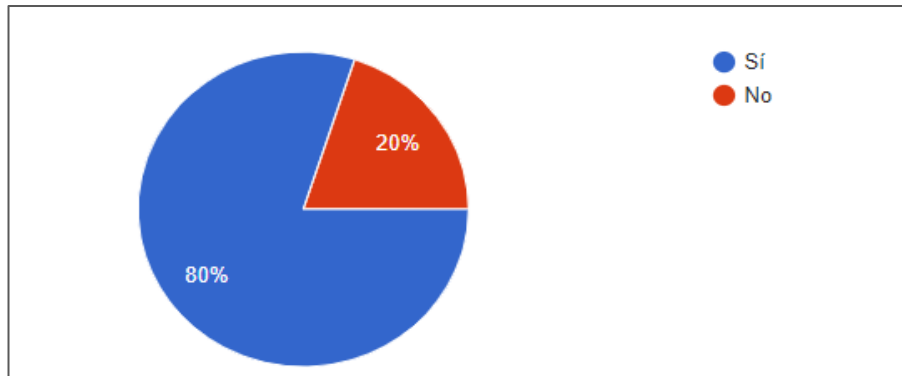


Figura 10. Personas encuestadas que les gustaría el servicio a domicilio

Elaborado por: Karen Aguaysa

Fuente: Encuesta aplicada

Análisis e interpretación de resultados:

De acuerdo con los resultados representados en la Figura 10, es posible observar que el 80% de las personas encuestadas, les gustaría que sus pedidos llegaran a domicilio y el 20% no creen necesario el hecho de que sus productos lleguen a domicilio, puesto que podrían retirar su pedido en el minimarket.

2.2.4 Procesamiento y análisis de datos

De acuerdo a la entrevista realizada al administrador del minimarket “Paty” se demostró que:

- Es necesario llevar el control de las ventas realizadas a diario para controlar de mejor manera la inversión del capital, así como también mantener un detalle de ganancias.
- Es importante innovar los mecanismos de venta, utilizando nueva tecnología, que faciliten los procesos de venta y que brinden un mejor servicio a sus clientes.
- Es importante controlar a tiempo el stock de sus productos, para garantizar la satisfacción de los clientes y no generar pérdidas de ventas y de clientes.

- Es necesario mantener el orden de los productos para, encontrarlos con facilidad para optimizar el tiempo del administrador, así como el de los clientes.

De acuerdo a la encuesta realizada a los clientes del minimarket “Paty”, se demostró que:

- La mayoría de personas encuestadas, se tardan un tiempo entre 00:15 a 00:30 minutos en realizar sus compras y que en varias ocasiones existe aglomeración de personas.
- Se considera que el negocio podría innovar en ciertos aspectos para mejorar el servicio que brindan a los clientes de la zona.
- Consideran que con la emergencia sanitaria por el covid 19 que atraviesa actualmente el país, ellos buscan nuevos mecanismos para proteger su salud y prevenir el contagio del mismo.
- A los clientes, les gustaría que exista una aplicación móvil que les permita realizar sus compras en el minimarket y algunos de ellos consideran necesario el servicio a domicilio, ya que de ésta manera ellos lograrían optimizar su tiempo y proteger su salud.

CAPÍTULO III

RESULTADOS Y DISCUSIÓN

3.1. Análisis y discusión de resultados

3.1.1. Determinación del framework de desarrollo web – móvil

Tomando en cuenta la información y características recabadas en la Tabla 1. Comparación de frameworks para aplicación web multiplataforma y Tabla 2. Comparación de frameworks para aplicación móvil multiplataforma, se ha considerado desarrollar la aplicación web en el framework de desarrollo llamado Angular y la aplicación móvil en el framework de desarrollo móvil llamado Ionic.

El framework de desarrollo conocido como Angular utiliza un lenguaje de programación basado en JavaScript y TypeScript, lo cual da la capacidad de desarrollar de manera muy sencilla y compatible con varios navegadores y a esto, se le suma que es multiplataforma, además que typescript llega a complementarlo con mucha más características que brindan facilidad de desarrollo, así también es un framework que tiene un gran porcentaje de reusabilidad de código, lo que permite realizar una programación con optimización de código y presenta una gran facilidad de uso, permitiendo que todas las personas sean capaces de aprender de manera rápida y sencilla.

Por otra parte, el framework de desarrollo móvil llamado Ionic, presenta grandes ventajas en el desarrollo de aplicaciones móviles, ya que maneja el mismo lenguaje de programación que Angular y otros marcos front-end lo que permite presentar un alto porcentaje de reutilización de código y mantener una integración perfecta con los mismos, también brinda la posibilidad de aprender a desarrollar de una manera fácil y presenta actualizaciones constantemente, con la finalidad de que los usuarios no tengan que recurrir a la instalación de paquetes aislados, por el contrario, brinda herramientas y funciones que van a la vanguardia con la evolución de la tecnología móvil.

3.1.2. Determinación de la metodología de desarrollo del proyecto

Para determinar la mejor metodología a aplicar en el siguiente proyecto, se ha realizado una tabla comparativa, entre las metodologías más utilizadas y más relevantes para el desarrollo de proyectos.

Tabla 5. Comparación de metodologías ágiles
Elaborado por: Karen Aguaysa

Características	XP	KANBAN	SCRUM
Tamaño del proyecto	Pequeños y medianos	Pequeños y medianos	Pequeños, medianos y grandes
Tamaño del equipo	Menor de 10 personas	Equipos menores de 10 personas	Equipos menores de 10 personas
Framework	Basado en la adaptabilidad, flexibilidad, y dinámico funcional	Basado en la gestión del sistema de control de la producción usando cartas que proveían una señal visual para provocar acción	Gestión y desarrollo de software, a través de un proceso iterativo - incremental
Objetivos	<ul style="list-style-type: none"> • Satisfacción cliente • Trabajo en grupo • Variables: Coste, tiempo, calidad y alcance 	<ul style="list-style-type: none"> • Visualizar el trabajo • Determinar el límite del trabajo • Medir tiempo 	<ul style="list-style-type: none"> • Resultados inmediatos • Requisitos cambiantes • Innovación y competitividad
Fases	<ul style="list-style-type: none"> • Planificación • Diseño • Codificación • Pruebas 	<ul style="list-style-type: none"> • Instrucción • Implementación en componentes con más problemas. • Implementación en componentes restantes • Revisión 	<ul style="list-style-type: none"> • Planeamiento • Montaje • Desarrollo • Liberación
Iteraciones	Plazo variable	Plazo fijo y variable	Plazo fijo
Roles	<ul style="list-style-type: none"> • Coach • Big Boss • Cliente 	Ninguno	<ul style="list-style-type: none"> • Scrum Master • Product Owner

Accesibilidad al cambio	No accesible a cambios durante la iteración, prioriza lo especificado por el cliente	Accesible al cambio en cualquier momento	No accesible a cambios durante el Sprint
Revisión	Integrar como mínimo una vez al día y realizar pruebas sobre el proceso	Exhaustiva del sistema para determinar qué puntos deben reordenarse	Revisión diaria de: Trabajo del día anterior Trabajo a realizar Impedimentos

Al verificar en la Tabla 5. Comparación de metodologías ágiles, cada una de las características más importantes de las metodologías comparadas, se elige la metodología XP (Xtreme Programming) como la más adecuada para implementarla en el desarrollo del presente proyecto, basándose en que es aquella metodología que se ajusta a las características del mismo y presenta un panorama muy interactivo con el cliente ya que se centra fundamentalmente en la satisfacción del usuario final.

3.2. Desarrollo de la propuesta

3.2.1. Fase I: Exploración

En esta fase del proyecto se recaba la información más relevante y concisa referente a las características que tendrá el sistema y las necesidades que tienen los usuarios finales del mismo, además se analizarán todos los requerimientos necesarios para satisfacer dichas necesidades.

Dentro de ésta fase se pretende familiarizar con todas las herramientas que han sido elegidas para el desarrollo del presente proyecto, de manera que sea mucho más fácil y rápido de realizarlo.

3.2.1.1. Levantamiento de información

Para el levantamiento de la información, se utilizó dos métodos de investigación, como son una encuesta dirigida hacia los clientes del minimarket “Paty” y también se realizó una entrevista personal con el administrados del minimarket, a través de éstos métodos se ha podido determinar las necesidades que presenta actualmente dicho negocio,

respecto a el manejo del inventario de sus distintos productos, así como la necesidad de implementar nuevas tecnologías de venta que permita mejorar la atención a sus clientes y sobre todo brindar seguridad en el bienestar de los mismos, debido a la actual emergencia sanitaria que atraviesa el país y el mundo entero.

Dentro de la encuesta que se ha realizado a los clientes, se establecieron preguntas cerradas con selección múltiple, éstas fueron desarrolladas de manera clara y enfocadas a conocer los problemas que encuentran actualmente en el proceso de atención al cliente en el minimarket y la posibilidad de que los clientes se adecuen a un nuevo mecanismo de adquisición de los productos, a través de aplicaciones móviles.

Dentro de la encuesta que ha realizado al administrador, se establecieron preguntas abiertas, en donde el mismo tuvo la oportunidad de expresar su criterio y necesidades en cuanto a la manera de administrar el inventario de sus productos y mostrar interés en innovar en su negocio, de manera que se adecue a las nuevas necesidades que presentan los clientes, el cuestionario anteriormente mencionado, consta de las preguntas estructuradas en el Capítulo II, apartado 2.1 Materiales. **¡Error! No se encuentra el origen de la referencia.**

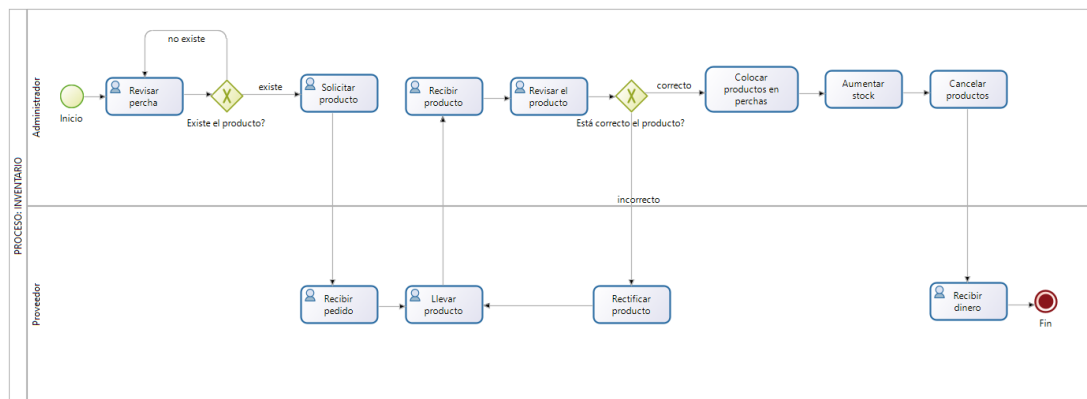
También la entrevista, permite conocer los requerimientos fundamentales que resalta el administrador que debe tener el sistemas para que contribuya a un mejoramiento de dichos procesos que actualmente se realizan de manera incorrecta dentro del minimarket, el cuestionario anteriormente mencionado, consta de las preguntas estructuradas en la Tabla 3. Modelo Entrevista Administrador

3.2.1.2. Descripción de procesos

Proceso de Inventario

Figura 11. Proceso de Inventario

Elaborado por: Karen Aguaysa



A través del diagrama representado en la Figura 11. Proceso de Inventario, se puede determinar las actividades funcionales que realizan cada uno de los actores, pudiendo comprender claramente el proceso de inventario.

Recursos: Dentro de los recursos de éste proceso se encuentran el recurso humano y el recurso material, que en éste caso serían los productos que son tangibles a las personas.

Actividades: Este proceso cuenta con una variedad de actividades para que sea completamente funcional, pero una de las actividades más relevante es el control del stock de los productos, puesto que el administrador del minimarket al vender un producto, no tiene la noción de si ya están agotándose los productos y podría llegar a no satisfacer las necesidades de los clientes.

Actores: Dentro de los actores que se encuentran en éste proceso son:

- Administrador
- Proveedor

Salida: El resultado del proceso, da como conclusión que, al realizar una inspección de las perchas, se obtiene el número de productos con los que cuenta actualmente el

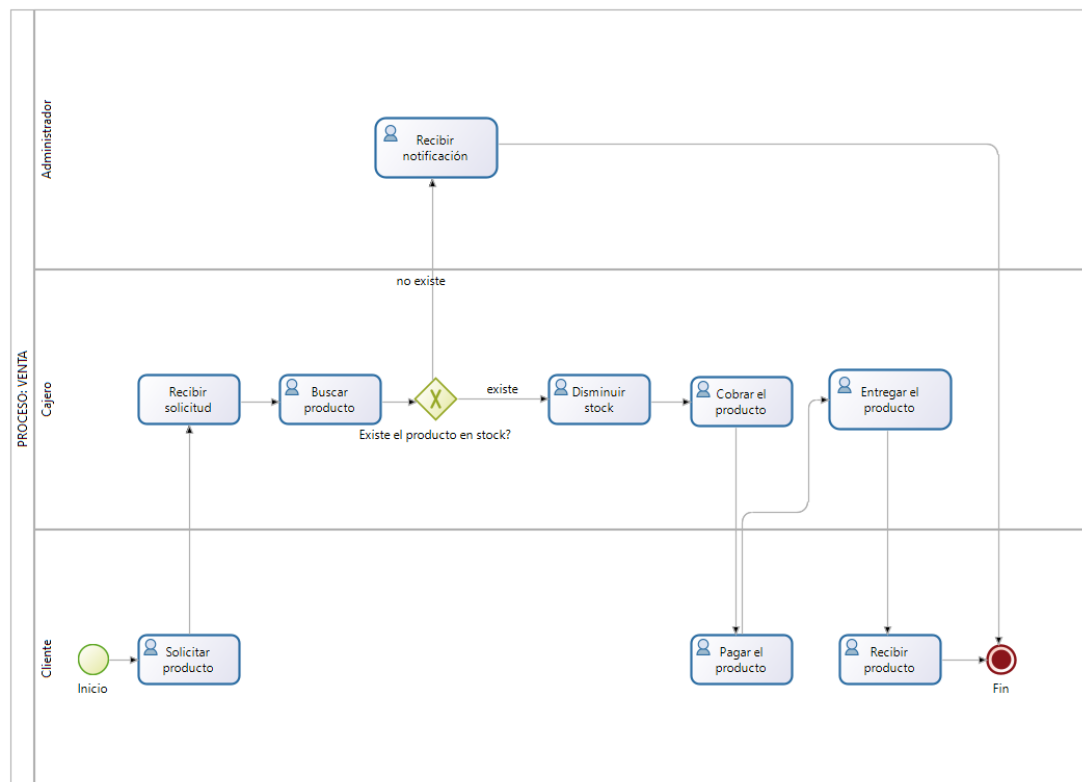
administrador para la venta, lo cual le permitirá tomar una decisión a futuro para adquirir de nuevos productos para el aumento del stock, con la finalidad de satisfacer las necesidades de sus clientes.

Destinatario: El destinatario de éste proceso, es el Administrador del minimarket, ya que es la persona encargada de dotar de productos y a quién le beneficiará que éste proceso se esté cumpliendo de manera correcta.

Proceso de venta

Figura 12. Proceso de Venta

Elaborado por: Karen Aguaysa



A través del diagrama representado en la Figura 12, se puede determinar las actividades funcionales que realizan cada uno de los actores, pudiendo comprender claramente el proceso de venta.

Recursos: Dentro de los recursos de éste proceso se encuentran el recurso humano, el recurso material que en éste caso serían los productos que son tangibles a las personas

y el recurso económico que viene representado por el valor en efectivo de cada uno de los productos que se adquieren.

Actividades: Las actividades que se realizan dentro de éste proceso son importantes para la venta, pero una de las actividades más destacadas es la disminución del stock del producto adquirido, ya que al realizarse ésta actividad permite mantener el control de la cantidad de productos existentes en las perchas, así como también lo importante de éste proceso, es el cálculo adecuado del total a cancelar de los clientes, en donde se debe tener precisión.

Actores: Los actores que se ven incluidos dentro de éste proceso son los siguientes:

- Administrador
- Cajero
- Cliente

Salida: El resultado del proceso, es un administrador informado acerca de la cantidad precisa de productos que posee en sus perchas, un cajero que se encarga de cobrar un valor exacto en efectivo y un cliente satisfecho que adquirió sus productos, cancelando un valor correcto y que no ha sufrido afectaciones de ningún tipo.

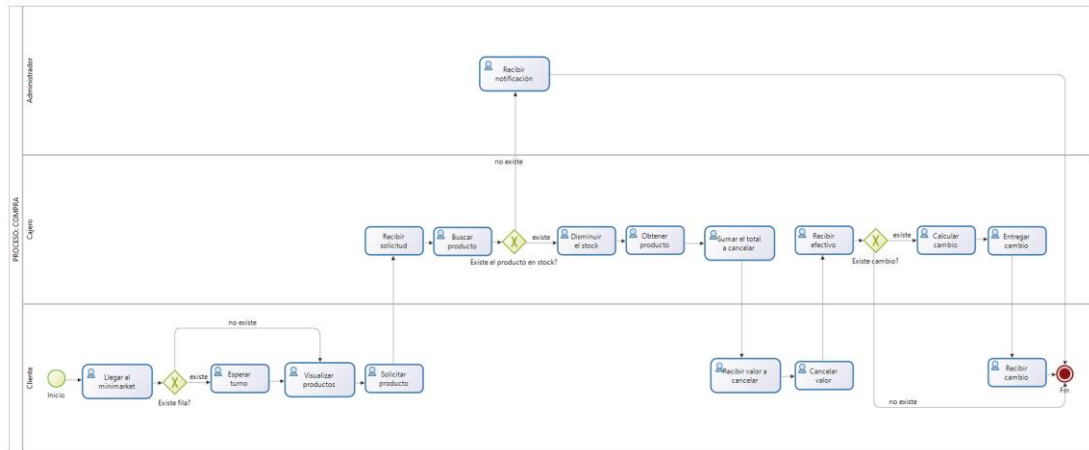
Destinatario: Los destinatarios de éste proceso son:

- **Administrador:** Recibe datos acerca del número de productos que tiene en stock para la venta.
- **Cajero:** Recibe datos informativos, acerca de lo que desea comprar el cliente y posteriormente recibe datos para calcularlos y difundirlos a los clientes.
- **Cliente:** Recibe información acerca de los productos que adquirió e información acerca de cuánto debe pagar por la adquisición del mismo.

Proceso de compra

Figura 13. Proceso de Compra

Elaborado por: Karen Aguaysa



A través del diagrama representado en la Figura 13, se puede determinar las actividades funcionales que realizan cada uno de los actores, pudiendo comprender claramente el proceso de compra.

Recursos: Dentro de los recursos de éste proceso se encuentran el recurso humano, el recurso material que en éste caso serían los productos que son tangibles a las personas y el recurso económico que viene representado por el valor en efectivo de cada uno de los productos que se adquieren.

Actividades: Las actividades que se realizan en éste proceso, son muy similares al proceso de venta, pero desde la perspectiva del cliente, y todas las actividades que a él le comprometen para adquirir cada uno de sus productos, entre las más destacadas se puede visualizar es la de iniciar a realizar la compra, primero observando si existe una fila y posteriormente respetando un turno para ser atendido, una vez que ha sido atendido visualiza los productos y escoge de acuerdo a la lista, posteriormente luego de haber seleccionado todos los productos, el cajero realiza un cálculo total del valor a cancelar, para que el cliente lo pague y se retire del minimarket.

Actores:

- **Administrador:** Recibe datos acerca del número de productos que tiene en stock para la venta.
- **Cajero:** Recibe datos informativos, acerca de lo que desea comprar el cliente y posteriormente recibe datos para calcularlos y difundirlos a los clientes.
- **Cliente:** Recibe información acerca de los productos que adquirió e información acerca de cuánto debe pagar por la adquisición del mismo.

Salida: El resultado del proceso, es un administrador informado acerca de la cantidad precisa de productos que posee en sus perchas, un cajero que se encarga de cobrar un valor exacto en efectivo y un cliente satisfecho que adquirió sus productos, cancelando un valor correcto pero que ha tenido que realizar un proceso que no ha permitido la optimización de su tiempo.

Destinatario: El destinatario de este proceso es únicamente el cliente, quien después de realizar todas las actividades anteriormente descritas, ha logrado adquirir la lista de productos por la que tuvo que acercarse al minimarket y completar su compra.

3.2.1.3. Análisis de resultados

En las encuestas realizadas a una serie de clientes que frecuentan el minimarket “Paty”, para realizar la compra de sus productos, se pudo identificar que por la actual emergencia sanitaria que se atraviesa por el covid-19, sería de gran utilidad contar con una aplicación móvil que permita realizar una lista de productos para adquirir y tan solo preocuparse por retirarlos o quien lo requiera podría realizar su pedido a domicilio, de manera que evitarían realizar largas filas en el minimarket y aglomeraciones de personas, por otra parte, se considera que la mayoría de personas cuentan con la tecnología necesaria para hacer uso de la aplicación.

En la entrevista realizada a administrador del Minimarket “Paty”, el Sr. Enrique Gavilema, ha manifestado la situación actual de la manera en que se llevan a cabo los procesos de inventario de los productos, en el cual se evidencian ciertas necesidades

de mejora, pudiendo implementarse un sistema que realice dicho proceso y lo mejore, permitiendo tener un control preciso, respecto a la cantidad de productos que constan en las perchas.

También se evidencia la necesidad de que el cajero debe tener la facilidad de realizar de manera correcta los cálculos de un conjunto de productos, pero de una manera que optimice su tiempo y brinde resultados confiables al cliente. Por lo que se determinaría dos roles del sistema principales:

- **Administrador:** Es el encargado de toda la parte organizacional del minimarket, es decir tendrá acceso al control de todo el sistema administrativo
- **Cajero:** Es el encargado de disminuir el stock de los productos con cada venta que realice y de realizar el cobro de los mismos, por lo que tendrá solo acceso al módulo de facturación.

Por otra parte, expresa la necesidad de mejorar el proceso de adquisición de productos por parte de los clientes, puesto que, a pesar de querer satisfacer todas las necesidades de sus clientes, muchas veces existe abarrotamiento de los mismos, lo que no le ha permitido brindar un servicio de calidad en su totalidad, por lo que se considera importante la creación de una aplicación que le permita solventar dicha situación y colaborar con las respectivas medidas de bioseguridad que establece el Ministerio de Salud del Ecuador y cuidar la salud y bienestar de todos quienes se han vuelto sus fieles clientes.

3.2.1.4. Definición de roles

Al establecer claramente los roles, se puede identificar las actividades y responsabilidades que tiene cada uno de los actores dentro del desarrollo del proyecto y de ésta manera permitir que se lleve de manera organizada el proceso de desarrollo del mismo.

