

UNIVERSIDAD NACIONAL DE CAJAMARCA

FACULTAD DE INGENIERÍA

ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS



“Influencia de la implementación de una aplicación web y móvil en el proceso de ventas de la empresa Nikos Pizza - Cajamarca”

TESIS PARA OPTAR EL TÍTULO PROFESIONAL DE INGENIERO DE SISTEMAS

AUTOR

Bach. Roberto Jhuliño Tafur Valdivia

ASESOR

Ing. Manuel Enrique Malpica Rodríguez
Ingeniero de Sistemas

CAJAMARCA – PERÚ

2025

CONSTANCIA DE INFORME DE ORIGINALIDAD

- FACULTAD DE INGENIERÍA -

1. **Investigador:** Roberto Jhuliño Tafur Valdivia

DNI: 74231117

Escuela Profesional: Ingeniería de Sistemas

2. **Asesor:** Ing. Manuel Enrique Malpica Rodríguez

Facultad: Ingeniería

3. **Grado académico o título profesional**

Bachiller Título profesional Segunda especialidad
 Maestro Doctor

4. **Tipo de Investigación:**

Tesis Trabajo de Investigación Trabajo de suficiencia profesional
 Trabajo académico

5. **Título de Trabajo de Investigación:**

Influencia de la implementación de una aplicación web y móvil en el proceso de ventas de la empresa Nikos Pizza - Cajamarca

6. **Fecha de evaluación:** 06 de noviembre de 2025

7. **Software antiplagio:** TURNITIN URKUND (OURIGINAL) (*)

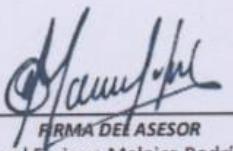
8. **Porcentaje de Informe de Similitud:** 18%

9. **Código Documento:** 524288197

10. **Resultado de la Evaluación de Similitud:**

APROBADO PARA LEVANTAMIENTO DE OBSERVACIONES O DESAPROBADO

Fecha Emisión: 08 de noviembre de 2025



Ing. Manuel Enrique Malpica Rodríguez
DNI: 26707158



Firmado digitalmente por:
BAZAN DIAZ Laura Sofía
FAU 20148258601 soft.
Motivo: En señal de
conformidad
Fecha: 08/11/2025 12:37:22-0500

UNIDAD DE INVESTIGACIÓN FI



'Universidad Nacional de Cajamarca'

"Norte de la Universidad Peruana"

Fundada por Ley 14015 del 13 de Febrero de 1962

FACULTAD DE INGENIERÍA

Teléf. N° 365976 Anexo N° 1129-1130



ACTA DE SUSTENTACIÓN PÚBLICA DE TESIS.

TITULO : *INFLUENCIA DE LA IMPLEMENTACIÓN DE UNA APLICACIÓN WEB Y MÓVIL EN EL PROCESO DE VENTAS DE LA EMPRESA NIKOS PIZZA - CAJAMARCA.*

ASESOR : *Dr. Ing. Manuel Malpica Rodríguez.*

En la ciudad de Cajamarca, dando cumplimiento a lo dispuesto por el Oficio Múltiple N° 0749-2025-PUB-SA-FI-UNC, de fecha 15 de diciembre de 2025, de la Secretaría Académica de la Facultad de Ingeniería, a los **diecisiete días del mes de diciembre de 2025**, siendo las doce horas (12:00 m.) en el Aula Taller 8 (Edificio 4K) de la Escuela Profesional de Ingeniería de Sistemas, de la Facultad de Ingeniería se reunieron los Señores Miembros del Jurado Evaluador:

Presidente : Dr. Ing. Edwin Alberto Valencia Castillo.
Vocal : Dr. Ing. Jaime Amador Meza Huamán.
Secretario : M.Cs. Ing. Roger Manuel Sánchez Chávez.

Para proceder a escuchar y evaluar la sustentación pública de la tesis titulada *INFLUENCIA DE LA IMPLEMENTACIÓN DE UNA APLICACIÓN WEB Y MÓVIL EN EL PROCESO DE VENTAS DE LA EMPRESA NIKOS PIZZA - CAJAMARCA*, presentado por el Bachiller en Ingeniería de Sistemas **ROBERTO JHULIÑO TAFUR VALDIVIA**, asesorado por el Dr. Ing. Manuel Enrique Malpica Rodríguez, para la obtención del Título Profesional

Los Señores Miembros del Jurado replicaron al sustentante, debatieron entre sí en forma libre y reservada y lo evaluaron de la siguiente manera:

EVALUACIÓN PRIVADA : **06** PTS.

EVALUACIÓN PÚBLICA : **10** PTS.

EVALUACIÓN FINAL : **16** PTS *Diecisiete* (En letras) *Dieciseis*

En consecuencia, se lo declara *APROBADO* con el calificativo de *Diecisiete* acto seguido, el presidente del jurado hizo saber el resultado de la sustentación, levantándose la presente a las *12:00* horas del mismo día, con lo cual se dio por terminado el acto, para constancia se firmó por quintuplicado.

Dr. Ing. Edwin Alberto Valencia Castillo.
Presidente

Dr. Ing. Jaime Amador Meza Huamán.
Vocal

Dr. Ing. Roger Manuel Sánchez Chávez.
Secretario

Dr. Ing. Manuel Enrique Malpica Rodríguez.
Asesor

**COPYRIGHT © 2025
2025
Todos los Derechos Reservados ®**

AGRADECIMIENTO

Quiero expresar mi más profundo agradecimiento a todas las personas que han hecho posible la realización de esta tesis. A mis profesores, especialmente a Ing. Manuel Malpica Rodríguez, quien con paciencia y dedicación me ha brindado una guía invaluable en cada etapa de esta investigación.