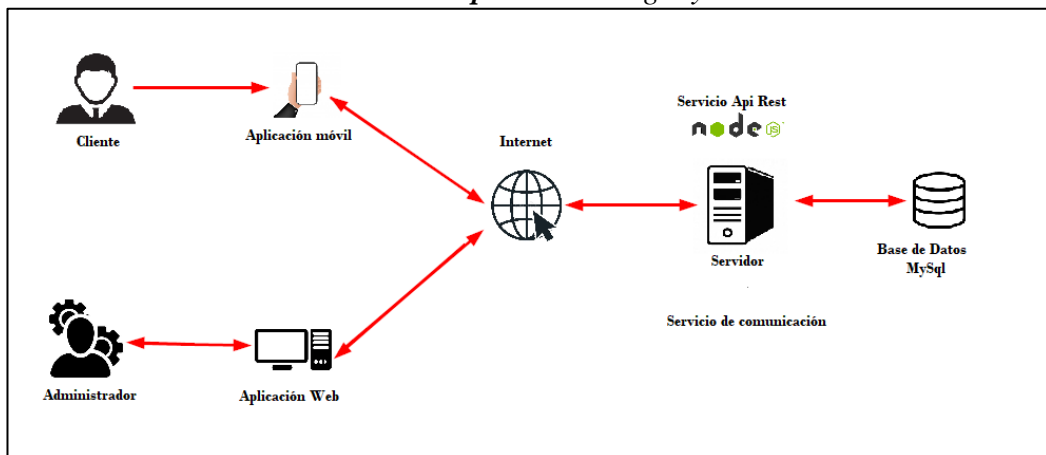
Tabla 6. Definición de Roles
Elaborado por: Karen Aguaysa

NOMBRE	ROL	DESCRIPCIÓN
Karen Aguaysa	Programador	Persona responsable de realizar el análisis, diseño, desarrollo y pruebas del sistema, acorde a las necesidades del usuario, respetando las características y tiempos definidos
Karen Aguaysa	Coach	Persona encargada de realizar revisiones frecuentemente para verificar que se cumpla lo anteriormente organizado.
Ing. Franklin Mayorga	Tester	Persona que requiere el sistema y quien pone a prueba el mismo ya que tiene el conocimiento del funcionamiento de los procesos.

3.2.1.5. Arquitectura del sistema

En la arquitectura del sistema, se puede observar el diseño general de la aplicación web/móvil y la manera en que interactuarán los distintos componentes con las tecnologías, de manera que se mantenga la comunicación para la ejecución de procesos de manera concisa.

Figura 14. Arquitectura de la aplicación
Elaborado por: Karen Aguaysa



Por lo que se puede observar en la Figura 14. Arquitectura de la aplicación, el sistema se encuentra basado en una arquitectura de tres capas, que permite mantener conectividad entre las mismas a través del paso de parámetros, lo cual la hace una arquitectura muy flexible para ajustarse a varios cambios, sin necesidad de afectar a

todos los módulos que conforman el sistema, convirtiéndola en una aplicación escalable en cualquiera de sus capas.

3.2.2. Fase II: Planificación

Se ha considerado que para la planificación del presente proyecto se desarrollarán historias de usuario para a través de las mismas ejecutar las diferentes actividades, contemplando el grado de importancia de las mismas.

3.2.2.1. Historias de usuario

Las historias de usuario se han desarrollado para definir las características y funciones principales que va a realizar el sistema propuesto y el producto final que espera el usuario del mismo. Una vez que sean realizadas las historias de usuarios.

Para la elaboración de las historias de usuario se ha planteado el modelo establecido en la Tabla 7.

*Tabla 7. Plantilla de Historia de Usuario
Elaborado por: Karen Aguaysa*

Historia de usuario			
Número:		Usuario:	
Nombre de la historia:			
Prioridad en el negocio:		Puntos estimados:	
Riesgo en el desarrollo:		Iteración asignada:	
Descripción:			
Responsable:			
Observaciones:			

A continuación, se describirá cada uno de los elementos de la plantilla:

- **Número:** Representa un identificador numérico que será exclusivo para cada una de las historias de usuario.
- **Usuario:** Persona a quien va dirigida la historia de usuario.
- **Prioridad en el negocio:** Valor (Alta, Media y Baja) de acuerdo a las necesidades del usuario final.

- **Puntos estimados:** El número de días necesarios para desarrollar dicha historia de usuario.

Para esto, es necesario elaborar una tabla de acuerdo a la prioridad de cada una de las historias de usuario.

Tabla 8. *Puntos estimados de la Historia de Usuario*
Elaborado por: Karen Aguaysa

PRIORIDAD	CRITERIO
5	Indispensable
4	Esencial
3	Obligatorio
2	Necesario
1	Conveniente

- **Riesgo en el desarrollo:** Valor (Alto, Medio y Bajo) de acuerdo a los riesgos existente para que se desarrolle dicha historia de usuario.
- **Iteración asignada:** Se refiere a la iteración que ha sido asignada a una respectiva historia de usuario.
- **Descripción:** En esta parte se detallan claramente las características, datos y validaciones que deben ser tomadas en cuenta al momento del desarrollo de la historia de usuario.
- **Responsable:** Es la persona o grupo de personas encargadas de desarrollar la historia de usuario.
- **Observaciones:** En este apartado se puede realizar aclaraciones que son relevantes para el desarrollo de la historia de usuario.

Una vez que se han especificado cada uno de los elementos que componen las historias de usuario, se ha desarrollado las historias de usuario necesarias para el desarrollo del sistema.

Tabla 9. Historia de Usuario - Estructura del sistema
Elaborado por: Karen Aguaysa

Historia de usuario			
Número:	001	Usuario:	Administrador
Nombre de la historia: Definición de la estructura del sistema			
Prioridad en el negocio:	Alta	Puntos estimados:	4
Riesgo en el desarrollo:	Alto	Iteración asignada:	1
Descripción: Conocer la estructura básica del presente proyecto, ayudará a visualizar la manera en que se van a integrar las diferentes tecnologías y herramientas de desarrollo del proyecto y también permitirá en un futuro facilitar el mantenimiento del mismo.			
Responsable: Karen Aguaysa			
Observaciones: Ninguna			

Tabla 10. Historia de Usuario - Diseño de la base de datos
Elaborado por: Karen Aguaysa

Historia de usuario			
Número:	002	Usuario:	Administrador
Nombre de la historia: Diseño de la base de datos			
Prioridad en el negocio:	Alta	Puntos estimados:	5
Riesgo en el desarrollo:	Alto	Iteración asignada:	1
Descripción: Se diseñará una base de datos que permita un fácil acceso de los usuarios a los datos de manera concisa, ésta debe ser desarrollada de la manera correcta para que realice todas las consultas necesarias.			
Responsable: Karen Aguaysa			
Observaciones: Ninguna			

Tabla 11. Historia de Usuario - Ingreso a la aplicación web
Elaborado por: Karen Aguaysa

Historia de usuario			
Número:	003	Usuario:	Administrador
Nombre de la historia: Ingreso a la aplicación web			
Prioridad en el negocio:	Alta	Puntos estimados:	4
Riesgo en el desarrollo:	Medio	Iteración asignada:	1
Descripción: Al ser la aplicación manejada por varios usuarios, es necesario que se ingrese a la misma a través del número de cédula y la contraseña, para mantener la seguridad de la aplicación web.			
Responsable: Karen Aguaysa			
Observaciones: El administrador del sistema, es el único que puede crear un usuario para el manejo de las distintas funciones.			

Tabla 12. Historia de Usuario - Restablecer contraseña
Elaborado por: Karen Aguaysa

Historia de usuario			
Número:	004	Usuario:	Administrador
Nombre de la historia: Restablecer contraseña			
Prioridad en el negocio:	Alta	Puntos estimados:	4
Riesgo en el desarrollo:	Medio	Iteración asignada:	1
Descripción: El administrador cuenta con la posibilidad de cambiar las contraseñas de todos los usuarios del sistema, sin embargo, otro usuario con distinto rol tiene la capacidad de cambiar su propia contraseña.			
Responsable: Karen Aguaysa			
Observaciones: Dentro del sistema existe una ventana que permite realizar dicho cambio de contraseña.			

Tabla 13. Historia de Usuario - Visualización de la pantalla de inicio
Elaborado por: Karen Aguaysa

Historia de usuario			
Número:	005	Usuario:	Administrador
Nombre de la historia: Visualización de la pantalla de inicio			
Prioridad en el negocio:	Alta	Puntos estimados:	4
Riesgo en el desarrollo:	Medio	Iteración asignada:	1
Descripción: La pantalla principal permitirá visualizar un menú de los distintos módulos a los que tiene acceso el usuario que se ha registrado.			
Responsable: Karen Aguaysa			
Observaciones: En la pantalla de inicio también debe encontrarse la sección donde permita abandonar la sesión y las notificaciones de la lista de los pedidos realizados a través de la aplicación móvil.			

Tabla 14. Historia de Usuario - Registro de rol
Elaborado por: Karen Aguaysa

Historia de usuario			
Número:	006	Usuario:	Administrador
Nombre de la historia: Registro de rol			
Prioridad en el negocio:	Alta	Puntos estimados:	5
Riesgo en el desarrollo:	Medio	Iteración asignada:	2
Descripción: Se visualizará una pantalla en donde se encuentren los campos para crear un nuevo rol dentro de la aplicación web.			
Responsable: Karen Aguaysa			
Observaciones: El administrador es el único que puede crear un nuevo rol.			

Tabla 15. Historia de Usuario - Visualización de Roles
Elaborado por: Karen Aguaysa

Historia de usuario			
Número:	007	Usuario:	Administrador
Nombre de la historia: Visualización de roles			
Prioridad en el negocio:	Alta	Puntos estimados:	4
Riesgo en el desarrollo:	Medio	Iteración asignada:	2
Descripción: Se visualizará en la pantalla una lista de todos los roles activos que se encuentren registrados en la aplicación.			
Responsable: Karen Aguaysa			
Observaciones: Ninguna.			

Tabla 16. Historia de Usuario - Edición de roles
Elaborado por: Karen Aguaysa

Historia de usuario			
Número:	008	Usuario:	Administrador
Nombre de la historia: Edición de roles			
Prioridad en el negocio:	Alta	Puntos estimados:	5
Riesgo en el desarrollo:	Medio	Iteración asignada:	2
Descripción: En la pantalla se visualizará un ícono de actualización en el que al dar click se cargarán los datos de dicho rol en la parte superior de la pantalla y aquellos que puedan ser modificados se activarán.			
Responsable: Karen Aguaysa			
Observaciones: Ninguna			

*Tabla 17. Historia de Usuario - Eliminación de roles
Elaborado por: Karen Aguaysa*

Historia de usuario			
Número:	009	Usuario:	Administrador
Nombre de la historia: Eliminación de roles			
Prioridad en el negocio:	Alta	Puntos estimados:	3
Riesgo en el desarrollo:	Medio	Iteración asignada:	2
Descripción: En la pantalla se visualizará un ícono de eliminación de rol, el mismo que al ser presionado emitirá una alerta para verificar que la acción sea la deseada por el usuario, una vez aceptada la misma se procederá a eliminar y se actualizará el listado.			
Responsable: Karen Aguaysa			
Observaciones: Ninguna			

*Tabla 18. Historia de Usuario - Registro de usuario
Elaborado por: Karen Aguaysa*

Historia de usuario			
Número:	010	Usuario:	Administrador
Nombre de la historia: Registro de usuario			
Prioridad en el negocio:	Alta	Puntos estimados:	5
Riesgo en el desarrollo:	Medio	Iteración asignada:	2
Descripción: Se visualizará una pantalla en donde se encuentren los campos para crear un nuevo usuario dentro de la aplicación web.			
Responsable: Karen Aguaysa			
Observaciones: El administrador es el único que puede crear un nuevo usuario.			

Tabla 19. Historia de Usuario - Visualización de usuario
Elaborado por: Karen Aguaysa

Historia de usuario			
Número:	011	Usuario:	Administrador
Nombre de la historia: Visualización de usuarios			
Prioridad en el negocio:	Alta	Puntos estimados:	4
Riesgo en el desarrollo:	Medio	Iteración asignada:	2
Descripción: Se visualizará en la pantalla una lista de todos los usuarios activos que se encuentren registrados en la aplicación.			
Responsable: Karen Aguaysa			
Observaciones: No se podrá visualizar las contraseñas de los usuarios.			

Tabla 20. Historia de Usuario - Edición de usuarios
Elaborado por: Karen Aguaysa

Historia de usuario			
Número:	012	Usuario:	Administrador
Nombre de la historia: Edición de usuarios			
Prioridad en el negocio:	Alta	Puntos estimados:	5
Riesgo en el desarrollo:	Medio	Iteración asignada:	3
Descripción: En la pantalla se visualizará un ícono de actualización en el que al dar click se cargarán los datos de dicho usuario en la parte superior de la pantalla y aquellos que puedan ser modificados se activarán.			
Responsable: Karen Aguaysa			
Observaciones: En este caso no se podrá actualizar la contraseña, puesto que para eso existe otro módulo de restablecimiento de contraseña.			

Tabla 21. Historia de Usuario - Eliminación de usuarios
Elaborado por: Karen Aguaysa

Historia de usuario			
Número:	013	Usuario:	Administrador
Nombre de la historia: Eliminación de usuarios			
Prioridad en el negocio:	Alta	Puntos estimados:	3
Riesgo en el desarrollo:	Medio	Iteración asignada:	3
Descripción: En la pantalla se visualizará un ícono de eliminación de usuario, el mismo que al ser presionado emitirá una alerta para verificar que la acción sea la deseada por el usuario, una vez aceptada la misma se procederá a eliminar y se actualizará el listado.			
Responsable: Karen Aguaysa			
Observaciones: Ninguna			

Tabla 22. Historia de Usuario - Registro de pasillo
Elaborado por: Karen Aguaysa

Historia de usuario			
Número:	014	Usuario:	Administrador
Nombre de la historia: Registro de pasillo			
Prioridad en el negocio:	Alta	Puntos estimados:	5
Riesgo en el desarrollo:	Medio	Iteración asignada:	3
Descripción: Se visualizará una pantalla en donde se encuentren los campos para crear un nuevo pasillo dentro de la aplicación web.			
Responsable: Karen Aguaysa			
Observaciones: El administrador es el único que puede crear un nuevo pasillo.			

Tabla 23. Historia de Usuario - Visualización de pasillos
Elaborado por: Karen Aguaysa

Historia de usuario			
Número:	015	Usuario:	Administrador
Nombre de la historia: Visualización de pasillos			
Prioridad en el negocio:	Alta	Puntos estimados:	4
Riesgo en el desarrollo:	Medio	Iteración asignada:	3
Descripción: Se visualizará en la pantalla una lista de todos los pasillos activos que se encuentren registrados en la aplicación.			
Responsable: Karen Aguaysa			
Observaciones: Ninguna.			

Tabla 24. Historia de Usuario - Edición de pasillos
Elaborado por: Karen Aguaysa

Historia de usuario			
Número:	016	Usuario:	Administrador
Nombre de la historia: Edición de pasillos			
Prioridad en el negocio:	Alta	Puntos estimados:	5
Riesgo en el desarrollo:	Medio	Iteración asignada:	3
Descripción: En la pantalla se visualizará un ícono de actualización en el que al dar click se cargarán los datos de dicho pasillo en la parte superior de la pantalla y aquellos que puedan ser modificados se activarán.			
Responsable: Karen Aguaysa			
Observaciones: Ninguna			

Tabla 25. Historia de Usuario - Eliminación de pasillos
Elaborado por: Karen Aguaysa

Historia de usuario			
Número:	017	Usuario:	Administrador
Nombre de la historia: Eliminación de pasillos			
Prioridad en el negocio:	Alta	Puntos estimados:	3
Riesgo en el desarrollo:	Medio	Iteración asignada:	3
Descripción: En la pantalla se visualizará un ícono de eliminación de pasillo, el mismo que al ser presionado emitirá una alerta para verificar que la acción sea la deseada por el usuario, una vez aceptada la misma se procederá a eliminar y se actualizará el listado.			
Responsable: Karen Aguaysa			
Observaciones: Ninguna			

Tabla 26. Historia de Usuario - Registro de percha
Elaborado por: Karen Aguaysa

Historia de usuario			
Número:	018	Usuario:	Administrador
Nombre de la historia: Registro de percha			
Prioridad en el negocio:	Alta	Puntos estimados:	5
Riesgo en el desarrollo:	Medio	Iteración asignada:	4
Descripción: Se visualizará una pantalla en donde se encuentren los campos para crear una nueva percha dentro de la aplicación web.			
Responsable: Karen Aguaysa			
Observaciones: El administrador es el único que puede crear una nueva percha.			

Tabla 27. Historia de Usuario - Visualización de perchas
Elaborado por: Karen Aguaysa

Historia de usuario			
Número:	019	Usuario:	Administrador
Nombre de la historia: Visualización de perchas			
Prioridad en el negocio:	Alta	Puntos estimados:	4
Riesgo en el desarrollo:	Medio	Iteración asignada:	4
Descripción: Se visualizará en la pantalla una lista de todas las perchas activas que se encuentren registrados en la aplicación.			
Responsable: Karen Aguaysa			
Observaciones: Ninguna.			

Tabla 28. Historia de Usuario - Edición de perchas
Elaborado por: Karen Aguaysa

Historia de usuario			
Número:	020	Usuario:	Administrador
Nombre de la historia: Edición de perchas			
Prioridad en el negocio:	Alta	Puntos estimados:	5
Riesgo en el desarrollo:	Medio	Iteración asignada:	4
Descripción: En la pantalla se visualizará un ícono de actualización en el que al dar click se cargarán los datos de dicha percha en la parte superior de la pantalla y aquellos que puedan ser modificados se activarán.			
Responsable: Karen Aguaysa			
Observaciones: Ninguna			

Tabla 29. Historia de Usuario - Eliminación de perchas
Elaborado por: Karen Aguaysa

Historia de usuario			
Número:	021	Usuario:	Administrador
Nombre de la historia: Eliminación de perchas			
Prioridad en el negocio:	Alta	Puntos estimados:	3
Riesgo en el desarrollo:	Medio	Iteración asignada:	4
Descripción: En la pantalla se visualizará un ícono de eliminación de percha, el mismo que al ser presionado emitirá una alerta para verificar que la acción sea la deseada por el usuario, una vez aceptada la misma se procederá a eliminar y se actualizará el listado.			
Responsable: Karen Aguaysa			
Observaciones: Ninguna			

Tabla 30. Historia de Usuario - Registro de producto
Elaborado por: Karen Aguaysa

Historia de usuario			
Número:	022	Usuario:	Administrador
Nombre de la historia: Registro de producto			
Prioridad en el negocio:	Alta	Puntos estimados:	5
Riesgo en el desarrollo:	Medio	Iteración asignada:	4
Descripción: Se visualizará una pantalla en donde se encuentren los campos para crear un nuevo producto dentro de la aplicación web.			
Responsable: Karen Aguaysa			
Observaciones: El administrador es el único que puede crear un nuevo producto. Ésta pantalla debe tener la capacidad de insertar una imagen a cada producto.			

Tabla 31. Historia de Usuario - Visualización de productos
Elaborado por: Karen Aguaysa

Historia de usuario			
Número:	023	Usuario:	Administrador
Nombre de la historia: Visualización de productos			
Prioridad en el negocio:	Alta	Puntos estimados:	4
Riesgo en el desarrollo:	Medio	Iteración asignada:	4
Descripción: Se visualizará en la pantalla una lista de todos los productos activos que se encuentren registrados en la aplicación.			
Responsable: Karen Aguaysa			
Observaciones: Ninguna.			

Tabla 32. Historia de Usuario - Edición de productos
Elaborado por: Karen Aguaysa

Historia de usuario			
Número:	024	Usuario:	Administrador
Nombre de la historia: Edición de productos			
Prioridad en el negocio:	Alta	Puntos estimados:	5
Riesgo en el desarrollo:	Medio	Iteración asignada:	5
Descripción: En la pantalla se visualizará un ícono de actualización en el que al dar click se cargarán los datos de dicho producto en la parte superior de la pantalla y aquellos que puedan ser modificados se activarán.			
Responsable: Karen Aguaysa			
Observaciones: Ninguna			

Tabla 33. Historia de Usuario - Eliminación de productos
Elaborado por: Karen Aguaysa

Historia de usuario			
Número:	025	Usuario:	Administrador
Nombre de la historia: Eliminación de productos			
Prioridad en el negocio:	Alta	Puntos estimados:	3
Riesgo en el desarrollo:	Medio	Iteración asignada:	5
Descripción: En la pantalla se visualizará un ícono de eliminación de un producto, el mismo que al ser presionado emitirá una alerta para verificar que la acción sea la deseada por el usuario, una vez aceptada la misma se procederá a eliminar y se actualizará el listado.			
Responsable: Karen Aguaysa			
Observaciones: Ninguna			

Tabla 34. Historia de Usuario - Registro de venta
Elaborado por: Karen Aguaysa

Historia de usuario			
Número:	026	Usuario:	Administrador
Nombre de la historia: Registro de venta			
Prioridad en el negocio:	Alta	Puntos estimados:	5
Riesgo en el desarrollo:	Alto	Iteración asignada:	5
Descripción: Se visualizará una pantalla en donde se encuentren los campos para crear una nueva venta dentro de la aplicación web.			
Responsable: Karen Aguaysa			
Observaciones: Ninguna.			

*Tabla 35. Historia de Usuario - Visualización de ventas
Elaborado por: Karen Aguaysa*

Historia de usuario			
Número:	027	Usuario:	Administrador
Nombre de la historia: Visualización de ventas			
Prioridad en el negocio:	Alta	Puntos estimados:	4
Riesgo en el desarrollo:	Medio	Iteración asignada:	1
Descripción: Se visualizará en la pantalla una lista todas las ventas que se encuentren registradas en la aplicación.			
Responsable: Karen Aguaysa			
Observaciones: Ninguna.			

*Tabla 36. Historia de Usuario - Recepción del listado de compras
Elaborado por: Karen Aguaysa*

Historia de usuario			
Número:	028	Usuario:	Administrador
Nombre de la historia: Recepción de listado			
Prioridad en el negocio:	Alta	Puntos estimados:	4
Riesgo en el desarrollo:	Alto	Iteración asignada:	5
Descripción: Se visualizará una pantalla donde se encuentre un chat con el cliente que le permita recibir el pedido.			
Responsable: Karen Aguaysa			
Observaciones: Ninguna			

*Tabla 37. Historia de Usuario - Ingreso a la aplicación
Elaborado por: Karen Aguaysa*

Historia de usuario			
Número:	029	Usuario:	Cliente
Nombre de la historia: Ingreso a la aplicación			
Prioridad en el negocio:	Alta	Puntos estimados:	5
Riesgo en el desarrollo:	Medio	Iteración asignada:	5
Descripción: Se mostrará una pantalla en la cual se podrá ingresar las credenciales para validar el cliente.			
Responsable: Karen Aguaysa			
Observaciones: Las credenciales a validar son las siguientes: <ul style="list-style-type: none"> • Cédula del cliente • Contraseña del cliente 			

*Tabla 38. Historia de Usuario - Registro de cliente
Elaborado por: Karen Aguaysa*

Historia de usuario			
Número:	030	Usuario:	Cliente
Nombre de la historia: Registro de cliente			
Prioridad en el negocio:	Alta	Puntos estimados:	4
Riesgo en el desarrollo:	Medio	Iteración asignada:	1
Descripción: Se mostrará una pantalla en la cual el cliente puede ingresar una serie de datos, que permita tener acceso a las funcionalidades de la aplicación.			
Responsable: Karen Aguaysa			
Observaciones: Los datos a obtener de los clientes, son los siguientes: <ul style="list-style-type: none"> • Cédula • Nombre • Apellido • Dirección • Teléfono 			

*Tabla 39. Historia de Usuario - Catálogo de productos
Elaborado por: Karen Aguaysa*

Historia de usuario			
Número:	031	Usuario:	Cliente
Nombre de la historia: Visualización del catálogo de productos			
Prioridad en el negocio:	Alta	Puntos estimados:	5
Riesgo en el desarrollo:	Medio	Iteración asignada:	6
Descripción: Se mostrará una pantalla donde se encuentren todos los productos disponibles para los clientes, detallando sus precios.			
Responsable: Karen Aguaysa			
Observaciones: Ninguna			

*Tabla 40. Historia de Usuario - Carrito con productos elegidos
Elaborado por: Karen Aguaysa*

Historia de usuario			
Número:	032	Usuario:	Cliente
Nombre de la historia: Carrito con productos escogidos			
Prioridad en el negocio:	Alta	Puntos estimados:	5
Riesgo en el desarrollo:	Medio	Iteración asignada:	6
Descripción: Se mostrará una pantalla con todos los productos seleccionados por los clientes para realizar la compra.			
Responsable: Karen Aguaysa			
Observaciones: La pantalla debe contener el valor total de la compra realizada por el cliente.			

Tabla 41. Historia de Usuario - Confirmación de datos del comprador
Elaborado por: Karen Aguaysa

Historia de usuario			
Número:	033	Usuario:	Cliente
Nombre de la historia: Confirmación de datos			
Prioridad en el negocio:	Alta	Puntos estimados:	4
Riesgo en el desarrollo:	Medio	Iteración asignada:	6
Descripción: Se mostrará una pantalla en la que se encuentran cargados los datos de la compra a realizar.			
Responsable: Karen Aguaysa			
Observaciones: Ninguna			

Tabla 42. Historia de Usuario - Confirmación de la compra
Elaborado por: Karen Aguaysa

Historia de usuario			
Número:	034	Usuario:	Cliente
Nombre de la historia: Confirmación de la compra			
Prioridad en el negocio:	Media	Puntos estimados:	3
Riesgo en el desarrollo:	Bajo	Iteración asignada:	6
Descripción: Se mostrará una pantalla que le confirme al cliente que se ha realizado exitosamente su compra.			
Responsable: Karen Aguaysa			
Observaciones: Ninguna			

Tabla 43. Historia de Usuario - Envío del listado
Elaborado por: Karen Aguaysa

Historia de usuario			
Número:	035	Usuario:	Cliente
Nombre de la historia: Envío del listado			
Prioridad en el negocio:	Alta	Puntos estimados:	5
Riesgo en el desarrollo:	Medio	Iteración asignada:	6
Descripción: Se mostrará una pantalla donde se encuentre el pedido que ha realizado al minimarket el cliente.			
Responsable: Karen Aguaysa			
Observaciones: Ninguna			

Tabla 44. Historia de Usuario - Actualización de datos del cliente
Elaborado por: Karen Aguaysa

Historia de usuario			
Número:	036	Usuario:	Cliente
Nombre de la historia: Actualización de datos			
Prioridad en el negocio:	Alta	Puntos estimados:	4
Riesgo en el desarrollo:	Medio	Iteración asignada:	6
Descripción: Se mostrará una pantalla con los datos que puede ser actualizados de la cuenta del cliente, para sus futuras compras.			
Responsable: Karen Aguaysa			
Observaciones: Ninguna			

3.2.2.2. Estimación de historias de usuario

Una vez detalladas las historias de usuario, se prosigue a realizar la estimación de las mismas, con la finalidad de llegar a obtener un intervalo de tiempo para el desarrollo del presente proyecto, tomando en cuenta el cumplimiento de los requerimientos establecidos.

A su vez la metodología XP (Extreme Programming), permite realizar dicha estimación, a través de una técnica conocida como Planning Poker, la misma que permite dividir las tareas de usuario en actividades mucho más pequeñas, permitiendo detectar riesgos a tiempo y antes de realizar cada una de las actividades.

Es importante señalar que, para el cumplimiento de las actividades, se ha dedicado un tiempo de 3 horas diarias durante los días laborables de la semana.

Tabla 45. *Estimación de historias de Usuarios*
Elaborado por: Karen Aguaysa

Número	Historia de Usuario	Tiempo Estimado	
		Horas	Días
001	Estructura del sistema	10	3
002	Diseño de la base de datos	10	3
003	Ingreso a la aplicación web	18	6
004	Restablecer contraseña	16	5
005	Visualización de la pantalla de inicio	15	5
006	Registro de rol	9	3
007	Visualización de Roles	8	2
008	Edición de roles	9	3
009	Eliminación de roles	6	2
010	Registro de usuario	12	4
011	Visualización de usuario	8	2
012	Edición de usuarios	16	5
013	Eliminación de usuarios	6	2
014	Registro de pasillo	9	3

015	Visualización de pasillos	8	2
016	Edición de pasillos	9	3
017	Eliminación de pasillos	6	2
018	Registro de percha	9	3
019	Visualización de perchas	8	2
020	Edición de perchas	9	3
021	Eliminación de perchas	6	2
022	Registro de producto	16	5
023	Visualización de productos	12	4
024	Edición de productos	16	5
025	Eliminación de productos	8	2
026	Registro de venta	30	10
027	Visualización de ventas	20	7
028	Recepción del listado de compras	40	14
029	Ingreso a la aplicación	18	6
030	Registro de cliente	12	4
031	Catálogo de productos	40	14
032	Carrito con productos elegidos	30	10
033	Confirmación de datos	9	3
034	Confirmación de la compra	6	2
035	Envío del listado	40	14
036	Actualización de datos del cliente	12	4
Total de tiempo estimado		519	169

3.2.2.3. Plan de entrega

Para la elaboración del siguiente cronograma para el plan de entrega, se considera las iteraciones asignadas a cada una de las historias de usuario, así como la prioridad que tiene cada una de éstas, teniendo en cuenta que cada una de las iteraciones, tendrá un tiempo estimado de cuatro semanas, durante los 5 días laborables.

Tabla 46. Plan de entrega
Elaborado por: Karen Aguaysa

Número	Historia de Usuario	Tiempo		Iteración asignada					
		Horas	Días	1	2	3	4	5	6
001	Estructura del sistema	10	3	X					
002	Diseño de la base de datos	10	3	X					
003	Ingreso a la aplicación web	18	6	X					
004	Restablecer contraseña	16	5	X					
005	Visualización de la pantalla de inicio	15	5	X					
006	Registro de rol	9	3		X				
007	Visualización de Roles	8	2		X				
008	Edición de roles	9	3		X				
009	Eliminación de roles	6	2		X				
010	Registro de usuario	12	4		X				
011	Visualización de usuario	8	2		X				
012	Edición de usuarios	16	5			X			
013	Eliminación de usuarios	6	2			X			
014	Registro de pasillo	9	3			X			
015	Visualización de pasillos	8	2			X			
016	Edición de pasillos	9	3			X			
017	Eliminación de pasillos	6	2			X			
018	Registro de percha	9	3				X		
019	Visualización de perchas	8	2				X		
020	Edición de perchas	9	3				X		
021	Eliminación de perchas	6	2				X		
022	Registro de producto	16	5				X		
023	Visualización de productos	12	4				X		
024	Edición de productos	16	5					X	
025	Eliminación de productos	8	2					X	
026	Registro de venta	30	10					X	

027	Visualización de ventas	20	7	X					
028	Recepción del listado de compras	40	14					X	
029	Ingreso a la aplicación	18	6					X	
030	Registro de cliente	12	4	X					
031	Catálogo de productos	40	14						X
032	Carrito con productos elegidos	30	10						X
033	Confirmación de datos	9	3						X
034	Confirmación de la compra	6	2						X
035	Envío del listado	40	14						X
036	Actualización de datos del cliente	12	4						X

3.2.3. Fase III: Iteraciones

Una vez realizado el plan de entrega en la Tabla 46, se procede a realizar una planificación, tomando en cuenta las iteraciones y agrupándolas a través de las mismas y de acuerdo al nivel de prioridad que se establecieron previamente a través de las historias de usuario, lo que permitirá desarrollar el presente proyecto de una manera organizada a través de diferentes etapas, permitiendo hacerlo más sencillo.

Tabla 47. Planificación de iteraciones
Elaborado por: Karen Aguaysa

	Nº	Historia de Usuario	Prioridad	Riesgo	Estado de Desarrollo	Prueba
Iteración 1	001	Estructura del sistema	Alta	Alto	Completo	Aceptada
	002	Diseño de la base de datos	Alta	Medio	Completo	Aceptada
	003	Ingreso a la aplicación web	Alta	Medio	Completo	Aceptada
	004	Restablecer contraseña	Alta	Medio	Completo	Aceptada
	005	Visualización de la pantalla de inicio	Alta	Medio	Completo	Aceptada
	027	Visualización de ventas	Alta	Medio	Completo	Aceptada
	030	Registro de cliente	Alta	Medio	Completo	Aceptada

Iteración 2	006	Registro de rol	Alta	Medio	Completo	Aceptada
	007	Visualización de Roles	Alta	Medio	Completo	Aceptada
	008	Edición de roles	Alta	Medio	Completo	Aceptada
	009	Eliminación de roles	Alta	Medio	Completo	Aceptada
	010	Registro de usuario	Alta	Medio	Completo	Aceptada
	011	Visualización de usuario	Alta	Medio	Completo	Aceptada
Iteración 3	012	Edición de usuarios	Alta	Medio	Completo	Aceptada
	013	Eliminación de usuarios	Alta	Medio	Completo	Aceptada
	014	Registro de pasillo	Alta	Medio	Completo	Aceptada
	015	Visualización de pasillos	Alta	Medio	Completo	Aceptada
	016	Edición de pasillos	Alta	Medio	Completo	Aceptada
	017	Eliminación de pasillos	Alta	Medio	Completo	Aceptada
Iteración 4	018	Registro de percha	Alta	Medio	Completo	Aceptada
	019	Visualización de perchas	Alta	Medio	Completo	Aceptada
	020	Edición de perchas	Alta	Medio	Completo	Aceptada
	021	Eliminación de perchas	Alta	Medio	Completo	Aceptada
	022	Registro de producto	Alta	Medio	Completo	Aceptada
	023	Visualización de productos	Alta	Medio	Completo	Aceptada
Iteración 5	024	Edición de productos	Alta	Medio	Completo	Aceptada
	025	Eliminación de productos	Alta	Medio	Completo	Aceptada
	026	Registro de venta	Alta	Alto	Completo	Aceptada
	028	Recepción del listado de compras	Alta	Alto	Completo	Aceptada
	029	Ingreso a la aplicación	Alta	Medio	Completo	Aceptada
Iteración 6	031	Catálogo de productos	Alta	Medio	Completo	Aceptada
	032	Carrito con productos elegidos	Alta	Medio	Completo	Aceptada
	033	Confirmación de datos.	Alta	Medio	Completo	Aceptada
	034	Confirmación de la compra	Media	Bajo	Completo	Aceptada
	035	Envío del listado	Alta	Medio	Completo	Aceptada
	036	Actualización de datos del cliente	Alta	Medio	Completo	Aceptada

3.2.4. Fase IV: Puesta en Producción

Dentro de la siguiente fase se procede a dar cumplimiento de cada una de las iteraciones establecidas con sus respectivas historias de usuarios que han sido asignadas.

3.2.4.1. Iteración 1

De acuerdo a lo establecido en la Tabla 46 del plan de entregas se ha establecido el cumplimiento de las siguientes historias de usuario.

Tabla 48. Planificación de la iteración 1
Elaborado por: Karen Aguaysa

	N°	Historia de Usuario	Prioridad	Riesgo
Iteración 1	001	Estructura del sistema	Alta	Alto
	002	Diseño de la base de datos	Alta	Medio
	003	Ingreso a la aplicación web	Alta	Medio
	004	Restablecer contraseña	Alta	Medio
	005	Visualización de la pantalla de inicio	Alta	Medio
	027	Visualización de ventas	Alta	Medio
	030	Registro de cliente	Alta	Medio

N° 001: Estructura del sistema

Para establecer la estructura del presente proyecto es importante detallar las diferentes herramientas de desarrollo que fueron utilizadas:

Hardware

- 1 computadora portátil Toshiba Satellite core i7 con sistema operativo Windows 10 Pro.
- 1 dispositivo móvil Smartphone Sony X2a con sistema Android versión 9.

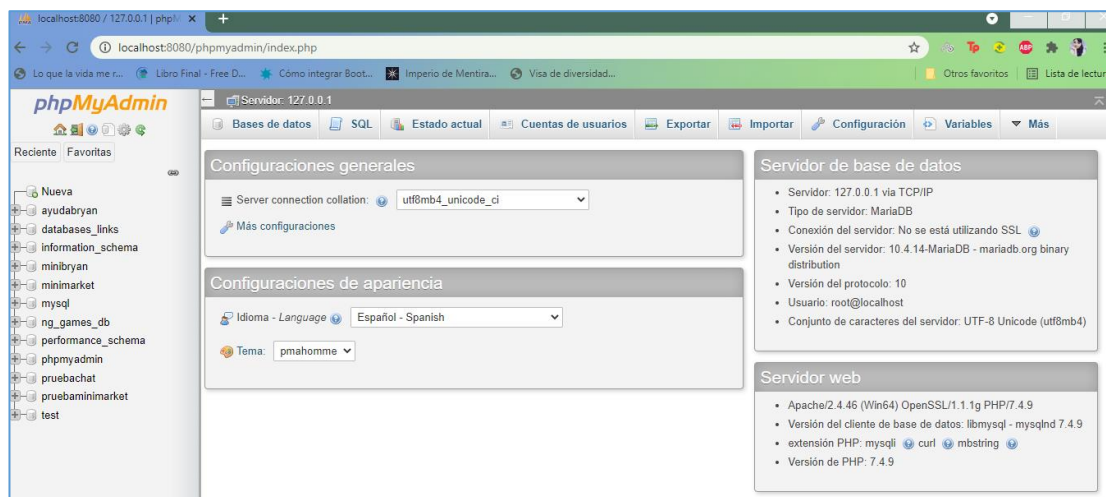
Software

Para el desarrollo del presente proyecto, se utilizaron las siguientes herramientas:

- **MySQL**

El siguiente gestor de base de datos, ha sido utilizado para el desarrollo del proyecto ya que tiene las mejores características para la implementación de la base de datos a la que tendrá conexión tanto la aplicación web como la móvil, permitiendo administrar los datos de manera confiable y precisa, además que al ser de código abierto, permite una mejor aceptación por parte de los usuarios y también presenta una gran ventaja al tener un modelo relacional, lo que permite manejar los datos de manera más organizada, manteniendo relación entre ellos para obtener precisión entre los datos.

Por otra parte, su entorno es muy amigable con los desarrolladores ya que permite crear de manera muy fácil la base de datos con sus respectivas relaciones, cuenta con unas interfaces atractivas y fáciles de manejar.



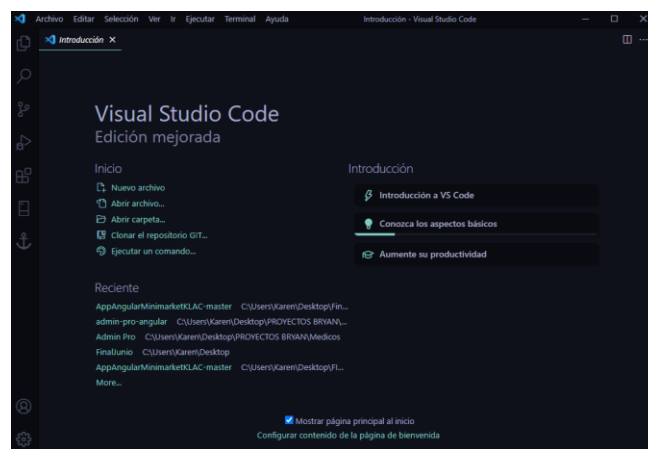
*Figura 15. Interfaz principal de MySQL
Elaborado por: Karen Aguaysa*

- **Visual Studio Code**

La siguiente herramienta ha sido considerada para ser el editor de código fuente, es decir será el programa en el cual se va a realizar la codificación del proyecto tanto para

la aplicación web, así como la aplicación móvil y todos los servicios necesarios que las mismas requieren para su ejecución.

Fue elegido como el framework de desarrollo debido a que presenta una interfaz amigable con el desarrollador, tiene la capacidad de escribir código y ejecutarlo con rapidez y facilidad, otras de sus ventajas es su forma de depurar y emitir los errores que han sido encontrados en el código durante el desarrollo, lo que facilita el trabajo del desarrollador.



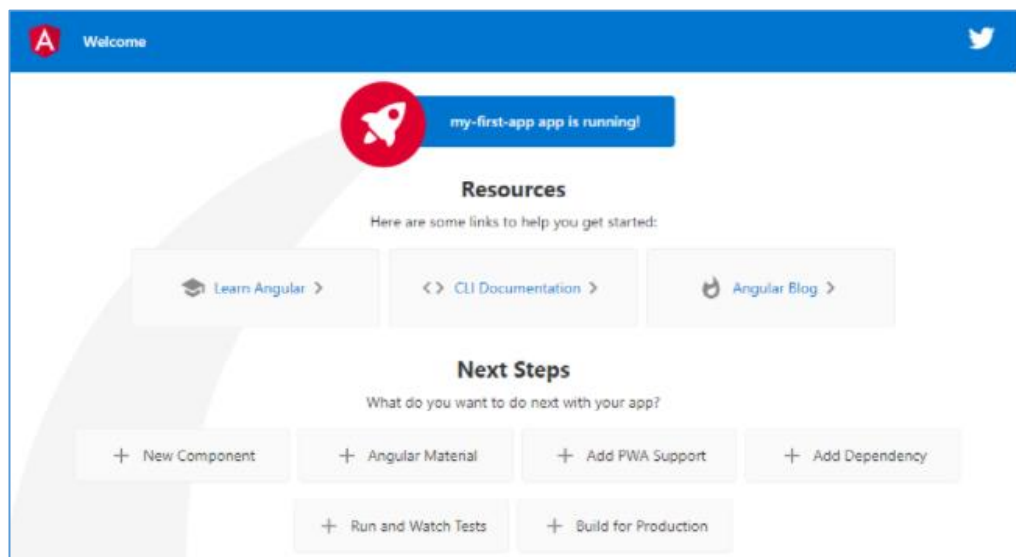
*Figura 16. Interfaz principal de Visual Studio Code
Elaborado por: Karen Aguaysa*

- **NodeJs**

El siguiente entorno ha sido escogido para la ejecución de todo aquello que vaya a ser codificado en JavaScript o a su vez en TypeScript, es decir presenta un entorno orientado a objetos que permite ejecutar las aplicaciones en tiempo real y tiene la capacidad de ejecutarlo en los ordenadores como aplicaciones independientes. Gracias todos éstos beneficios es que ha sido elegido para desarrollar el presente proyecto conjuntamente con otros implementos para complementar este entorno como son:

- Express
- Socket.io
- HttpClient
- **Angular**

Es la plataforma de desarrollo ha sido elegida como parte de la estructura del presente proyecto ya que será en ésta en donde se desarrolle el front-end de la aplicación web del sistema que será encargado de manejar el administrador del minimarket, la principal característica que posee es la reutilización de código al trabajar posteriormente con Ionic, que maneja el mismo lenguaje JavaScript y permitirá ahorrar recursos. Además, presenta una gran velocidad y rendimiento al ejecutar la aplicación desarrollada, haciéndola escalable, así también permitirá crear sus componentes de estilos que, de igual manera, pueden ser reutilizados dentro de la misma aplicación o en otras como Ionic ya que no existe diferencias entre las mismas y eso permitirá mantener un diseño similar entre las dos aplicaciones y mantener congruencia entre las mismas.



*Figura 17. Interfaz inicial del proyecto en Angular
Elaborado por: Karen Aguaysa*

- **Ionic**

Este framework de desarrollo ha sido elegido para crear la aplicación móvil del proyecto ya que presenta una variedad de características ventajosas para el desarrollo, en especial la adaptabilidad que tiene con otras plataformas de desarrollo como angular, permitiendo integrar la aplicación web con la móvil y tiene la capacidad de desplegar todas las aplicaciones en múltiples plataformas.

Nº 002: Diseño de la base de datos

En la Figura 18 se muestra el diseño de la base de datos, en un modelo relacional, en el cual se puede identificar todos los campos que formarán parte de cada una de las entidades y las características que presentan cada uno de éstos y el mismo que será implementado en el gestor de base de datos MySQL.

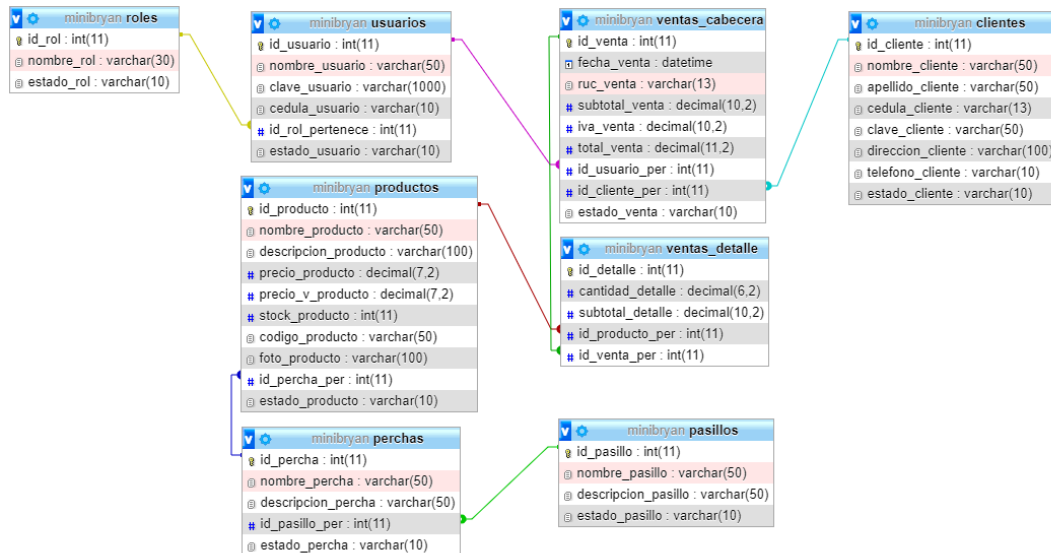


Figura 18. Modelo relacional de la base de datos
Elaborado por: Karen Aguaysa

Nº 003: Ingreso a la aplicación web

El ingreso a la aplicación web se lo realiza a través de las credenciales del administrador del sistema, o de los usuarios a los que el mismo haya dado los permisos para gestionar la aplicación. Entre las credenciales necesarias para el ingreso, se considera la cédula del usuario y su respectiva contraseña.

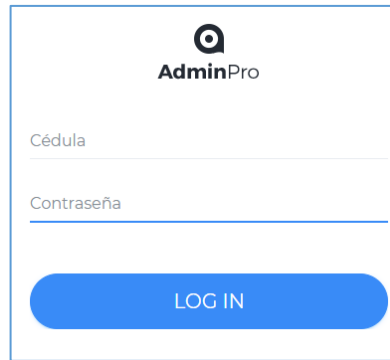


Figura 19. Ingreso a la aplicación web
Elaborado por: Karen Aguaysa

N° 004: Restablecer contraseña

El restablecimiento de la contraseña, se lo realiza a través de una pantalla en donde le solicita al usuario que ingrese el número de cédula con el que se encuentra registrado y posteriormente la actualización de la contraseña, siendo antes ésta confirmada.



Figura 20. Interfaz de la actualización de contraseña
Elaborado por: Karen Aguaysa

N° 005: Visualización de la pantalla de inicio

La pantalla de inicio se visualiza con una imagen, referente a la actividad a la que se dedica el minimarket, conjuntamente con un menú en el cual, dependiendo del rol del usuario, tendrá acceso a las diferentes funcionalidades de la aplicación web.

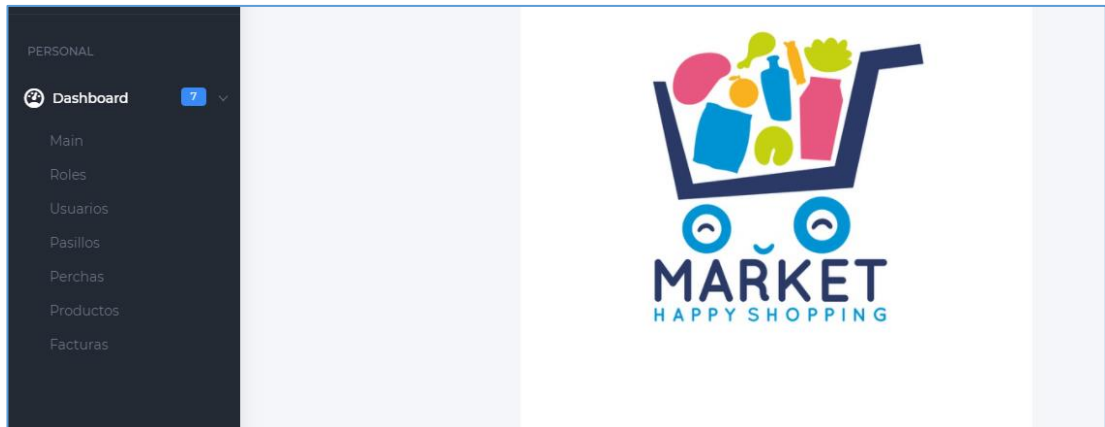


Figura 21. Interfaz de la pantalla de inicio
Elaborado por: Karen Aguaysa

Nº 027: Visualización de ventas

En esta pantalla, se pueden visualizar todas las ventas que se han realizado de manera progresiva, en la que se puede filtrar cierta información para conocimiento del administrador de las ventas registradas.

Nº 030: Registro de cliente

El registro del cliente se lleva a cabo por la aplicación móvil, a través de la cual el cliente puede ingresar una serie de datos necesarios para la creación de su cuenta para posteriormente registrado, ingresar a la aplicación y puede disfrutar de lo servicios que ésta brinda.



Figura 22. Interfaz de registro del cliente
Elaborado por: Karen Aguaysa

3.2.4.2. Iteración 2

Tabla 49. Planificación de la iteración 2

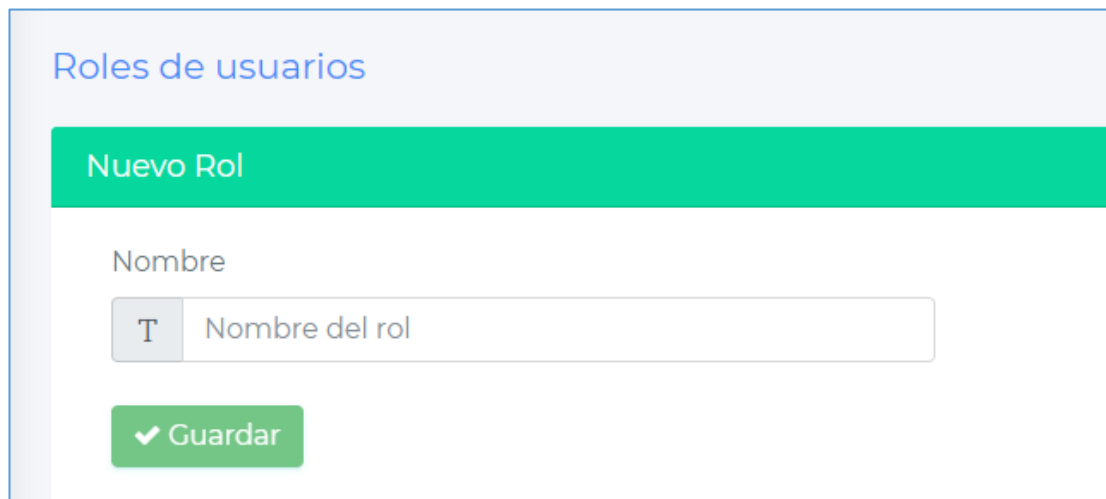
Elaborado por: Karen Aguaysa

	Nº	Historia de Usuario	Prioridad	Riesgo
Iteración 2	006	Registro de rol	Alta	Medio
	007	Visualización de Roles	Alta	Medio
	008	Edición de roles	Alta	Medio
	009	Eliminación de roles	Alta	Medio
	010	Registro de usuario	Alta	Medio
	011	Visualización de usuario	Alta	Medio

Nº 006: Registro de rol

La aplicación web cuenta con un formulario que permite registrar un nuevo rol, donde el administrador puede ingresar los datos para la creación del mismo, como se muestra en la siguiente imagen.

Entre los tipos de roles que se pueden crear, son de tipo administrativo o cajero.



The image shows a web interface titled "Roles de usuarios". Below the title is a green header bar with the text "Nuevo Rol". Underneath, there is a form with a label "Nombre" and a text input field containing "Nombre del rol". To the left of the input field is a small grey box with the letter "T". Below the input field is a green button with a white checkmark and the text "Guardar".

Figura 23. Interfaz registro de rol
Elaborado por: Karen Aguaysa

Nº 007: Visualización de Roles

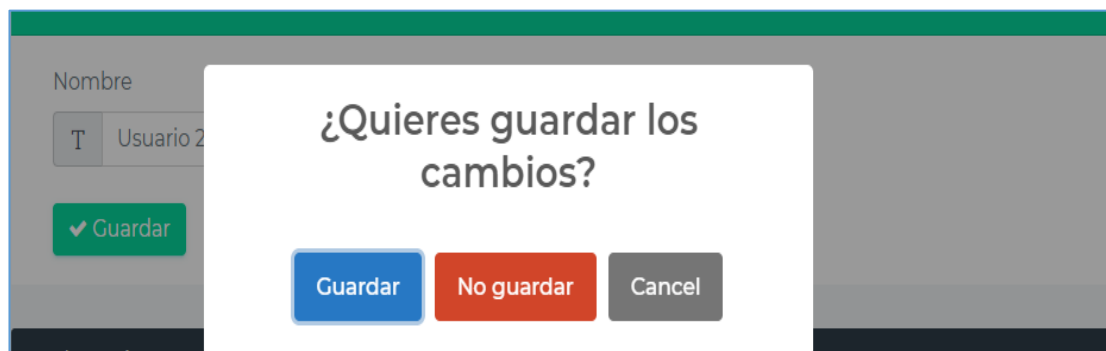
La aplicación web también cuenta con un listado de los roles que se encuentran activos dentro de la base de datos y junto a cada uno de éstos la opción de poder editarlos y también eliminarlos, en caso de requerir alguna de éstas opciones.



*Figura 24. Interfaz de visualización de roles
Elaborado por: Karen Aguaysa*

Nº 008: Edición de roles

Cuando el usuario requiera editar los roles, aquellos datos que puedan ser modificados, se activarán los mismos y pasan a ser editables, además que emitirá un mensaje de alerta cuando los cambios hayan sido ejecutados y correctamente guardados nuevamente en la base de datos.



*Figura 25. Interfaz de edición de roles
Elaborado por: Karen Aguaysa*

Nº 009: Eliminación de roles

Cuando el usuario requiera eliminar uno de los roles activos, se procede a deshabilitarlo en la base de datos, a través del botón anteriormente mencionado de eliminar, además que emitirá un mensaje de confirmación que permita al usuario estar seguro de la acción que va a llevarse a cabo.

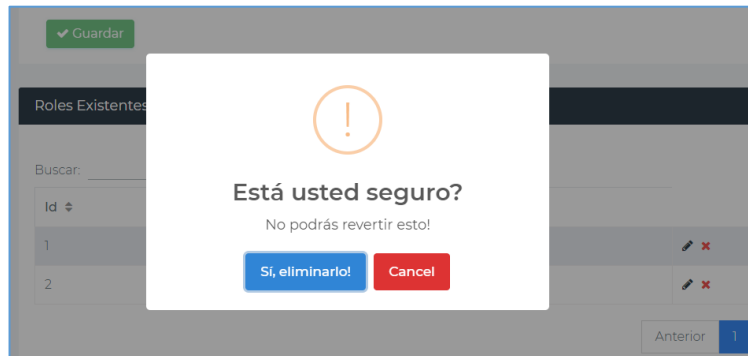


Figura 26. *Interfaz de eliminación de roles*
Elaborado por: Karen Aguaysa

Nº 010: Registro de usuario

La aplicación web cuenta con un formulario que permite registrar un nuevo usuario, donde el administrador puede ingresar los datos para la creación del mismo, como se muestra en la siguiente imagen.

A screenshot of a user registration form. The form is enclosed in a blue border and contains the following fields: "Nombre:" with a text input field containing "Nombre de usuario" and a person icon; "Cédula:" with a text input field containing "Cédula del usuario" and a document icon; "Rol:" with a dropdown menu showing "Seleccione rol" and a list icon; "Contraseña:" with a text input field containing "Contraseña del usuario" and a lock icon; and "Confirmar Contraseña:" with a text input field containing "Confirmar la contraseña" and a lock icon.

Figura 27. *Interfaz de eliminación de roles*
Elaborado por: Karen Aguaysa

Nº 011: Visualización de usuarios

La aplicación web también cuenta con un listado de los usuarios que se encuentran activos dentro de la base de datos y junto a cada uno de éstos la opción de poder editarlos y también eliminarlos, en caso de requerir alguna de ésta opciones.

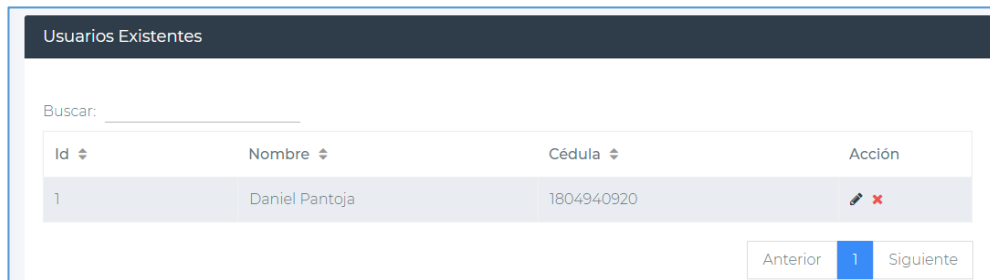


Figura 28. Interfaz de visualización de usuarios

Elaborado por: Karen Aguaysa

3.2.4.3. Iteración 3

Tabla 50. Planificación de la iteración 3

Elaborado por: Karen Aguaysa

	Nº	Historia de Usuario	Prioridad	Riesgo
Iteración 3	012	Edición de usuarios	Alta	Medio
	013	Eliminación de usuarios	Alta	Medio
	014	Registro de pasillo	Alta	Medio
	015	Visualización de pasillos	Alta	Medio
	016	Edición de pasillos	Alta	Medio
	017	Eliminación de pasillos	Alta	Medio

Nº 012: Edición de usuarios

Cuando el usuario requiera editar los usuarios existentes, aquellos datos que puedan ser modificados, se activarán los mismos y pasan a ser editables, además que emitirá un mensaje de alerta cuando los cambios hayan sido ejecutados y correctamente guardados nuevamente en la base de datos.

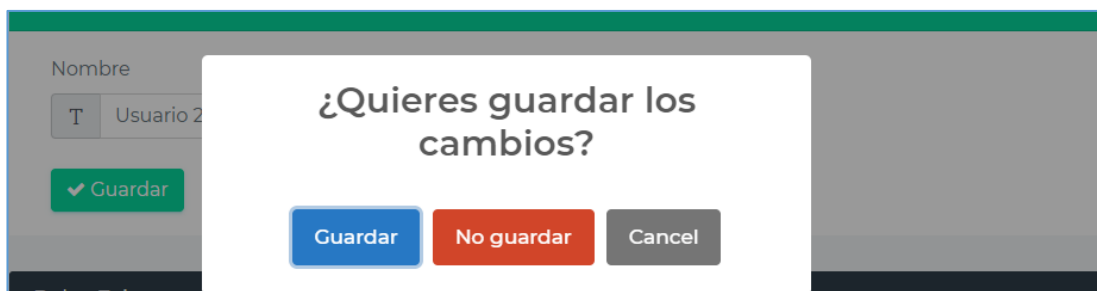


Figura 29. Interfaz de edición de usuarios
Elaborado por: Karen Aguaysa

Nº 013: Eliminación de usuarios

Cuando el usuario requiera eliminar uno de los usuarios activos, se procede a deshabilitarlo en la base de datos, a través del botón anteriormente mencionado de eliminar, además que emitirá un mensaje de confirmación que permita al usuario estar seguro de la acción que va a llevarse a cabo.

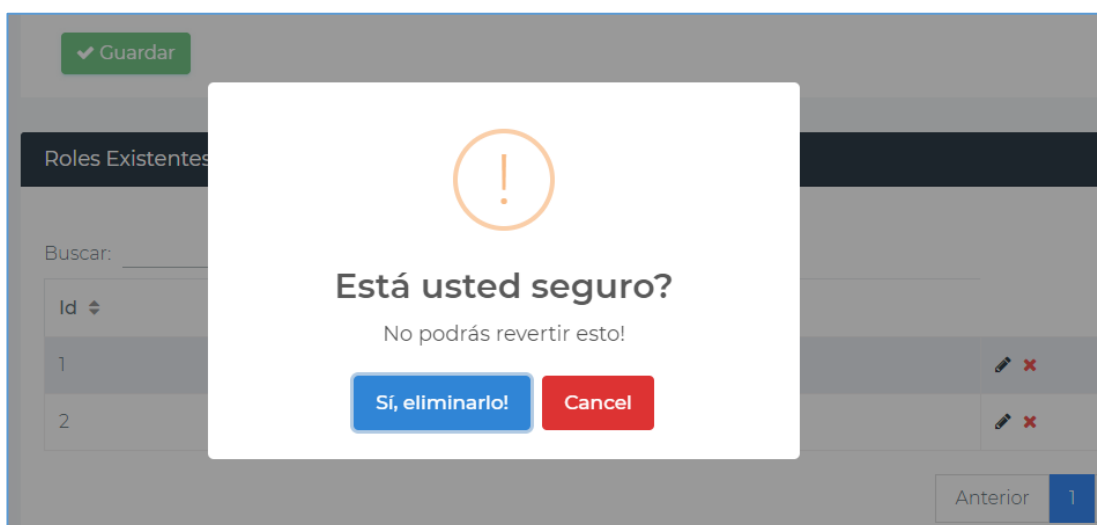


Figura 30. Interfaz de eliminación de usuarios
Elaborado por: Karen Aguaysa

Nº 014: Registro de pasillo

La aplicación web cuenta con un formulario que permite registrar un nuevo pasillo, donde el administrador puede ingresar los datos para la creación del mismo, como se muestra en la siguiente imagen.

Figura 31. Interfaz de registro de pasillo
Elaborado por: Karen Aguaysa

Nº 015: Visualización de pasillos

La aplicación web también cuenta con un listado de los pasillos que se encuentran activos dentro de la base de datos y junto a cada uno de éstos la opción de poder editarlos y también eliminarlos, en caso de requerir alguna de ésta opciones.

Id	Nombre	Descripción	Acción
1	Pasillo A	Embutidos	

Figura 32. Interfaz de visualización de pasillos
Elaborado por: Karen Aguaysa

Nº 016: Edición de pasillos

Cuando el usuario requiera editar los pasillos, aquellos datos que puedan ser modificados, se activarán los mismos y pasan a ser editables, además que emitirá un mensaje de alerta cuando los cambios hayan sido ejecutados y correctamente guardados nuevamente en la base de datos.

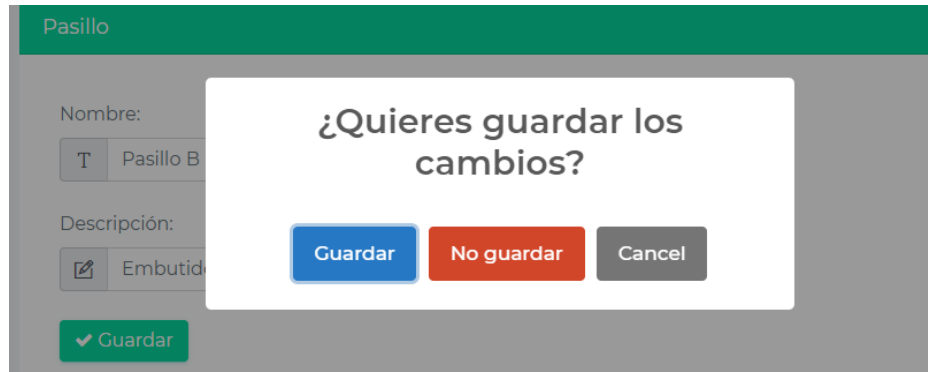


Figura 33. *Interfaz de edición de pasillos*
Elaborado por: Karen Aguaysa

N° 017: Eliminación de pasillos

Cuando el usuario requiera eliminar uno de los pasillos activos, se procede a deshabilitarlo en la base de datos, a través del botón anteriormente mencionado de eliminar, además que emitirá un mensaje de confirmación que permita al usuario estar seguro de la acción que va a llevarse a cabo.

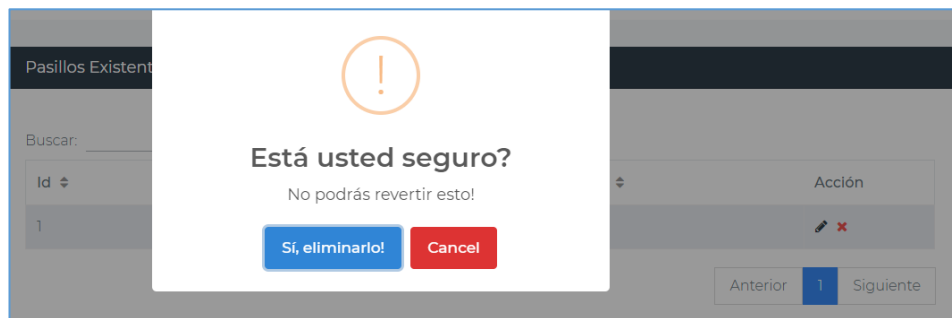


Figura 34. *Interfaz de eliminación de pasillos*
Elaborado por: Karen Aguaysa

3.2.4.4. Iteración 4

*Tabla 51. Planificación de la iteración 4
Elaborado por: Karen Aguaysa*

	Nº	Historia de Usuario	Prioridad	Riesgo
Iteración 4	018	Registro de percha	Alta	Medio
	019	Visualización de perchas	Alta	Medio
	020	Edición de perchas	Alta	Medio
	021	Eliminación de perchas	Alta	Medio
	022	Registro de producto	Alta	Medio
	023	Visualización de productos	Alta	Medio

Nº 018: Registro de percha

La aplicación web cuenta con un formulario que permite registrar una nueva percha, donde el administrador puede ingresar los datos para la creación de la misma, como se muestra en la siguiente imagen.

Registro de Perchas

Percha

Nombre:

Descripción:

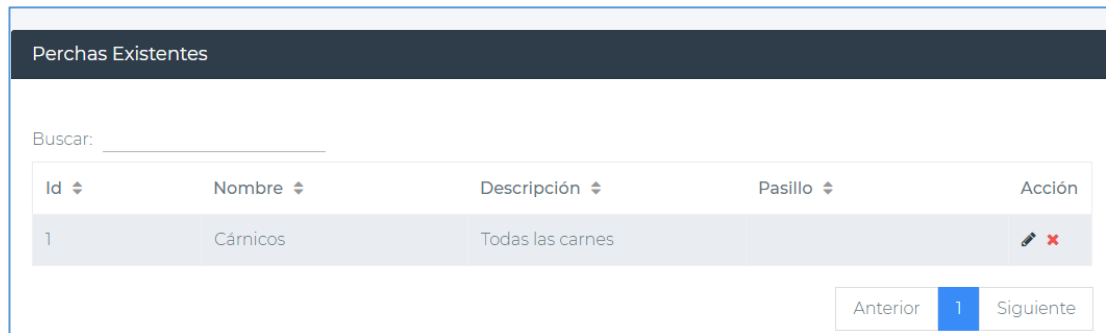
Pasillo:



✓ Guardar

*Figura 35. Interfaz de registro de percha
Elaborado por: Karen Aguaysa*

Nº 019: Visualización de perchas

La aplicación web también cuenta con un listado de las perchas que se encuentran activas dentro de la base de datos y junto a cada una de éstas la opción de poder editarlas y también eliminarlas, en caso de requerir alguna de ésta opciones.



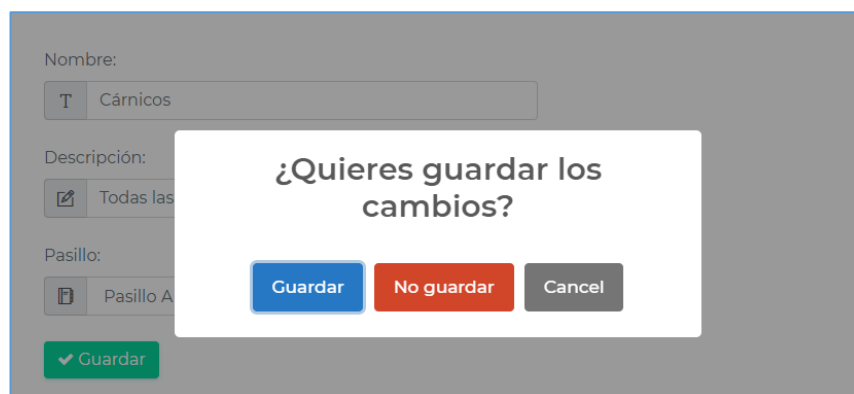
Id	Nombre	Descripción	Pasillo	Acción
1	Cárnicos	Todas las carnes		 

Anterior 1 Siguiente

Figura 36. *Interfaz de visualización de perchas*
Elaborado por: Karen Aguaysa

Nº 020: Edición de perchas

Cuando el usuario requiera editar las perchas existentes, aquellos datos que puedan ser modificados, se activarán los mismos y pasan a ser editables, además que emitirá un mensaje de alerta cuando los cambios hayan sido ejecutados y correctamente guardados nuevamente en la base de datos.



Nombre:

Descripción:

Pasillo:

¿Quieres guardar los cambios?

Figura 37. *Interfaz de edición de perchas*
Elaborado por: Karen Aguaysa

Nº 021: Eliminación de perchas

Cuando el usuario requiera eliminar uno de las perchas activas, se procede a deshabilitarla en la base de datos, a través del botón anteriormente mencionado de eliminar, además que emitirá un mensaje de confirmación que permita al usuario estar seguro de la acción que va a llevarse a cabo.

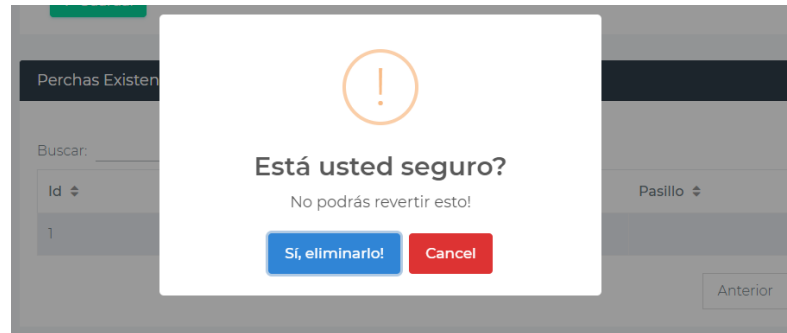


Figura 38. Interfaz de eliminación de perchas
Elaborado por: Karen Aguaysa

Nº 022: Registro de producto

La aplicación web cuenta con un formulario que permite registrar un nuevo producto, donde el administrador puede ingresar los datos para la creación del mismo, como se muestra en la siguiente imagen.

A screenshot of a web application interface titled "Registro de Productos". The form has a green header bar with the word "Producto". Below the header, there are four input fields, each with a small icon to its left. The first field is labeled "Nombre:" and contains the placeholder text "Nombre del producto". The second field is labeled "Imagen:" and is empty. The third field is labeled "Descripción:" and contains the placeholder text "Descripción del producto". The fourth field is labeled "Precio:" and contains the placeholder text "Precio del producto". Below these four fields, there are two more input fields. The first is labeled "Precio de venta:" and contains the placeholder text "Precio de venta del producto". The second field is empty. The form is enclosed in a blue border.

Figura 39. Interfaz de registro de producto
Elaborado por: Karen Aguaysa

Nº 023: Visualización de productos

La aplicación web también cuenta con un listado de los productos que se encuentran activos dentro de la base de datos y junto a cada uno de éstos la opción de poder editarlos y también eliminarlos, en caso de requerir alguna de ésta opciones.

Id	Nombre	Imagen	Precio V.	Cantidad	Código	Percha	Acción
1	Coca cola		0.6	10	7862107350470		
2	Vita Leche		0.8	50	7862107350474		
3	Papas Rufles		0.3	40	7862107354474		

[Anterior](#)
[Siguiente](#)

Figura 40. Interfaz de visualización de productos
Elaborado por: Karen Aguaysa

3.2.4.5. Iteración 5

Tabla 52. Planificación de la iteración 5
Elaborado por: Karen Aguaysa

	Nº	Historia de Usuario	Prioridad	Riesgo
Iteración 5	024	Edición de productos	Alta	Medio
	025	Eliminación de productos	Alta	Medio
	026	Registro de venta	Alta	Alto
	028	Recepción del listado de compras	Alta	Alto
	029	Ingreso a la aplicación	Alta	Medio

Nº 023: Edición de productos

Cuando el usuario requiera editar los productos, aquellos datos que puedan ser modificados, se activarán los mismos y pasan a ser editables, además que emitirá un

mensaje de alerta cuando los cambios hayan sido ejecutados y correctamente guardados nuevamente en la base de datos.

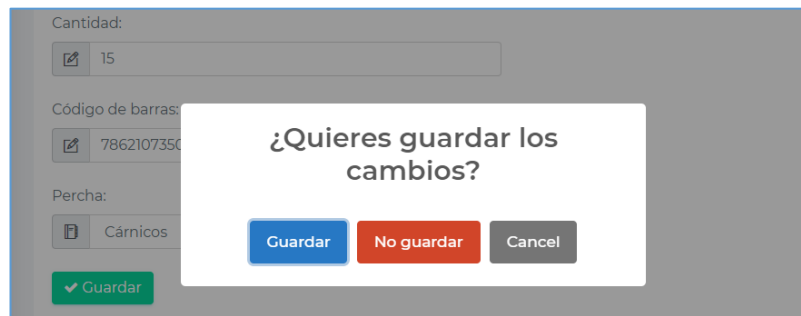


Figura 41. Interfaz de edición de productos
Elaborado por: Karen Aguaysa

Nº 023: Eliminación de productos

Cuando el usuario requiera eliminar uno de los productos activos, se procede a deshabilitarlo en la base de datos, a través del botón anteriormente mencionado de eliminar, además que emitirá un mensaje de confirmación que permita al usuario estar seguro de la acción que va a llevarse a cabo.

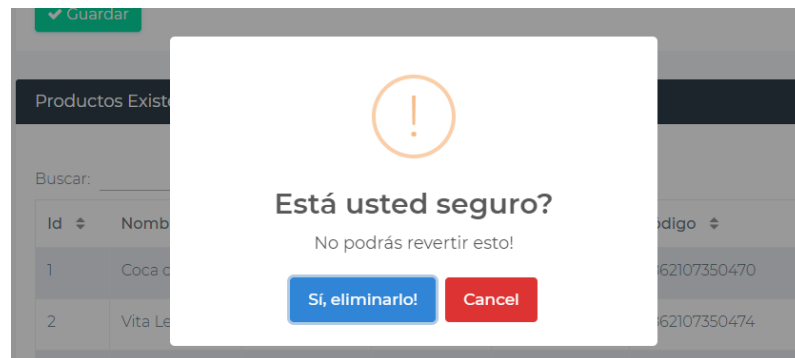
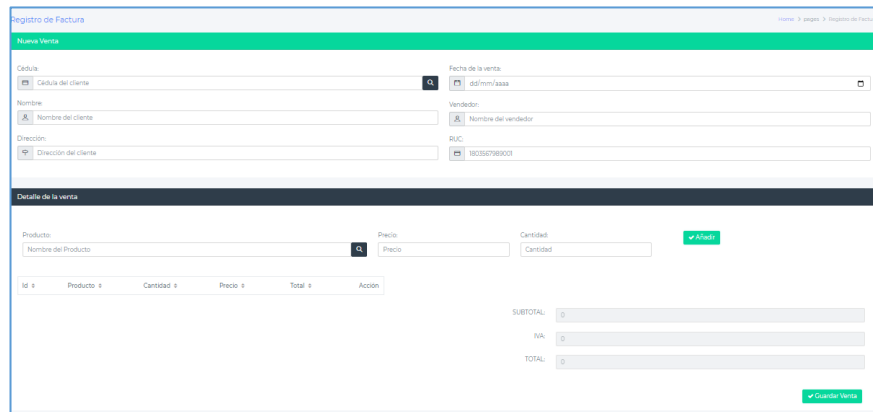


Figura 42. Interfaz de eliminación de productos
Elaborado por: Karen Aguaysa

Nº 023: Registro de venta

La aplicación web cuenta con un formulario que permite registrar una nueva venta, donde el administrador puede ingresar los datos para la creación de la misma, como se muestra en la siguiente imagen.

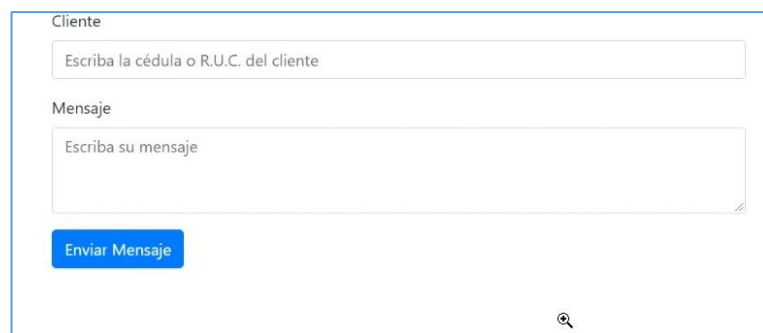


The screenshot displays the 'Registro de Factura' web interface. At the top, there is a navigation bar with 'Inicio' and 'Registros de Facturas'. Below this is a green header for 'Nueva Venta'. The form is divided into two main sections: 'Cédula' and 'Detalle de la venta'. The 'Cédula' section includes fields for 'Cédula del cliente', 'Fecha de la venta', 'Nombre', 'Dirección del cliente', 'Nombre del vendedor', and 'R.U.C.'. The 'Detalle de la venta' section features a table with columns for 'ID', 'Producto', 'Cantidad', 'Precio', 'Total', and 'Acción'. Below the table, there are input fields for 'SUBTOTAL', 'IVA', and 'TOTAL', along with a green 'Guardar Venta' button.

Figura 43. Interfaz de registro de venta
Elaborado por: Karen Aguaysa

Nº 023: Recepción del listado de compras

La aplicación web cuenta con una ventana tipo chat que permite la recepción de los mensajes de los clientes de la aplicación móvil, con la lista de los productos que se requiere al minimarket.



The screenshot shows a chat interface. At the top, it is labeled 'Cliente'. Below this is a text input field with the placeholder 'Escriba la cédula o R.U.C. del cliente'. Underneath is another text input field with the placeholder 'Escriba su mensaje'. A blue button labeled 'Enviar Mensaje' is positioned below the message input field. At the bottom right of the chat area, there is a magnifying glass icon.

Figura 44. Interfaz de recepción de listado de compras
Elaborado por: Karen Aguaysa

N° 023: Ingreso a la aplicación

El ingreso a la aplicación móvil, se lo realiza a través de credenciales, que en este caso son la cédula del usuario, conjuntamente con la contraseña establecida al momento de crear la cuenta.



Figura 45. Interfaz de ingreso a la aplicación móvil
Elaborado por: Karen Aguaysa

3.2.4.6. Iteración 6

Tabla 53. Planificación de la iteración 6
Elaborado por: Karen Aguaysa

	N°	Historia de Usuario	Prioridad	Riesgo
Iteración 6	031	Catálogo de productos	Alta	Medio
	032	Carrito con productos elegidos	Alta	Medio
	033	Confirmación de datos del comprador	Alta	Medio
	034	Confirmación de la compra	Media	Bajo
	035	Envío del listado	Alta	Medio
	037	Actualización de datos del cliente	Alta	Medio

Nº 031: Catálogo de productos

En la aplicación móvil, se cuenta con una pantalla donde se encuentran los listados de los productos existentes, con las características principales de los mismos, que el cliente necesita conocer antes de realizar la compra.

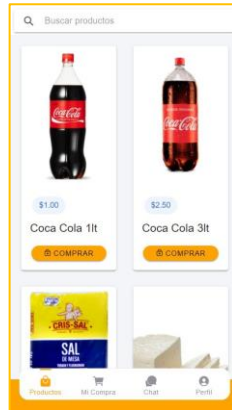


Figura 46. *Interfaz de catálogo de productos*
Elaborado por: Karen Aguaysa

Nº 032: Carrito con productos elegidos

En la aplicación móvil, una vez que el cliente ha seleccionado todos los productos que va a adquirir, existe una pantalla en donde se encuentra un listado con todos los productos elegidos por el cliente, así como con el total a cancelar de acuerdo a si el cliente desea irlos a retirar personalmente o a domicilio (lo cual genera un valor de recargo extra).

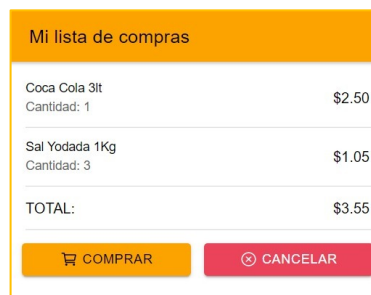


Figura 47. *Interfaz del carrito de productos*
Elaborado por: Karen Aguaysa

Nº 033: Confirmación de datos

En esta parte de la aplicación móvil, se muestran los datos de la compra para confirmar que son los productos que se desea obtener y las cantidades adecuadas, caso contrario se puede realizar modificaciones.



*Figura 48. Interfaz de confirmación de datos
Elaborado por: Karen Aguaysa*

Nº 034: Confirmación de la compra

Se emite una pantalla en donde el cliente confirma que su compra ha sido realizada y que el listado pronto estará listo para su disponibilidad.



*Figura 49. Interfaz de confirmación de la compra
Elaborado por: Karen Aguaysa*

Nº 035: Envío del listado

En la aplicación móvil, a través de una pantalla de tipo chat se enviará el listado de todos los productos que han sido adquiridos por el cliente y también establece un canal de comunicación entre el administrador y el cliente.



Figura 50. Interfaz de envío del listado
Elaborado por: Karen Aguaysa

Nº 036: Actualización de datos del cliente

En la aplicación móvil, a través de un formulario se puede recargar los datos del cliente, y ponerlos en modo editable aquellos que puedan ser modificados para que el cliente los edite en caso de ser necesario.

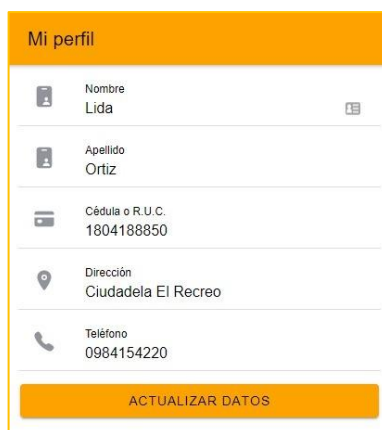


Figura 51. Interfaz de actualización de datos
Elaborado por: Karen Aguaysa

3.2.5. Codificación

3.2.5.1. Servidor

Para establecer conexión entre la base de datos de MySQL con las interfaces desarrolladas en Angula y Ionic, se ha desarrollado un servidor en NodeJs, a través del módulo mysql que se puede instalar en el mismo.

- **Renovación de token**

El siguiente método permite renovar el token una vez que haya caducado, debido al cierre de la aplicación o algún tipo de conflicto que se haya generado con la misma, permitiendo llevar a cada usuario un identificador a través del mismo para registrar el login.

```
const renovarToken = async (req, res) => {
  const { id_usuario, cedula_usuario } = req;

  try {
    const token = await generarJWT(id_usuario, cedula_usuario);
    const { nombre_rol, ...usuario } = await obtenerUsuarioPorCedula(cedula_usuario);

    res.json({
      ok: true,
      token,
      usuario,
      nombre_rol,
    });
  } catch (error) {
    if (error.code) {
      return res.status(error.code).json({
        ok: false,
        msg: error.msg,
      });
    } else {
      return res.status(400).json({
        ok: false,
        msg: 'Hubo un error',
      });
    }
  }
};
```

*Figura 52. Método de renovación de token
Elaborado por: Karen Aguaysa*

- **Gestión de roles**

El servidor cuenta con un método que permite registrar los roles existentes dentro de la aplicación, lo que posteriormente permitirá llevar un control de las funciones a las que el usuario va a tener acceso, dependiendo del rol que cumpla, permitiendo de esta manera llevar un control y organización de la información.

```
roles.route.js X
routes > roles.route.js > ...
1  const { Router } = require('express');
2  const { check } = require('express-validator');
3  const {
4    obtenerRoles,
5    obtenerRolPorId,
6    crearRol,
7    actualizarRolPorId,
8    eliminarRolPorId,
9  } = require('../controllers/roles.controller');
10 const { validarCampos } = require('../middlewares/validar_campos');
11 const {
12   validarRolExistente,
13   validarRolExistenteActualizable,
14 } = require('../middlewares/validar_campos_existentes');
15
16 const router = Router();
17
18 router.get('/', obtenerRoles);
19 router.get(
20   '/:id',
21   [check('id', 'El id no es válido').isNumeric(), validarCampos],
22   obtenerRolPorId
23 );
24 router.post(
25   '/',
26   [
27     check('nombre_rol', 'El nombre es obligatorio').notEmpty(),
28     validarCampos,
29     validarRolExistente,
30   ],
31   crearRol
32 );
33 router.put(
34   '/:id',
35   [
36     check('id', 'El id no es válido').isNumeric(),
37     check('nombre_rol', 'El nombre es obligatorio').notEmpty(),
38     validarCampos,
39     validarRolExistenteActualizable,
40   ],
41   actualizarRolPorId
42 );
43 router.delete(
44   '/:id',
45   [check('id', 'El id no es válido').isNumeric(), validarCampos],
46   eliminarRolPorId
47 );
48
49 module.exports = router;
```

*Figura 53. Métodos para la gestión de roles
Elaborado por: Karen Aguaysa*

- **Subir imágenes**

El controlador de imágenes, permite la carga de las mismas a través de un método que una vez que han sido cargadas las imágenes de los productos, se guardan en una carpeta del mismo servidor, permitiendo tener las mismas siempre disponibles para hacer una consulta, para guardar dichas imágenes se valida las diferentes extensiones y se genera el archivo, para posteriormente asígnale una ruta de acceso.

```

// Procesar la imagen...
const file = req.files.imagen;

const nombreCortado = file.name.split('.'); // wolverine.1.3.jpg
const extensionArchivo = nombreCortado[nombreCortado.length - 1];

// Validar extension
const extensionesValidas = ['png', 'jpg', 'jpeg', 'gif'];
if (!extensionesValidas.includes(extensionArchivo)) {
  return res.status(400).json({
    ok: false,
    msg: 'No es una extensión permitida',
  });
}

// Generar el nombre del archivo
const nombreArchivo = `${uidv4()}.${extensionArchivo}`;

// Path para guardar la imagen
const path = `./uploads/${tipo}/${nombreArchivo}`;

```

Figura 54. Método para procesamiento de imágenes
Elaborado por: Karen Aguaysa

También cuenta con el método de actualizar la imagen, en caso de que se requiera dicha modificación, el proceso anterior es el mismo una vez que haya sido modificada o subida la imagen, se retorna la misma para la visualización en el front-end de la aplicación.

```

// Actualizar base de datos
actualizarImagen(tipo, id, nombreArchivo)
  .then(() => {
    res.json({
      ok: true,
      msg: 'Archivo subido',
      nombreArchivo,
    });
  })
  .catch((error) =>
    res.status(error.code).json({
      ok: false,
      msg: error.msg,
    })
  );
});

const retornaImagen = (req, res = response) => {
  const tipo = req.params.tipo;
  const foto = req.params.foto;

  const pathImg = path.join(__dirname, `../uploads/productos/${foto}`);

  // imagen por defecto
  if (fs.existsSync(pathImg)) {
    res.sendFile(pathImg);
  } else {
    const pathImg = path.join(__dirname, `../uploads/NoImage.png`);
    res.sendFile(pathImg);
  }
};

```

Figura 55. Método de retorno de imagen
Elaborado por: Karen Aguaysa

- **Registro de ventas**

Los métodos para el registro de la venta, consta en crear un objeto, puesto que las ventas se generan con una cabecera y un detalle, en donde se realiza una transacción, la cual no puede ser ejecutada en caso de que exista un fallo al subir el detalle de la venta, es por esto que se realiza un método que verifique que el trigger del detalle se haya realizado para determinar que fue exitosa la transacción, caso contrario se realiza un roll-back de la venta.

```
// Obteniendo datos de los productos
for (const producto of productos) {
  let nuevoProducto = await obtenerProductoPorId(producto.id_producto);
  nuevoProducto.cantidad = producto.stock_producto;
  listaProductos = [...listaProductos, nuevoProducto];
}

// Calculando subtotal e iva
const subtotal = listaProductos.reduce(
  (valor_anterior, valor_actual) =>
    valor_anterior + valor_actual.precio_v_producto * valor_actual.cantidad,
  0
);
const iva = subtotal * Number(process.env.IVA);

// Creando el objeto de venta
const venta = {
  productos: listaProductos,
  // TODO Borrar luego de revisar que el trigger funcione
  fecha_venta: new Date(),
  // TODO poner el ruc de la tienda
  ruc_venta,
  subtotal_venta: subtotal,
  iva_venta: iva,
  // TODO Borrar luego de revisar que el trigger funcione
  total_venta: subtotal + iva,

  id_usuario_per: req.id_usuario,
  id_cliente_per,
  // TODO Cambiar a activo
  estado_venta: 'activo',
};

const ventaRealizada = await insertarVenta(venta);

return res.json({
  ok: true,
  venta: ventaRealizada,
});
} catch (error) {
  return res.status(400).json({
    ok: false,
    msg: error.msg,
  });
}
};
```

*Figura 56. Método de registro de ventas
Elaborado por: Karen Aguaysa*

Una vez que se ha verificado que la venta cumple con todos los requerimientos y se ha realizado satisfactoriamente la transacción, se procede a guardar la misma en la base de datos, a través de un arreglo que le permite guardar la cabecera principal y toda la lista de productos adquiridos por dicho cliente.

```
const insertarVenta = async(venta) => {
  let { productos, ...cabecera } = venta;
  return new Promise((resolve, reject) => {
    pool.getConnection((error, connection) => {
      if (error) {
        reject({ code: 500, msg: 'No se ha podido establecer conexión' });
        return;
      }

      console.log('Conexion establecida: ', connection.threadId);
      // Iniciando transaccion
      connection.beginTransaction((error) => {
        if (error) {
          reject({ code: 500, msg: 'No se ha podido iniciar la transacción' });
          return;
        }
        // Insertando cabecera
        connection.query(
          'INSERT INTO ventas_cabecera SET ?',
          cabecera,
          async(error, result) => {
            if (error) {
              console.log(error);
              connection.rollback(() => {
                reject({ code: 400, msg: 'No se ha podido crear la factura' });
                return;
              });
            }
          }
        );

        if (result) {
          const { insertId } = result;
          cabecera.id_venta = insertId;
          let productos_comprados = [];
          const cantidad_productos = productos.length;
          // Insertando productos al detalle
          for (const producto of productos) {
            console.log(producto);
            const venta_detalle = {
              id_venta_per: insertId,
              id_producto_per: producto.id_producto,
              cantidad_detalle: producto.cantidad,
              subtotal_detalle: producto.cantidad * producto.precio_v_producto,
              estado_detalle: 'activo',
            };
          }
        }
      }
    }
  });
}
```

*Figura 57. Método de insertar ventas
Elaborado por: Karen Aguaysa*

3.2.5.2. Aplicación web

- **Conexión de las interfaces con los servicios**

Para establecer conexión con el servidor a través de la aplicación desarrollada en Angular, se procede a crear servicios que conecten hacia el mismo, el servicio es creado con todos los métodos necesarios para crear la interacción entre el servidor y el front-end de la aplicación.

```
import { Injectable } from '@angular/core';
import { HttpClient } from '@angular/common/http';
import { environment } from '../../environments/environment';
import { Pasillo } from '../interfaces/pasillo.interface';
import { map } from 'rxjs/operators';

const urlAPI = environment.urlAPI;

@Injectable({
  providedIn: 'root',
})
export class AppMinimarketPasilloService {
  constructor(private http: HttpClient) {}

  // -----PASILLOS-----

  // crearPasillo
  crearPasillo(pasillo: Pausillo) {
    return this.http.post(`${urlAPI}/pasillos/`, pasillo);
  }

  // listar pasillos
  obtenerPasillos() {
    return this.http
      .get(`${urlAPI}/pasillos/`)
      .pipe(map((res) => res['pasillos']));
  }

  actualizarPasillo(pasillo: Pausillo) {
    return this.http.put(`${urlAPI}/pasillos/${pasillo.id_pasillo}`, pasillo);
  }

  borrarPasillo(pasillo: Pausillo) {
    return this.http.delete(`${urlAPI}/pasillos/${pasillo.id_pasillo}`);
  }
}
```

Figura 58. Método de conexión de las interfaces con los servicios
Elaborado por: Karen Aguaysa

Posteriormente se realiza la codificación en el archivo TypeScript de cada una de las interfaces, para establecer comunicación entre el servidor y la interfaz desarrollada en html, permitiendo hacer uso de todos los métodos creados en el servidor.

```
constructor(  
  private fb: FormBuilder,  
  private appMinimarketPasilloService: AppMinimarketPasilloService  
) {  
  this.cargarPasillos();  
}  
  
private cargarPasillos() {  
  this.appMinimarketPasilloService.obtenerPasillos().subscribe((resp) => {  
    console.log(resp);  
    this.listaPasillos = resp;  
  });  
}  
  
ngOnInit(): void {  
  this.formaIngresoPasillos = this.fb.group({  
    id_pasillo: [0],  
    nombre_pasillo: ['', [Validators.required]],  
    descripcion_pasillo: ['', [Validators.required]],  
    estado_pasillo: ['',  
  });  
}  
  
registrarFormularioPasillo() { ...  
}  
  
actualizarPasillo(pasillo: Pasillo) {  
  this.esActualizacion = true;  
  this.formaIngresoPasillos.setValue({  
    nombre_pasillo: pasillo.nombre_pasillo,  
    descripcion_pasillo: pasillo.descripcion_pasillo,  
    id_pasillo: pasillo.id_pasillo,  
    estado_pasillo: '',  
  });  
}  
  
eliminarPasillo(pasillo: Pasillo) {  
  Alertas.MensajeAdvertencia(  
    'Está usted seguro?',  
    'No podrás revertir esto!'  
  ).then((result) => {  
    if (result.isConfirmed) {  
      this.appMinimarketPasilloService.borrarPasillo(pasillo).subscribe(  
        (resp) => {  
          Swal.fire('Eliminado!', 'Su archivo ha sido eliminado', 'success');  
          this.cargarPasillos();  
        },  
        (err) => {  
          Alertas.MensajeError('Error', err?.error?.msg);  
        }  
      );  
    }  
  });  
}
```

*Figura 59. Métodos de conexión con la interfaz en Angular
Elaborado por: Karen Aguaysa*

- **Registro de productos**

El método de registro de productos, cumple con la funcionalidad de guardar cada nuevo producto ingresado, a través de un llamado del servidor que contiene los métodos de carga del mismo.

A través de un objeto creado, trae y lleva datos de una interface de productos para poder manipular los datos en la interfaz.

```
registrarFormularioProducto() {
  console.log(this.formaIngresoProducto);
  if (!this.esActualizacion) {
    this.appMinimarketProductoService
      .crearProducto(this.formaIngresoProducto.value)
      .subscribe(
        (resp) => {
          console.log(resp),
            Alertas.MensajeExitoso(
              'Producto creado',
              resp['producto'].nombre_producto
            );
          this.cargarProductos();
          this.formaIngresoProducto.patchValue({
            nombre_producto: '',
            descripcion_producto: '',
            precio_producto: '',
            precio_v_producto: '',
            stock_producto: '',
            codigo_producto: '',
            foto_producto: '',
            id_percha_per: '',
          });
        },
        (err) => {
          if (err?.error?.msg) {
            Alertas.MensajeError('Error', err?.error?.msg);
          }
        }
      );
  } else {
    Alertas.MensajeAdvertenciaActualizar(
      '¿Quieres guardar los cambios?'
    ).then((result) => {
      if (result.isConfirmed) {
        this.appMinimarketProductoService
          .actualizarProducto(this.formaIngresoProducto.value)
          .subscribe((resp) => {
            console.log(resp);
            Alertas.MensajeExitoso(
              'Producto actualizado',
              resp['producto'].nombre_producto
            );
            this.esActualizacion = false;
          });
      }
    });
  }
}
```

Figura 60. Método de registro de productos
Elaborado por: Karen Aguaysa

- **Registro de ventas**

Para el registro de una venta, es importante dividir el formulario en dos partes importantes, que vendrían a ser la cabecera y el detalle, por lo que se genera en un método el registro del mismo, a través de una lista que permite cargar con todos los detalles de la tabla de productos seleccionados para la venta.

```
registrarDetalle() {
  if (!this.esActualizacion) {
    const existe = this.listaDetalle.find(
      (prod) =>
        prod.id_producto === this.formaIngresoProducto.value.id_producto
    );
    if (!existe) {
      this.listaDetalle.push(this.formaIngresoProducto.value);
    } else {
      Alertas.MensajeError('Error', 'Ya esta agregado a la compra');
    }
  } else {
    this.listaDetalle = this.listaDetalle.map((prod) => {
      if (prod.id_producto === this.formaIngresoProducto.value.id_producto) {
        return this.formaIngresoProducto.value;
      } else {
        return prod;
      }
    });
    console.log(this.listaDetalle);
    this.esActualizacion = false;
  }

  this.calcularTotales();
}
```

*Figura 61. Método de registro de detalle
Elaborado por: Karen Aguaysa*

Para cargar el detalle con una lista de productos, es necesario abrir una ventana modal, que traiga consigo los datos del producto seleccionado, para lo que se realiza un método que permite la transferencia de datos de una ventana hija a una ventana padre, a través de un componente modal propio de Angular.

```

openModal() {
  const modalRef = this.modal.open(ModalComponent, {
    size: 'lg',
    scrollable: true,
  });
  modalRef.componentInstance.pasarProducto.subscribe((datos) => {
    if (datos) {
      this.formaIngresoProducto.setValue({
        id_producto: datos.id_producto,
        nombre_producto: datos.nombre_producto,
        stock_producto: datos.stock_producto,
        precio_v_producto: datos.precio_v_producto,
      });

      // console.log(datos);
      //this.actualizarProducto(datos);
    }
  });
}

```

Figura 62. Método para abrir la ventana modal
Elaborado por: Karen Aguaysa

Una vez calculados todos los datos y guardada la cabecera y un objeto detalle, se procede a guardar la factura, creando anteriormente una interface de cabecera y otra de detalle que permite llevar un conjunto de datos a través de un objeto.

```

guardarFactura() {
  let factura: Factura = {
    id_cliente_per: 1,
    ruc_venta: '1804775912001',
    productos: this.listaDetalle,
  };

  console.log(factura);
  this.appMinimarketFacturaService.crearVenta(factura).subscribe(
    (respo) => {
      console.log(respo);
    },
    (err) => {
      console.log(err);
    }
  );
}

```

Figura 63. Método para guardar factura
Elaborado por: Karen Aguaysa

- **Recepción de mensajes de los clientes**

Para la recepción de los mensajes de los clientes, se utiliza socket.io que es una librería que permite mantener contacto en tiempo real entre el servidor y el cliente, y al ser en tiempo real permite recibir los mensajes de forma inmediata en el lado del administrador, para escuchar dichos mensajes, se realiza la configuración de los métodos de emisión y de transmisión, verificando el status de la conexión del socket.

```

import { Injectable } from '@angular/core';
import { Socket } from 'ngx-socket-io';

@Injectable({
  providedIn: 'root',
})
export class WebsocketService {
  socketStatus = false;
  constructor(private socket: Socket) {
    this.checkStatus();
  }

  checkStatus() {
    this.socket.on('connect', () => {
      console.log('Conectado al servidor');
      this.socketStatus = true;
    });

    this.socket.on('disconnect', () => {
      console.log('Desconectado al servidor');
      this.socketStatus = false;
    });
  }

  emit(evento: string, payload?: any, callback?: Function) {
    this.socket.emit(evento, payload, callback);
  }

  listen(evento: string) {
    return this.socket.fromEvent(evento);
  }
}

```

Figura 64. Método de creación del socket
Elaborado por: Karen Aguaysa

Posteriormente, se crea un servicio en el cual existen tres métodos el de enviar mensaje, que es encargado de establecer conexión con el cliente, el de recibir el mensaje y el de obtener el pedido para registrar la lista en la parte del front-end de la aplicación.

```

import { Injectable } from '@angular/core';
import { WebsocketService } from './websocket.service';

@Injectable({
  providedIn: 'root',
})
export class ChatService {
  constructor(public wsService: WebsocketService) {}

  sendMessage(remite: string, destinatario: string, mensaje: string) {
    const payload = {
      remitente,
      destinatario,
      mensaje,
    };

    this.wsService.emit('mensaje', payload);
  }

  getMessages() {
    return this.wsService.listen('mensaje-nuevo');
  }

  getPedidos() {
    return this.wsService.listen('pedido-nuevo');
  }
}

```

Figura 65. Método del llamado al socket
Elaborado por: Karen Aguaysa

3.2.5.3. Aplicación móvil

- **Visualización de productos**

Para la visualización del catálogo de productos, se realiza un método que recorre toda la lista de la base de datos para obtener los productos que se encuentran activos dentro de la base de datos, permitiendo analizar las características de cada uno de éstos y traerlos al front-end de la aplicación.

```
6  @Component({
7    selector: 'app-tab1',
8    templateUrl: 'tab1.page.html',
9    styleUrls: ['tab1.page.scss'],
10 })
11 export class Tab1Page {
12   products: Producto[] = [];
13   comprados: boolean[] = [];
14   constructor(
15     private productService: ProductoService,
16     private purchaseService: PurchaseService
17   ) {
18     this.loadProducts();
19   }
20
21   loadProducts() {
22     this.productService.getProducts().subscribe(
23       (resp) => {
24         this.products = resp.productos;
25         console.log(this.products);
26       },
27       (error) => {
28         console.log(error);
29       }
30     );
31   }
32
33   ionViewWillEnter() {
34     this.comprados = [];
35     this.products.map((product) => {
36       const existe = this.purchaseService.purchase.some(
37         (p) => p.id_producto === product.id_producto
38       );
39       this.comprados.push(existe);
40     });
41   }
42 }
43
```

*Figura 66. Método de visualización del catálogo de productos
Elaborado por: Karen Aguaysa*

- **Envío de mensajes**

Para el envío de mensajes a través de la aplicación móvil, se utiliza el servidor anteriormente creado con la librería socket.io y se procede a realizar una conexión entre la interfaz y el mismo, para empezar a emitir los mensajes.

Aquí se ejecutan dos métodos, el de enviar mensaje para mantener conexión con el socket y se envíe la cabecera del mensaje y el de enviar pedido, para que sea el cuerpo del mensaje con la lista de productos a comprar, trabajando el mismo como un objeto.

```
mensajes: any[] = [];  
constructor(private chatService: ChatService) {}  
  
ngOnInit(): void {  
  this.mensajes$ = this.chatService  
    .getMessages()  
    .subscribe((msg: Mensaje) => {  
      if (msg.destinatario === this.nombre || msg.remitente === this.nombre) {  
        console.log(this.mensajes);  
        this.mensajes.push(msg);  
      }  
      // setTimeout(() => {  
        //   this.elemento.scrollTop = this.elemento.scrollHeight;  
        // }, 50);  
    });  
}  
  
ngOnDestroy() {  
  this.mensajes$.unsubscribe();  
}  
  
enviarMensaje() {  
  // if (this.texto.trim() === '') return;  
  this.chatService.sendMessage(this.nombre, 'Admin', this.texto);  
  this.texto = '';  
}  
  
enviarPedido() {  
  const pedido: Pedido = {  
    id_cliente: 1,  
    nombre_cliente: this.nombre,  
    direccion_cliente: 'Ambato',  
    fecha_envio: new Date(),  
    remitente: 'Admin',  
    productos: [  
      {  
        id_producto: 1,  
        nombre_producto: 'Ruffles',  
      },  
      {  
        id_producto: 2,  
        nombre_producto: 'Doritos',  
      },  
    ],  
  };  
  this.chatService.sendPedido(pedido);  
}
```

Figura 67. Método de envío de mensajes
Elaborado por: Karen Aguaysa

3.2.6. Fase V: Pruebas

3.2.6.1. Pruebas de aceptación

Las pruebas de aceptación se realizan para demostrar que el sistema, cumple con todos los requerimientos que anteriormente fueron especificados, permitiendo verificar el estado del mismo para posteriormente ponerlo en producción, de ésta manera se puede analizar el buen funcionamiento de la aplicación y que su rendimiento sea óptimo para los usuarios finales, que son los encargados de expresar si la prueba es satisfactoria y satisface las necesidades de los mismos.

*Tabla 54. Prueba de aceptación 001
Elaborado por: Karen Aguaysa*

PRUEBA DE ACEPTACIÓN	
Número:	001
Historia de usuario N°:	003
Nombre: Ingreso a la aplicación web.	
Descripción: Se mostrará una pantalla, donde el usuario pueda ingresar sus credenciales, en este caso cédula y contraseña para poder acceder a las funciones de la aplicación.	
Condiciones de ejecución: Ninguna.	
Interfaz: La pantalla contiene dos campos, en el primero para ingresar la cédula y en el segundo para el ingreso de la contraseña y un botón que permite ingresar a la aplicación web.	
Resultado esperado: En caso de que las credenciales sean correctas, el usuario ingresará a la pantalla de home, caso contrario será redirigido al formulario de ingreso de credenciales.	
Evaluación de la prueba: Satisfactoria.	

*Tabla 55. Prueba de aceptación 002
Elaborado por: Karen Aguaysa*

PRUEBA DE ACEPTACIÓN	
Número:	002
Historia de usuario N°:	004
Nombre: Restablecer contraseña.	
Descripción: Se mostrará una pantalla, donde el usuario pueda ingresar sus nuevas credenciales, en este caso cédula y la nueva contraseña, también cuenta con un campo de verificación de la misma.	
Condiciones de ejecución: Encontrarse en el perfil de administrador del sistema.	
Interfaz: La pantalla contiene dos campos, en el primero para ingresar la cédula, el segundo para el ingreso de la contraseña y en el tercero para la confirmación de la misma, un botón que permite guardar los cambios.	
Resultado esperado: En caso de que el proceso sea correcto, el administrados habrá realizado los cambios y tendrá acceso a las funciones del sistema.	
Evaluación de la prueba: Satisfactoria.	

*Tabla 56. Prueba de aceptación 003
Elaborado por: Karen Aguaysa*

PRUEBA DE ACEPTACIÓN	
Número:	003
Historia de usuario N°:	006
Nombre: Registro de rol.	
Descripción: Se mostrará una pantalla, donde el usuario pueda ingresar los datos para crear un rol, en este caso el nombre del mismo.	
Condiciones de ejecución: Encontrarse en el perfil de administrador del sistema.	
Interfaz: La pantalla contiene un campo en el cual se puede ingresar el nombre del nuevo rol a crear.	
Resultado esperado: En caso de que el proceso sea correcto, el administrados habrá registrado un nuevo rol dentro del sistema, el cual no puede ser administrador.	
Evaluación de la prueba: Satisfactoria.	

*Tabla 57. Prueba de aceptación 004
Elaborado por: Karen Aguaysa*

PRUEBA DE ACEPTACIÓN	
Número:	004
Historia de usuario N°:	007
Nombre: Visualización de Roles.	
Descripción: Se mostrará en la pantalla, una tabla donde consten todos los roles que se encuentren activos en la base de datos.	
Condiciones de ejecución: Encontrarse en el perfil de administrador del sistema.	
Interfaz: La pantalla contiene una tabla con tres campos, el primero con la numeración del rol, el segundo con el nombre y el tercero con los botones de acciones como editar y eliminar.	
Resultado esperado: Al visualizar dicha pantalla, la tabla se encuentra cargada con todos los roles activos en la base de datos y se actualiza constantemente.	
Evaluación de la prueba: Satisfactoria.	

*Tabla 58. Prueba de aceptación 005
Elaborado por: Karen Aguaysa*

PRUEBA DE ACEPTACIÓN	
Número:	005
Historia de usuario N°:	008
Nombre: Edición de Roles.	
Descripción: Se mostrará en la pantalla, con la tabla de roles en donde se encontrará un ícono de edición, el mismo que al dar click, cargará los datos para ser editados.	
Condiciones de ejecución: Encontrarse en el perfil de administrador del sistema.	
Interfaz: La pantalla contiene tres campos, el primero con la numeración del rol, el segundo con el nombre y el tercero con la acción de edición.	
Resultado esperado: En caso de que el proceso sea correcto, el administrados habrá realizado los cambios necesarios en el rol seleccionado	
Evaluación de la prueba: Satisfactoria.	

Tabla 59. Prueba de aceptación 006
Elaborado por: Karen Aguaysa

PRUEBA DE ACEPTACIÓN	
Número:	006
Historia de usuario N°:	009
Nombre: Eliminación de roles.	
Descripción: Se mostrará en la pantalla, con la tabla de roles en donde se encontrará un ícono de eliminación, el mismo que al dar click, aparecerá un mensaje de alerta para confirmar la acción y eliminar el rol seleccionado.	
Condiciones de ejecución: Encontrarse en el perfil de administrador del sistema.	
Interfaz: La pantalla contiene tres campos, el primero con la numeración del rol, el segundo con el nombre y el tercero con la acción de eliminación.	
Resultado esperado: En caso de que el administrador haya dado click sobre el botón eliminar, y en la alerta enviada confirme la acción, el rol seleccionado será eliminado.	
Evaluación de la prueba: Satisfactoria.	

Tabla 60. Prueba de aceptación 007
Elaborado por: Karen Aguaysa

PRUEBA DE ACEPTACIÓN	
Número:	007
Historia de usuario N°:	010
Nombre: Registro de usuario.	
Descripción: Se mostrará una pantalla, donde el administrador pueda ingresar todos los datos necesarios para el nuevo usuario.	
Condiciones de ejecución: Encontrarse en el perfil de administrador del sistema.	
Interfaz: La pantalla contiene los siguientes campos: <ul style="list-style-type: none"> • Nombre • Cédula • Rol de usuario • Contraseña • Confirmación de la contraseña 	
Resultado esperado: En caso de que el proceso sea correcto, el administrador habrá registrado un nuevo usuario dentro del sistema.	
Evaluación de la prueba: Satisfactoria.	

Tabla 61. Prueba de aceptación 008
Elaborado por: Karen Aguaysa

PRUEBA DE ACEPTACIÓN	
Número:	008
Historia de usuario N°:	011
Nombre: Visualización de Usuarios.	
Descripción: Se mostrará en la pantalla, una tabla donde consten todos los usuarios que se encuentren activos en la base de datos.	
Condiciones de ejecución: Encontrarse en el perfil de administrador del sistema.	
Interfaz: La pantalla contiene una tabla con cuatro campos, el primero con la numeración del usuario, el segundo con el nombre, el tercero con el tipo de rol que desempeña y el cuarto con los botones de acciones como editar y eliminar.	
Resultado esperado: Al visualizar dicha pantalla, la tabla se encuentra cargada con todos los usuarios activos en la base de datos y se actualiza constantemente.	
Evaluación de la prueba: Satisfactoria.	

Tabla 62. Prueba de aceptación 009
Elaborado por: Karen Aguaysa

PRUEBA DE ACEPTACIÓN	
Número:	009
Historia de usuario N°:	012
Nombre: Edición de Usuarios.	
Descripción: Se mostrará en la pantalla, con la tabla de usuarios, en donde se encontrará un ícono de edición, el mismo que al dar click, cargará los datos para ser editados, con excepción de la contraseña del mismo.	
Condiciones de ejecución: Encontrarse en el perfil de administrador del sistema.	
Interfaz: La pantalla contiene cuatro campos, el primero con el nombre, el segundo con el apellido, el tercero con la cédula y el cuarto con el rol que desempeña.	
Resultado esperado: En caso de que el proceso sea correcto, el administrados habrá realizado los cambios necesarios en el usuario seleccionado	
Evaluación de la prueba: Satisfactoria.	

Tabla 63. Prueba de aceptación 010
Elaborado por: Karen Aguaysa

PRUEBA DE ACEPTACIÓN	
Número:	010
Historia de usuario N°:	013
Nombre: Eliminación de roles.	
Descripción: Se mostrará en la pantalla, con la tabla de usuarios en donde se encontrará un ícono de eliminación, el mismo que al dar click, aparecerá un mensaje de alerta para confirmar la acción y eliminar el usuario seleccionado.	
Condiciones de ejecución: Encontrarse en el perfil de administrador del sistema.	
Interfaz: La pantalla contiene una tabla con cuatro campos, el primero con la numeración del usuario, el segundo con el nombre, el tercero con el tipo de rol que desempeña y el cuarto con el botón de eliminación.	
Resultado esperado: En caso de que el administrador haya dado click sobre el botón eliminar, y en la alerta enviada confirme la acción, el usuario seleccionado será eliminado.	
Evaluación de la prueba: Satisfactoria.	

Tabla 64. Prueba de aceptación 011
Elaborado por: Karen Aguaysa

PRUEBA DE ACEPTACIÓN	
Número:	011
Historia de usuario N°:	014
Nombre: Registro de pasillo.	
Descripción: Se mostrará una pantalla, donde el administrador pueda ingresar todos los datos necesarios para el nuevo pasillo.	
Condiciones de ejecución: Encontrarse en el perfil de administrador del sistema.	
Interfaz: La pantalla contiene los siguientes campos: <ul style="list-style-type: none"> • Nombre • Descripción del pasillo 	
Resultado esperado: En caso de que el proceso sea correcto, el administrador habrá registrado un nuevo pasillo dentro del sistema.	
Evaluación de la prueba: Satisfactoria.	

Tabla 65. Prueba de aceptación 012
Elaborado por: Karen Aguaysa

PRUEBA DE ACEPTACIÓN	
Número:	012
Historia de usuario N°:	015
Nombre: Visualización de Pasillos.	
Descripción: Se mostrará en la pantalla, una tabla donde consten todos los pasillos que se encuentren activos en la base de datos.	
Condiciones de ejecución: Encontrarse en el perfil de administrador del sistema.	
Interfaz: La pantalla contiene una tabla con cuatro campos, el primero con la numeración del pasillo, el segundo con el nombre, el tercero con la descripción del mismo y el cuarto con los botones de acciones como editar y eliminar.	
Resultado esperado: Al visualizar dicha pantalla, la tabla se encuentra cargada con todos los pasillos activos en la base de datos y se actualiza constantemente.	
Evaluación de la prueba: Satisfactoria.	

Tabla 66. Prueba de aceptación 013
Elaborado por: Karen Aguaysa

PRUEBA DE ACEPTACIÓN	
Número:	013
Historia de usuario N°:	016
Nombre: Edición de Pasillos.	
Descripción: Se mostrará en la pantalla, con la tabla de pasillos, en donde se encontrará un ícono de edición, el mismo que al dar click, cargará los datos para ser editados.	
Condiciones de ejecución: Encontrarse en el perfil de administrador del sistema.	
Interfaz: La pantalla contiene dos campos, el primero con el nombre, el segundo con la descripción del pasillo.	
Resultado esperado: En caso de que el proceso sea correcto, el administrados habrá realizado los cambios necesarios en el pasillo seleccionado	
Evaluación de la prueba: Satisfactoria.	

Tabla 67. Prueba de aceptación 014
Elaborado por: Karen Aguaysa

PRUEBA DE ACEPTACIÓN	
Número:	014
Historia de usuario N°:	017
Nombre: Eliminación de pasillos.	
Descripción: Se mostrará en la pantalla, con la tabla de pasillos en donde se encontrará un ícono de eliminación, el mismo que al dar click, aparecerá un mensaje de alerta para confirmar la acción y eliminar el pasillo seleccionado.	
Condiciones de ejecución: Encontrarse en el perfil de administrador del sistema.	
Interfaz: La pantalla contiene una tabla con cuatro campos, el primero con la numeración del usuario, el segundo con el nombre, el tercero con la descripción del mismo y el cuarto con el botón de eliminación.	
Resultado esperado: En caso de que el administrador haya dado click sobre el botón eliminar, y en la alerta enviada confirme la acción, el pasillo seleccionado será eliminado.	
Evaluación de la prueba: Satisfactoria.	

Tabla 68. Prueba de aceptación 015
Elaborado por: Karen Aguaysa

PRUEBA DE ACEPTACIÓN	
Número:	015
Historia de usuario N°:	018
Nombre: Registro de percha.	
Descripción: Se mostrará una pantalla, donde el administrador pueda ingresar todos los datos necesarios para la nueva percha.	
Condiciones de ejecución: Encontrarse en el perfil de administrador del sistema.	
Interfaz: La pantalla contiene los siguientes campos: <ul style="list-style-type: none"> • Nombre • Descripción • Pasillo al que pertenece 	
Resultado esperado: En caso de que el proceso sea correcto, el administrador habrá registrado una nueva percha dentro del sistema.	
Evaluación de la prueba: Satisfactoria.	

Tabla 69. Prueba de aceptación 016
Elaborado por: Karen Aguaysa

PRUEBA DE ACEPTACIÓN	
Número:	016
Historia de usuario N°:	019
Nombre: Visualización de Perchas.	
Descripción: Se mostrará en la pantalla, una tabla donde consten todas las perchas que se encuentren activas en la base de datos.	
Condiciones de ejecución: Encontrarse en el perfil de administrador del sistema.	
Interfaz: La pantalla contiene una tabla con cinco campos, el primero con la numeración del pasillo, el segundo con el nombre, el tercero con la descripción del mismo y el cuarto con el pasillo al que pertenece y el quinto con los botones de acciones como editar y eliminar.	
Resultado esperado: Al visualizar dicha pantalla, la tabla se encuentra cargada con todas las perchas activas en la base de datos y se actualiza constantemente.	
Evaluación de la prueba: Satisfactoria.	

Tabla 70. Prueba de aceptación 017
Elaborado por: Karen Aguaysa

PRUEBA DE ACEPTACIÓN	
Número:	017
Historia de usuario N°:	020
Nombre: Edición de Perchas.	
Descripción: Se mostrará en la pantalla, con la tabla de perchas, en donde se encontrará un ícono de edición, el mismo que al dar click, cargará los datos para ser editados.	
Condiciones de ejecución: Encontrarse en el perfil de administrador del sistema.	
Interfaz: La pantalla contiene tres campos, el primero con el nombre, el segundo con la descripción de la percha y el tercero con el pasillo al que pertenece.	
Resultado esperado: En caso de que el proceso sea correcto, el administrados habrá realizado los cambios necesarios en la percha seleccionada.	
Evaluación de la prueba: Satisfactoria.	

Tabla 71. Prueba de aceptación 018
Elaborado por: Karen Aguaysa

PRUEBA DE ACEPTACIÓN	
Número:	018
Historia de usuario N°:	021
Nombre: Eliminación de Perchas.	
Descripción: Se mostrará en la pantalla, con la tabla de pasillos en donde se encontrará un ícono de eliminación, el mismo que al dar click, aparecerá un mensaje de alerta para confirmar la acción y eliminar el pasillo seleccionado.	
Condiciones de ejecución: Encontrarse en el perfil de administrador del sistema.	
Interfaz: La pantalla contiene una tabla con cuatro campos, el primero con la numeración de la percha, el segundo con el nombre, el tercero con la descripción y el cuarto con el botón de eliminación.	
Resultado esperado: En caso de que el administrador haya dado click sobre el botón eliminar, y en la alerta enviada confirme la acción, la percha seleccionada será eliminada.	
Evaluación de la prueba: Satisfactoria.	

Tabla 72. Prueba de aceptación 019
Elaborado por: Karen Aguaysa

PRUEBA DE ACEPTACIÓN	
Número:	019
Historia de usuario N°:	022
Nombre: Registro de Producto.	
Descripción: Se mostrará una pantalla, donde el administrador pueda ingresar todos los datos necesarios para el nuevo producto.	
Condiciones de ejecución: Encontrarse en el perfil de administrador del sistema.	
Interfaz: La pantalla contiene los siguientes campos: <ul style="list-style-type: none"> • Nombre • Descripción • Precio • Precio de venta • Cantidad • Código de barras • Percha a la que pertenece 	
Resultado esperado: En caso de que el proceso sea correcto, el administrador habrá registrado un nuevo producto dentro del sistema.	
Evaluación de la prueba: Satisfactoria.	

Tabla 73. Prueba de aceptación 020
Elaborado por: Karen Aguaysa

PRUEBA DE ACEPTACIÓN	
Número:	020
Historia de usuario N°:	023
Nombre: Visualización de Productos.	
Descripción: Se mostrará en la pantalla, una tabla donde consten todos los productos que se encuentren activos en la base de datos.	
Condiciones de ejecución: Encontrarse en el perfil de administrador del sistema.	
Interfaz: La pantalla contiene una tabla con seis campos, el primero con la numeración del producto, el segundo con el nombre, el tercero con el precio de venta, el cuarto con la cantidad de stock existente, el quinto con la percha donde se encuentra y el sexto con los botones de acciones como editar y eliminar.	
Resultado esperado: Al visualizar dicha pantalla, la tabla se encuentra cargada con todos los productos activos en la base de datos y se actualiza constantemente.	
Evaluación de la prueba: Satisfactoria.	

Tabla 74. Prueba de aceptación 021
Elaborado por: Karen Aguaysa

PRUEBA DE ACEPTACIÓN	
Número:	021
Historia de usuario N°:	024
Nombre: Edición de Productos.	
Descripción: Se mostrará en la pantalla, con la tabla de productos, en donde se encontrará un ícono de edición, el mismo que al dar click, cargará los datos para ser editados.	
Condiciones de ejecución: Encontrarse en el perfil de administrador del sistema.	
Interfaz: La pantalla contiene una tabla con seis campos, el primero con la numeración del producto, el segundo con el nombre, el tercero con el precio de venta, el cuarto con la cantidad de stock existente, el quinto con la percha donde se encuentra.	
Resultado esperado: En caso de que el proceso sea correcto, el administrados habrá realizado los cambios necesarios en el producto seleccionado	
Evaluación de la prueba: Satisfactoria.	

Tabla 75. Prueba de aceptación 022
Elaborado por: Karen Aguaysa

PRUEBA DE ACEPTACIÓN	
Número:	022
Historia de usuario N°:	025
Nombre: Eliminación de Productos.	
Descripción: Se mostrará en la pantalla, con la tabla de productos en donde se encontrará un ícono de eliminación, el mismo que al dar click, aparecerá un mensaje de alerta para confirmar la acción y eliminar el producto seleccionado.	
Condiciones de ejecución: Encontrarse en el perfil de administrador del sistema.	
Interfaz: La pantalla contiene una tabla con seis campos, el primero con la numeración del producto, el segundo con el nombre, el tercero con el precio de venta, el cuarto con la cantidad de stock existente, el quinto con la percha donde se encuentra y el sexto con el botón de eliminación.	
Resultado esperado: En caso de que el administrador haya dado click sobre el botón eliminar, y en la alerta enviada confirme la acción, el producto seleccionado será eliminado.	
Evaluación de la prueba: Satisfactoria.	

Tabla 76. Prueba de aceptación 023
Elaborado por: Karen Aguaysa

PRUEBA DE ACEPTACIÓN	
Número:	023
Historia de usuario N°:	026
Nombre: Registro de venta.	
Descripción: Se mostrará en la pantalla, donde el usuario pueda ingresar todos los datos necesarios para generar una nueva venta.	
Condiciones de ejecución: Ninguna.	
Interfaz: La pantalla contiene los siguientes campos: <ul style="list-style-type: none"> • Cédula del cliente • Nombre • Dirección • Fecha de la venta • Vendedor • RUC Y una tabla en donde se registrará el detalle de la venta, con los siguientes datos. <ul style="list-style-type: none"> • Producto • Precio 	

- Cantidad
- Total

Resultado esperado:

En caso de que el administrador haya dado click sobre el botón guardar, la cabecera y el detalle de la venta serán guardados en la base de datos.

Evaluación de la prueba: Satisfactoria.

*Tabla 77. Prueba de aceptación 024
Elaborado por: Karen Aguaysa*

PRUEBA DE ACEPTACIÓN	
Número:	024
Historia de usuario N°:	027
Nombre: Visualización de ventas.	
Descripción: Se mostrará en la pantalla, con una tabla cargada de todas las ventas realizadas.	
Condiciones de ejecución: Ninguna.	
Interfaz: La tabla contiene los siguientes campos: <ul style="list-style-type: none"> • Cédula del cliente • Nombre • Dirección Y el detalla, con los siguientes datos. <ul style="list-style-type: none"> • Producto • Precio • Cantidad • Total 	
Resultado esperado: En caso de que el administrador haya ingresado a dicha pantalla, tendrá la posibilidad de consultar las ventas.	
Evaluación de la prueba: Satisfactoria.	

Tabla 78. Prueba de aceptación 025
Elaborado por: Karen Aguaysa

PRUEBA DE ACEPTACIÓN	
Número:	025
Historia de usuario N°:	028
Nombre: Recepción del listado de compras.	
Descripción: Se mostrará una pantalla donde lleguen todos los mensajes enviados a través de la aplicación móvil.	
Condiciones de ejecución: Ninguna.	
Interfaz: La pantalla tiene un cuadro de texto, en donde se cargan todos los mensajes recibidos.	
Resultado esperado: Cuando el usuario ingrese a esta pantalla, se puede consultar todos los mensajes recibidos por los clientes.	
Evaluación de la prueba: Satisfactoria.	

Tabla 79. Prueba de aceptación 026
Elaborado por: Karen Aguaysa

PRUEBA DE ACEPTACIÓN	
Número:	026
Historia de usuario N°:	029
Nombre: Ingreso a la aplicación móvil.	
Descripción: Se mostrará una pantalla donde el cliente debe ingresar todas las credenciales necesarias, en este caso cédula y contraseña.	
Condiciones de ejecución: Ninguna.	
Interfaz: La pantalla tiene dos cuadros de texto, en el primero se ingresa la cédula del cliente y en el segundo la contraseña del mismo, así también cuenta con un botón de ingreso y otro de registro.	
Resultado esperado: Cuando el cliente ingrese las credenciales correctamente, tendrá acceso a las funcionalidades de la aplicación móvil, caso contrario puede dirigirse a crear una cuenta.	
Evaluación de la prueba: Satisfactoria.	

Tabla 80. Prueba de aceptación 027
Elaborado por: Karen Aguaysa

PRUEBA DE ACEPTACIÓN	
Número:	027
Historia de usuario N°:	030
Nombre: Registro de cliente.	
Descripción: Se mostrará una pantalla donde con varios campos donde el cliente debe ingresar todos los datos necesarios para la creación de la nueva cuenta.	
Condiciones de ejecución: Ninguna.	
Interfaz: La pantalla cuenta con un botón de registro y con los siguientes campos: <ul style="list-style-type: none"> • Nombre • Apellido • Cédula • Dirección • Teléfono • Contraseña • Confirmar contraseña 	
Resultado esperado: Cuando el cliente ingrese todos los datos correctamente y presione sobre el botón de registro, se creará su nueva cuenta.	
Evaluación de la prueba: Satisfactoria.	

Tabla 81. Prueba de aceptación 028
Elaborado por: Karen Aguaysa

PRUEBA DE ACEPTACIÓN	
Número:	028
Historia de usuario N°:	031
Nombre: Catálogo de productos.	
Descripción: Se mostrará una pantalla donde se encuentren todos los productos que están activos en la base de datos, con sus respectivas características.	
Condiciones de ejecución: Ninguna.	
Interfaz: La pantalla cuenta con un catálogo de los productos existentes y un botón en la parte inferior de cada uno de éstos, para realizar la compra del mismo.	
Resultado esperado: Cuando el cliente presiona el botón de comprar, el producto es agregado al carrito.	
Evaluación de la prueba: Satisfactoria.	

Tabla 82. Prueba de aceptación 029
Elaborado por: Karen Aguaysa

PRUEBA DE ACEPTACIÓN	
Número:	029
Historia de usuario N°:	032
Nombre: Carrito con productos elegidos.	
Descripción: Se mostrará una pantalla donde se encuentren todos los productos que han sido comprados por el cliente.	
Condiciones de ejecución: Ninguna.	
Interfaz: La pantalla cuenta con una lista con los productos adquiridos, con un botón para confirmar la compra y otro para cancelar.	
Resultado esperado: Cuando el cliente presiona el botón de comprar, se ejecuta la compra, caso contrario se regresa a la pantalla de catálogo.	
Evaluación de la prueba: Satisfactoria.	

Tabla 83. Prueba de aceptación 030
Elaborado por: Karen Aguaysa

PRUEBA DE ACEPTACIÓN	
Número:	030
Historia de usuario N°:	033
Nombre: Confirmación de datos.	
Descripción: Se mostrará una pantalla donde se encuentra el producto seleccionado y se puede agregar más cantidad del mismo o a su vez eliminar del carrito.	
Condiciones de ejecución: Ninguna.	
Interfaz: La pantalla cuenta con una imagen del producto, y un botón de agregar, y otro de disminuir, también con un botón de comprar y otro de cancelar.	
Resultado esperado: Cuando el cliente presiona el botón de agregar, se suma la cantidad de stock o viceversa con el botón de disminuir y cuando se presiona el botón de comprar, se ejecuta la compra, caso contrario se regresa a la pantalla del carrito con los productos.	
Evaluación de la prueba: Satisfactoria.	

Tabla 84. Prueba de aceptación 031
Elaborado por: Karen Aguaysa

PRUEBA DE ACEPTACIÓN	
Número:	031
Historia de usuario N°:	034
Nombre: Confirmación de la compra.	
Descripción: Se mostrará una pantalla donde se envía un mensaje al cliente de que su compra ha sido realizada con éxito.	
Condiciones de ejecución: Ejecutar la compra en el carrito de productos.	
Interfaz: La pantalla cuenta con el logo del minimarket y un mensaje de confirmación de la compra.	
Resultado esperado: Cuando el cliente haya realizado la compra, se emitirá el mensaje que la confirme.	
Evaluación de la prueba: Satisfactoria.	

Tabla 85. Prueba de aceptación 032
Elaborado por: Karen Aguaysa

PRUEBA DE ACEPTACIÓN	
Número:	032
Historia de usuario N°:	035
Nombre: Envío del listado.	
Descripción: Se mostrará una pantalla donde se encuentre el mensaje enviado con los datos del cliente y los productos que se han comprado.	
Condiciones de ejecución: Ninguna.	
Interfaz: La pantalla cuenta con un cuadro de texto con el pedido enviado al dueño del minimarket.	
Resultado esperado: Cuando el cliente haya realizado la compra, se emitirá el chat y será enviado al dueño del minimarket.	
Evaluación de la prueba: Satisfactoria.	

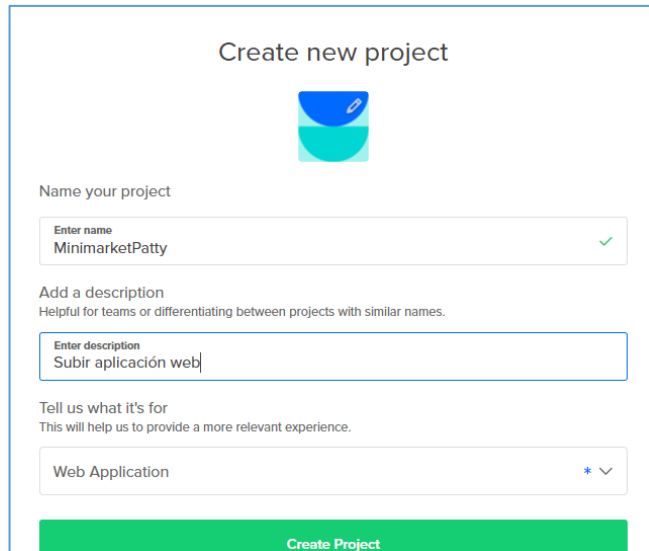
Tabla 86. Prueba de aceptación 033
Elaborado por: Karen Aguaysa

PRUEBA DE ACEPTACIÓN	
Número:	033
Historia de usuario N°:	036
Nombre: Actualización de datos del cliente.	
Descripción: Se mostrará una pantalla donde se puedan cargar los datos existentes del cliente y que puedan ser modificados.	
Condiciones de ejecución: Ninguna.	
Interfaz: La pantalla cuenta un botón de actualizar datos y con los siguientes campos: <ul style="list-style-type: none"> • Nombre • Apellido • Cédula • Dirección • Teléfono 	
Resultado esperado: Cuando el cliente haya presionado sobre el botón de actualizar datos, lo que haya sido modificado se guardará en la base de datos.	
Evaluación de la prueba: Satisfactoria.	

3.2.6.2. Pruebas de rendimiento

Para realizar las pruebas de rendimiento se ha subido los servicios a un alojamiento de llamado DigitalOcean que se caracteriza por el nivel de escalabilidad de los proyectos que se gestionan en el mismo, contando con una variedad de funciones que permite un mejor rendimiento del api.

Para alojar el api, es necesario crear un nuevo proyecto, con una descripción importante.



Create new project

Name your project

Enter name
MinimarketPatty ✓

Add a description
Helpful for teams or differentiating between projects with similar names.

Enter description
Subir aplicación web

Tell us what it's for
This will help us to provide a more relevant experience.

Web Application * v

Create Project

Figura 68. *Interfaz de DigitalOcean*
Elaborado por: Karen Aguaysa

Una vez creado el nuevo proyecto, se procede a iniciar un droplet para alojar el api desarrollado en nodeJs.

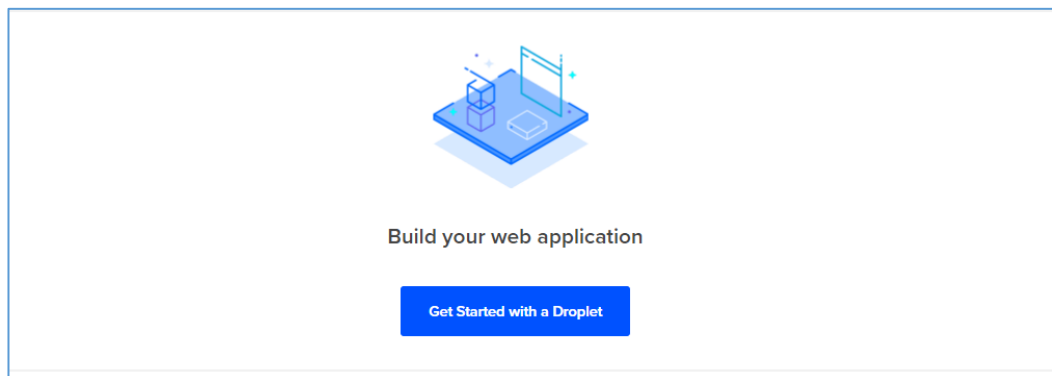


Figura 69. *Creación del droplet*
Elaborado por: Karen Aguaysa

De esta manera ya que ha sido creado el droplet, se puede alojar el api a través de algunos comandos, utilizando las credenciales que ha brindado el alojamiento, la ip del mismo es la siguiente: 143.198.156.98 y se puede conectar al mismo a través del comando:

ssh root@143.198.156.98

```
C:\Users\Karen>ssh root@143.198.156.98
Welcome to Ubuntu 20.04.2 LTS (GNU/Linux 5.4.0-73-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

System information as of Mon Jul  5 19:51:10 UTC 2021

System load:  0.0           Users logged in:      1
Usage of /:   10.0% of 24.06GB IPv4 address for eth0: 143.198.156.98
Memory usage: 61%         IPv4 address for eth0: 10.48.0.5
Swap usage:   0%          IPv4 address for eth1: 10.124.0.2
Processes:   106

0 updates can be applied immediately.

*** System restart required ***
Last login: Mon Jul  5 17:29:22 2021 from 186.47.139.84
root@ubuntu-s-1vcpu-1gb-sfo3-01:~#
```

*Figura 70. Comando ssh en el servidor
Elaborado por: Karen Aguaysa*

Al ejecutar el api, se podrá observar un mensaje de conexión satisfactoria en la consola, a través del siguiente comando: **npm run start:dev**

```
root@productionserver: ~/projects/node-api-minimarket

> api-rest-minimarket@1.0.0 start:dev
> nodemon app.js

sh: 1: nodemon: not found
root@productionserver:~/projects/node-api-minimarket# npm install nodemon
added 116 packages, and audited 202 packages in 10s

11 packages are looking for funding
  run `npm fund` for details

1 high severity vulnerability

To address all issues, run:
  npm audit fix

Run `npm audit` for details.
root@productionserver:~/projects/node-api-minimarket# npm run start:dev

> api-rest-minimarket@1.0.0 start:dev
> nodemon app.js

[nodemon] 2.0.9
[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): *.*
[nodemon] watching extensions: js,mjs,json
[nodemon] starting `node app.js`
Servidor corriendo en el puerto 3000

Branch 'master' set up to track remote branch 'master' from 'origin'.
PS C:\Users\Karen\Desktop\ProyectoSimple\minimarket-rest-api-master> npm run start:dev
```

*Figura 71. Servidor encendido
Elaborado por: Karen Aguaysa*

Una vez que se ha levantado el api en el droplet creado, se obtendrá la siguiente pantalla, en la que se puede observar la activación de la misma.

```

root@productionserver: ~/projects/node-api-minimarket
Load Balance 4 instances of api.js:
$ pm2 start api.js -i 4

Monitor in production:
$ pm2 monitor

Make pm2 auto-boot at server restart:
$ pm2 startup

To go further checkout:
http://pm2.io/

-----
PM2] Spawning PM2 daemon with pm2_home=/root/.pm2
PM2] PM2 Successfully daemonized
PM2] Starting /root/projects/node-api-minimarket/app.js in fork_mode (1 instance)
PM2] Done.

```

id	name	namespace	version	mode	pid	uptime	Ⓜ	status	cpu	mem	user	watching
0	app	default	1.0.0	Fork	16928	0s	0	online	0%	31.1mb	root	disabled

```

root@productionserver:~/projects/node-api-minimarket#

```

Figura 72. El servicio corriendo online
Elaborado por: Karen Aguaysa

Para realizar las pruebas de carga se instala ApacheBench a través del comando: **sudo apt install apache2-utils**

Posteriormente, se ejecuta la línea de comando: `b -n 5000 -c 100 http://localhost:80/` en donde 5000 son el número de solicitudes enviadas y 100 la concurrencia con que han sido enviadas.

```

Concurrency Level:      100
Time taken for tests:   4.020 seconds
Complete requests:     5000
Failed requests:       0
Non-2xx responses:     5000
Total transferred:     2230000 bytes
HTML transferred:     695000 bytes
Requests per second:   1243.77 [#]/sec (mean)
Time per request:      80.401 [ms] (mean)
Time per request:      0.804 [ms] (mean, across all concurrent requests)
Transfer rate:         541.72 [Kbytes/sec] received

Connection Times (ms)
  min   mean[+/-sd] median   max
Connect:    0    0  0.8    0     7
Processing:  3    80 15.9   78   129
Waiting:    2    80 15.9   78   129
Total:      9    80 15.6   78   129

Percentage of the requests served within a certain time (ms)
 50%    78
 66%    84
 75%    89
 80%    92
 90%   102
 95%   109
 98%   118
 99%   120
100%   129 (longest request)
root@productionserver:~#

```

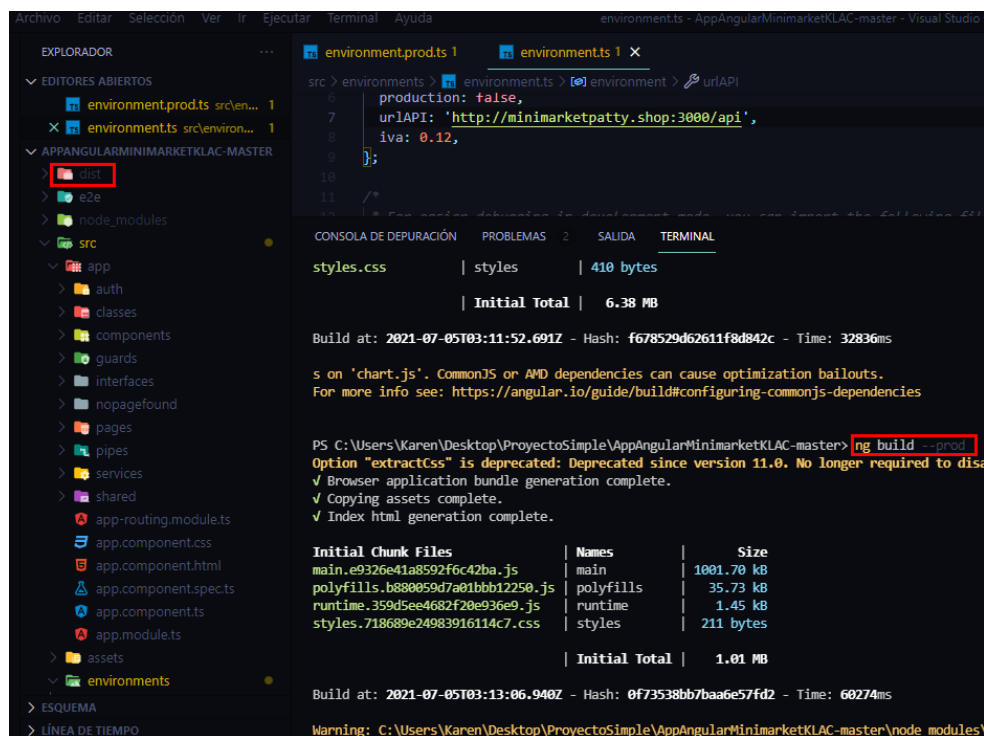
Figura 73. Prueba de rendimiento
Elaborado por: Karen Aguaysa

Se ha obtenido como resultado que todas las solicitudes con la concurrencia anteriormente especificadas, fueron completadas en un tiempo de 4.02 segundos, obteniendo con rapidez un porcentaje del 100% en un corto tiempo específico y todas éstas han sido procesadas y escritas.

3.2.7. Fase IV: Implantación

3.2.7.1. Instalación de la app en el servidor

Se ejecuta el comando **ng build --serve** para desplegar la aplicación y poderla subirla al servidor. Al completarse la tarea se creará una carpeta llamada **dist** en la cual se encuentran los archivos para enviar al servidor.



```
src > environments > environment.ts > environment > urlAPI
6   production: false,
7   urlAPI: 'http://minimarketpatty.shop:3000/api',
8   iva: 0.12,
9 };
10
11 /*
12 * For better debugging in development mode you can insert the following file
13 * into the 'src/environments/environment.ts' file.
14 */

CONSOLE DE DEPURACIÓN  PROBLEMAS  SALIDA  TERMINAL

styles.css | styles | 410 bytes

| Initial Total | 6.38 MB

Build at: 2021-07-05T03:11:52.691Z - Hash: f678529d62611f8d842c - Time: 32836ms

s on 'chart.js'. CommonJS or AMD dependencies can cause optimization bailouts.
For more info see: https://angular.io/guide/build#configuring-commonjs-dependencies

PS C:\Users\Karen\Desktop\ProyectoSimple\AppAngularMinimarketKLAC-master> ng build --prod
Option "extractCss" is deprecated: Deprecated since version 11.0. No longer required to disa
√ Browser application bundle generation complete.
√ Copying assets complete.
√ Index html generation complete.

Initial Chunk Files | Names | Size
main.e9326e41a8592f6c42ba.js | main | 1001.70 kB
polyfills.b880059d7a01bbb12250.js | polyfills | 35.73 kB
runtime.359d5ee4682f20e936e9.js | runtime | 1.45 kB
styles.718689e24983916114c7.css | styles | 211 bytes

| Initial Total | 1.01 MB

Build at: 2021-07-05T03:13:06.940Z - Hash: 0f73538bb7baa6e57fd2 - Time: 60274ms

Warning: C:\Users\Karen\Desktop\ProyectoSimple\AppAngularMinimarketKLAC-master\node_modules\
```

*Figura 74. Ejecución del comando build
Elaborado por: Karen Aguaysa*

El contenido de la carpeta dist es el siguiente:

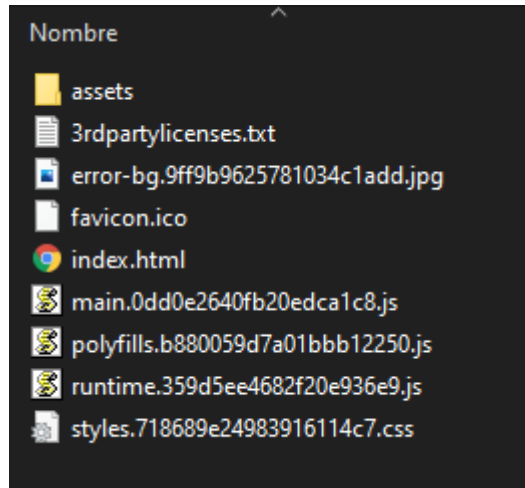


Figura 75. Carpeta dist
Elaborado por: Karen Aguaysa

Luego se instala FileZilla para subir los archivos al servidor de la siguiente manera; Se selecciona la pestaña archivo y Gestor de Sitios

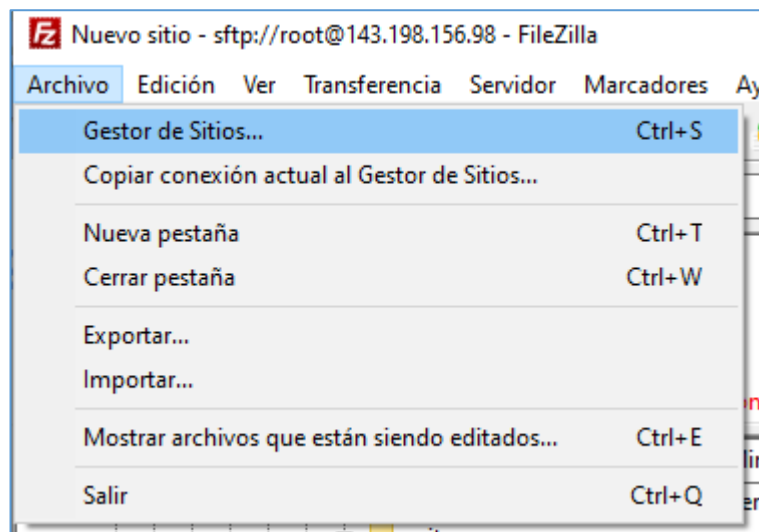


Figura 76. Instalación de FileZilla
Elaborado por: Karen Aguaysa

Se selecciona la opción nuevo sitio, seguido del protocolo SFTP - SSH File Transfer Protocol y luego se escribe la dirección IP del servidor.

El modo de acceso será por medio de archivo de claves, el usuario será root y finalmente se selecciona la opción de conectar y se obtendrá el acceso al servidor desde FileZilla.

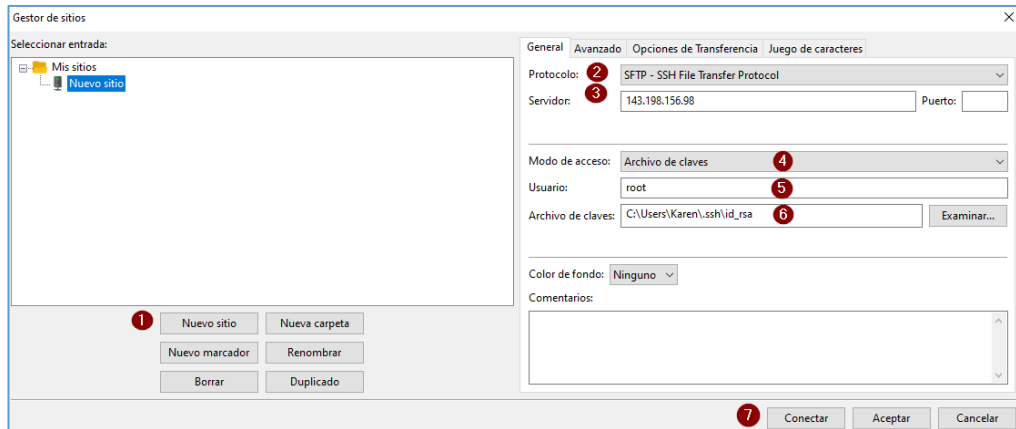


Figura 77. Configuración de FileZilla
Elaborado por: Karen Aguaysa

Posteriormente se ubica, en el directorio de la carpeta **dist**, para luego arrastrar dicha carpeta a el servidor, el cual se encuentra a la derecha.

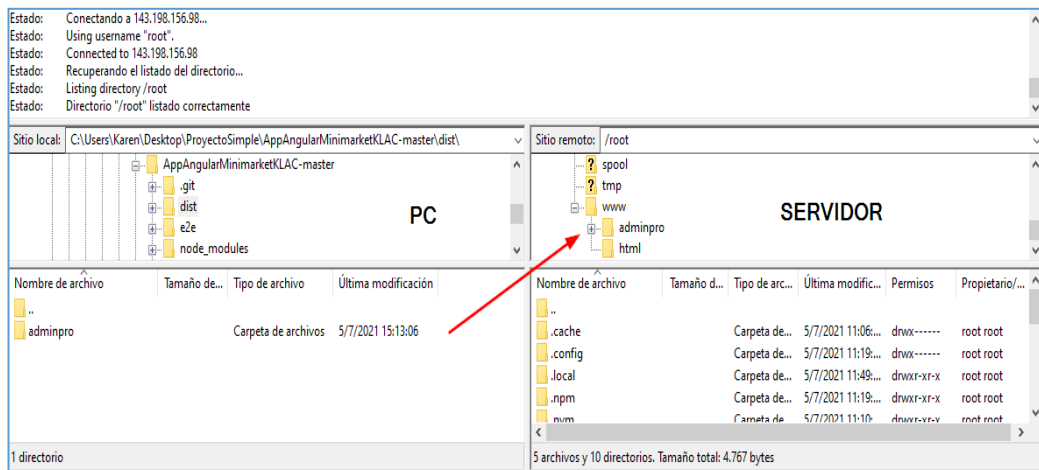


Figura 78. Directorio de la carpeta dist
Elaborado por: Karen Aguaysa

Se procede a la instalación de nginx en el servidor con el comando **sudo apt install nginx**.


```

root@ubuntu-s-1vcpu-1gb-sfo3-01:~/projects/node-api-minimarket# sudo apt install nginx
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  fontconfig-config fonts-dejavu-core libfontconfig1 libgd3 libjbig0 libjpeg-turbo8 libjp
  libnginx-mod-http-xslt-filter libnginx-mod-mail libnginx-mod-stream libtiff5 libwebp6 l
Suggested packages:
  libgd-tools fcgiwrap nginx-doc ssl-cert
The following NEW packages will be installed:
  fontconfig-config fonts-dejavu-core libfontconfig1 libgd3 libjbig0 libjpeg-turbo8 libjp
  libnginx-mod-http-xslt-filter libnginx-mod-mail libnginx-mod-stream libtiff5 libwebp6 l
0 upgraded, 17 newly installed, 0 to remove and 0 not upgraded.
Need to get 2431 kB of archives.
After this operation, 7891 kB of additional disk space will be used.
Do you want to continue? [Y/n] y

```

Figura 79. Instalación de nginx
Elaborado por: Karen Aguaysa

Se ubica en la ruta `/etc/nginx/sites-available` con el comando `cd /etc/nginx/sites-available` y se crea el archivo de configuración para redirigir el sitio web.

```

root@ubuntu-s-1vcpu-1gb-sfo3-01:~# cd /etc/nginx/sites-available/
root@ubuntu-s-1vcpu-1gb-sfo3-01:/etc/nginx/sites-available# nano minimarketpatty.shop
root@ubuntu-s-1vcpu-1gb-sfo3-01:/etc/nginx/sites-available# █

```

Figura 80. Ingreso a la carpeta etc
Elaborado por: Karen Aguaysa

El archivo `minimarket.shop` debe quedar de la siguiente manera en la cual se especifica el puerto que va a escuchar, la dirección de los archivos para el sitio web y la página principal que se abrirá al ingresar al sitio web.

```

server{
    listen 80;

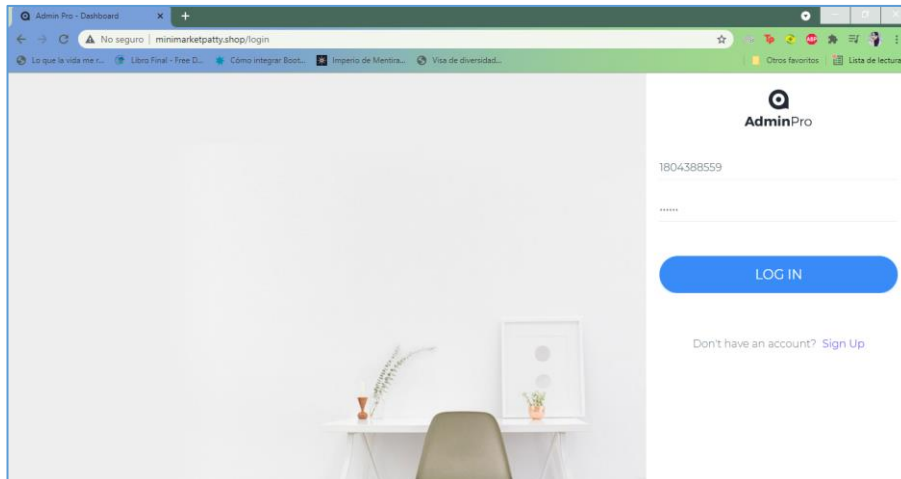
    server_name minimarketpatty.shop www.minimarketpatty.shop;
    root /var/www/adminpro;

    location / {
        index index.html;
        try_files $uri $uri/ /index.html;
    }
}

```

Figura 81. Configuración del server
Elaborado por: Karen Aguaysa

Se reinicia el servicio de nginx y se comprueba que la página web esté funcionando correctamente, ingresando en un navegador.



*Figura 82. Aplicación web en el dominio
Elaborado por: Karen Aguaysa*

3.2.7.2. Creación del .apk de la aplicación móvil

Desarrollar una apk para instalar la aplicación en un dispositivo Android, requiere de los siguientes pasos ya que es un paquete que requiere de algunas instalaciones.

Primeramente, se ejecuta el comando **ionic build**, para que se copien los archivos en la plataforma nativa y para que se abra el IDE nativo del mismo.

```
PROBLEMS  OUTPUT  TERMINAL  DEBUG CONSOLE
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Projects\AppMobileMinimarketKAC> ionic build
> ng.cmd run app:build
Compiling @angular/core : es2015 as esm2015
Compiling @angular/common : es2015 as esm2015
Compiling @angular/platform-browser : es2015 as esm2015
Compiling @angular/forms : es2015 as esm2015
Compiling @angular/platform-browser-dynamic : es2015 as esm2015
Compiling @ionic/storage-angular : es2015 as esm2015
Compiling @angular/router : es2015 as esm2015
Compiling @angular/common/http : es2015 as esm2015
Compiling @ionic/angular : es2015 as esm2015
```

*Figura 83. Ejecución del comando build
Elaborado por: Karen Aguaysa*

```

PS C:\Projects\AppMobileMinimarketKLAC> ionic build
> ng.cmd run app:build
Compiling @angular/core : es2015 as esm2015
Compiling @angular/common : es2015 as esm2015
Compiling @angular/platform-browser : es2015 as esm2015
Compiling @angular/forms : es2015 as esm2015
Compiling @angular/platform-browser-dynamic : es2015 as esm2015
Compiling @ionic/storage-angular : es2015 as esm2015
Compiling @angular/router : es2015 as esm2015
Compiling @angular/common/http : es2015 as esm2015
Compiling @ionic/angular : es2015 as esm2015
✓ Browser application bundle generation complete.
✓ Copying assets complete.
✓ Index html generation complete.

Initial Chunk Files | Names | Size
vendor.js | vendor | 4.72 MB
polyfills.js | polyfills | 252.43 kB
styles.css | styles | 28.55 kB
main.js | main | 71.02 kB
runtime.js | runtime | 9.80 kB

| Initial Total | 5.02 MB

Lazy Chunk Files | Names | Size
swiper-bundle-44a9b1f9-js.js | swiper-bundle-44a9b1f9-js | 200.02 kB
polyfills-core-js.js | polyfills-core-js | 92.39 kB
11.js | - | 72.60 kB
17.js | - | 72.29 kB
2.js | - | 63.70 kB
33.js | - | 56.49 kB
1.js | - | 47.59 kB
26.js | - | 47.15 kB
31.js | - | 37.46 kB
common.js | common | 37.29 kB
30.js | - | 32.95 kB
tab1-tab1-module.js | tab1-tab1-module | 32.90 kB
21.js | - | 32.50 kB
6.js | - | 32.27 kB
19.js | - | 32.15 kB
pages-auth-auth-module.js | pages-auth-auth-module | 31.78 kB
16.js | - | 31.12 kB
0.js | - | 29.80 kB
20.js | - | 26.68 kB

```

*Figura 84. Resultado de la ejecución del comando build
Elaborado por: Karen Aguaysa*

Una vez que se han realizado las configuraciones anteriores, se procede a instalar la plataforma Android, a través del siguiente comando: **npx cap add Android**

```

PS C:\Projects\AppMobileMinimarketKLAC> npx cap add android
✓ Installing android dependencies in 16.76s
✓ Adding native android project in: C:\Projects\AppMobileMinimarketKLAC\android in 74.49ms
✓ Syncing Gradle in 604.00µs
✓ add in 16.84s
✓ Copying web assets from www to android\app\src\main\assets\public in 876.40ms
✓ Copying native bridge in 2.32ms
✓ Copying capacitor.config.json in 1.87ms
✓ copy in 910.38ms
✓ Updating Android plugins in 10.01ms
  Found 0 Capacitor plugins for android:
✓ update android in 34.60ms

Now you can run npx cap open android to launch Android Studio
PS C:\Projects\AppMobileMinimarketKLAC>

```

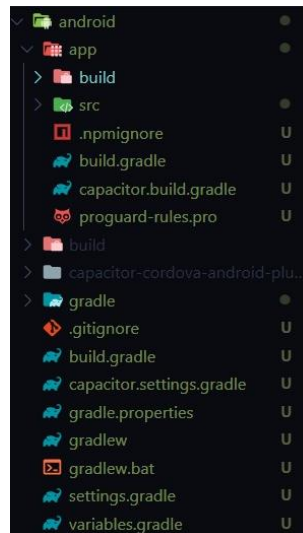
*Figura 85. Ejecución del comando capacitor
Elaborado por: Karen Aguaysa*

Una vez realizado el paso anterior, se procede a ejecutar el comando **npx cap sync**, que permite realizar una compilación de ionic, y se copia los assets y los plugins

```
PS C:\Projects\AppMobileMinimarketKLAC> npx cap sync
√ Copying web assets from www to android\app\src\main\assets\public in 1.01s
√ Copying native bridge in 1.71ms
√ Copying capacitor.config.json in 1.80ms
√ copy in 1.05s
√ Updating Android plugins in 12.90ms
  Found 0 Capacitor plugins for android:
√ update android in 39.49ms
√ copy in 308.00µp
√ update web in 5.00µp
Sync finished in 1.108s
PS C:\Projects\AppMobileMinimarketKLAC>
```

*Figura 86. Compilación de Ionic
Elaborado por: Karen Aguaysa*

Ya que se han realizado los pasos anteriores, en el siguiente repositorio se encuentra generado nuestro archivo .apk que permitirá la ejecución de la aplicación móvil.



*Figura 87. Repositorio del apk
Elaborado por: Karen Aguaysa*

CAPÍTULO IV

CONCLUSIONES Y RECOMENDACIONES

4.1. Conclusiones

- El reconocimiento de los procesos que realizaba de manera manual, fueron importantes para el desarrollo de la aplicación web/móvil ya que permitió identificar los puntos clave que deseaba automatizar el administrador para automatizar dichos procesos.
- El framework de Angular con Ionic, basado en TypeScript facilitó el desarrollo de las aplicaciones web/móvil ya que, al manejar el mismo lenguaje de programación, permitió reutilizar varios componentes y funcionalidades, además permitió la compatibilidad con el servidor, accediendo a conectarse hacia una base de datos centralizada.
- La tecnología de websocket a través de la librería socket.io que se encuentra basada en el protocolo ws, lo que facilitó la conexión bidireccional entre el cliente y el servidor, con su característica principal y la más importante, que es en tiempo real, permitiendo el desarrollo de un chat uno a uno que mantiene la comunicación inmediata entre el emisor y el receptor.
- El desarrollo del presente proyecto, utilizó una metodología ágil, la cual fue de gran valor e importancia, puesto que permitió organizar el trabajo en partes pequeñas y estructuradas, haciendo mucho más fácil documentación del desarrollo y brindó una experiencia agradable al desarrollador ya que mantuvo el orden de las cosas y un panorama claro acerca de los requerimientos y las necesidades que se debían cumplir.
- Para el correcto funcionamiento del sistema se recomienda el uso de un servidor web que podría ser compartido o dedicado, pero que debe tener como característica por lo menos 2GB de RAM, de 1 a 2 CPU virtuales y tener una versión instalada de Node.js 14.0 o superiores.
- La importancia del desarrollo de proyectos basados en nuevos lenguajes de programación como TypeScript, permite mejorar la experiencia del

desarrollador, puesto que son herramientas que ayudan a la verificación de errores en el código, así como también da paso a la escalabilidad del mismo y la documentación que resulta ser muy importante en el desarrollo del proyecto.

- Al mantener una comunicación directa, entre el cliente y el administrador del minimarket, a través de un canal de comunicación como es el chat, mejora las relaciones interpersonales que se establecen entre éstas y permite conocer más de cerca a cada uno de éstos, intentando satisfacer las necesidades del cliente y brindar un servicio de calidad para que éste se sienta cómodo y elija realizar sus compras en el mismo.
- La aplicación desarrollada, generó un gran impacto tanto en el personal administrativo del minimarket, como en los clientes, puesto que los procesos mejoraron para las dos partes y sobre todo tienen la oportunidad de innovar en nuevas tecnologías y los clientes sienten que sus necesidades son tomadas en cuenta, lo que fortalece los lazos entre comprador y vendedor.

4.2. Recomendaciones

- Se recomienda utilizar servidores que cuenten con las características necesarias de acuerdo al negocio, para que las aplicaciones sean eficientes y cumplan con los requerimientos especificados por los usuarios, de manera que no exista inconvenientes al usar servidores básicos que no generan o procesan los datos que se requieren.
- Para el correcto funcionamiento del sistema se recomienda el uso de un servidor que podría ser compartido o dedicado, con un hardware de por lo menos 2GB de RAM, de 1 a 2 CPU virtuales y tener una versión instalada de Node.js 14.0 o superiores.
- Para el desarrollo de futuros proyectos en Ionic, conviene seguir innovando en la funcionalidad de Capacitor, puesto que a pesar ser reciente su lanzamiento ya contiene funcionalidades que permiten mejorar el desarrollo de proyectos, brindando acceso al sdk de cada una de las plataformas, lo que es de gran ayuda ya que genera un alto rendimiento la aplicación.

- Se debe considerar la ampliación del sistema, conjuntamente con otros módulos contables, que ayudarán a la mejor administración del minimarket “Patty” puesto que éste va creciendo a diario y es importante mantener un registro contable de todos los movimientos que se realizan dentro del mismo.
- Se recomienda fomentar a los clientes del minimarket “Patty” el uso de nuevas tecnologías que les permita seguir innovando en su conocimiento de nuevas herramientas y de ésta manera preservar ciertos protocolos de bioseguridad, que actualmente se manejan en la sociedad para preservar la salud y bienestar de las familias.

BIBLIOGRAFÍA

- [1] D. Chicaiza, “Facultad De Ingeniería En Sistemas Electrónica E Industrial Carrera De Ingeniería Industrial En Procesos De,” p. 250, 2020.
- [2] A. Gómez, “Aplicación Android para la empresa Travelling-Service,” p. 89, 2014, [Online]. Available: https://repositorio.uam.es/bitstream/handle/10486/662281/gomez_matesanz_al_fonso_tfg.pdf?sequence=1.
- [3] C. D. E. I. D. E. Sistemas, “Diseño y desarrollo de un sistema basado en entornos web y móvil para brindar soporte en la evaluación de la función auditiva en niños.,” 2018.
- [4] A. V. Gavilima, “Diseño e implementación de un sistema Web de inventarios y facturación para el control de productos terminados utilizando software libre para los almacenes ‘Pichaví’ perteneciente a la unidad educativa Cotacachi,” 2015, [Online]. Available: <http://repositorio.utn.edu.ec/handle/123456789/7700>.
- [5] J. A. Arévalo, “Gestión de la Información, gestión de contenidos y conocimiento,” Salamanca, 2007. http://eprints.rclis.org/11273/1/Jornadas_GRUPO_SIOU.pdf.
- [6] A. G. M. DE TORRE, “LA WEB MÓVIL,” 2019. <https://www.ceupe.com/blog/la-web-movil.html>.
- [7] T. Naeem, “Integración de aplicaciones,” XNUM, 2009. <https://www.astera.com/es/type/blog/application-integration/>.
- [8] F. Culebras, “Servicios:Aplicaciones web móvil,” 2020. <https://www.solucionesip.com/servicios/aplicaciones-web-movil>.
- [9] Anónimo, “GESTIÓN DE CALIDAD,” 2017. <https://concepto.de/gestion-de-calidad/>.
- [10] S. Á. Santás, “Logística para la Ventas,” *udima*, 2018. udima.es/es/logistica-venta-124.html.
- [11] E. T. Canelo, “Control Interno de los Inventarios,” 2014. <https://www.auditool.org/blog/control-interno/939-control-interno-de-los-inventarios>.
- [12] B. B. EDWIN, “GESTIÓN DE INVENTARIOS,” JIMBO, 2010. <https://logisticayabastecimiento.jimdofree.com/gestión-de-inventarios/>.
- [13] Centria, “Desarrollo de aplicaciones web y móviles,” 2020. <https://centria.eu/desarrollo-de-aplicaciones-web-y-moviles/>.
- [14] P. Arizmendi, *AngularJS*. Paiminix, 2018.
- [15] R. Varela, “Qué es Ionic y todas sus ventajas,” 2018. <https://www.randyvarela.es/ionic-definicion-ventajas/>.
- [16] Microsoft, “Bases de datos,” 2017. <https://docs.microsoft.com/es->

es/sql/relational-databases/databases/databases?view=sql-server-ver15.

- [17] *M. Rouse*, “MySQL,” 2020. <https://searchdatacenter.techtarget.com/es/definicion/MySQL>.
- [18] OpenJS, “Acerca de Node.js,” 2019. <https://nodejs.org/es/about/>.
- [19] Anónimo, “Introducción a Express/Node,” 2020. https://developer.mozilla.org/es/docs/Learn/Server-side/Express_Nodejs/Introduction.
- [20] *E. G. Maida and J. Pacienza*, “METODOLOGIAS DE DESARROLLO DE SOFTWARE,” Universidad Católica Argentina, 2015.