A mi familia, quienes han sido un pilar de apoyo incondicional, brindándome su amor y aliento a lo largo de este camino académico. Su respaldo emocional y motivación me han dado la fortaleza para perseverar y culminar esta etapa.

A mis amigos y compañeros de estudios, quienes han compartido conmigo largas horas de trabajo, reflexión y aprendizaje mutuo. Gracias por los momentos, el apoyo y colaboración, que hicieron que este proceso fuera más llevadero.

Finalmente, agradezco a los directivos de Nikos Pizza, quienes me brindaron su tiempo y disposición para acceder a la información necesaria para la realización de esta investigación. Su apertura y colaboración fueron esenciales para comprender a fondo el funcionamiento de su proceso de ventas y la influencia de la tecnología en su negocio.

DEDICATORIA

Dedico este trabajo a mi mamá, quien con su amor y esfuerzo ha sido la mayor inspiración en mi vida y quien, desde el primer día, creyó en mis capacidades. A ella, que me enseñó a luchar por mis sueños y a no rendirme ante las adversidades, dedico cada logro de esta etapa académica.

Extiendo esta dedicatoria a mi hermana, por su cariño, alegría y por acompañarme en cada paso de este camino. Gracias por ser un apoyo constante y una presencia que inspira.

CONTENIDO

RESUMEN	xvii
ABSTRACT	xviii
CAPÍTULO I. INTRODUCCIÓN	1
CAPÍTULO II. MARCO TEÓRICO	4
2.1 Antecedentes teóricos de la investigación.....	4
2.1.1 Antecedentes internacionales	4
2.1.2 Antecedentes nacionales	5
2.1.3 Antecedentes locales	6
2.2 Bases teóricas	7
2.2.1 Aplicación web y móvil	7
2.2.2 Proceso de ventas	10
2.2.3 Definición de términos básicos	13
CAPÍTULO III. MATERIALES Y MÉTODOS.....	16
3.1 Procedimiento.....	17
3.1.1 Requisitos	20
3.1.2 Análisis.....	23
3.1.3 Diseño.....	34
3.1.4 Implementación.....	53
3.1.5 Pruebas	84
3.1.6 Despliegue.....	86
3.1.7 Mantenimiento	90
3.2 Tratamiento, análisis de datos y presentación de resultados	91

3.2.1	Tratamiento	91
3.2.2	Análisis de datos.....	91
3.2.3	Presentación de resultados	108
CAPÍTULO IV. ANÁLISIS Y DISCUSIÓN DE RESULTADOS		113
CAPÍTULO V. CONCLUSIONES Y RECOMENDACIONES		116
5.1	Conclusiones	116
5.2	Recomendaciones.....	117
REFERENCIAS BIBLIOGRÁFICAS		118
ANEXOS.....		123
Anexo 1: Operacionalización de variables.....		123
Anexo 2: Matriz de consistencia		124
Anexo 3: Cuestionario para la variable independiente.....		125
Anexo 4: Ficha de observación para la variable dependiente, dimensión tiempo		127
Anexo 5: Cuestionario para la variable dependiente, dimensión satisfacción y calidad....		128
Anexo 6: Fichas de validación para el cuestionario de la variable independiente		131
Anexo 7: Fichas de validación para el cuestionario de la variable dependiente		133
Anexo 8: Alfa de cronbach para el cuestionario 1		135
Anexo 9: Alfa de cronbach para el cuestionario 2		136
Anexo 10: Fichas de validación de ficha de observación		137
Anexo 11: Número de lecturas según la duración del ciclo		139
Anexo 12: Pruebas de normalidad para la dimensión tiempo.....		140
Anexo 13: Pruebas de normalidad para la dimensión satisfacción		141
Anexo 14: Pruebas de normalidad para la dimensión calidad		142
Anexo 15: Pruebas de aceptación de usuario		143

Anexo 16: Capex y opex 145

Anexo 17: Casos de uso de la aplicación web y móvil 148

ÍNDICE DE TABLAS

Tabla I: Especificación de caso de uso registrar pedido	29
Tabla II: Especificación de caso de uso registrar pago	31
Tabla III: Especificación de caso de uso generar reporte de ventas.....	32
Tabla IV: Categorización de frameworks por criterios	36
Tabla V: Puntajes de cada framework por criterio.....	37
Tabla VI: Categorización de hosting por criterio.....	38
Tabla VII: Puntajes para cada hosting por criterio.....	39
Tabla VIII: Categorización del servidor (VPS) por criterio.....	40
Tabla IX: Pesos de criterio los servidores	40
Tabla X: Tiempo promedio y número de observaciones en la fase de pretest para la dimensión tiempo.....	93
Tabla XI: Tiempo promedio y número de observaciones en la fase de postest para la dimensión tiempo.....	95
Tabla XII: Detalle del tiempo que tarda registrar un pedido en cada ciclo de observación....	96
Tabla XIII: Detalle del tiempo que tarda para elaborar un reporte de ventas en cada ciclo. ...	97
Tabla XIV: Detalle del tiempo que tarda en registrar un pago de un pedido en cada ciclo.....	98
Tabla XV: Estadísticas de muestras emparejadas para la dimensión tiempo	100
Tabla XVI: Pruebas de medias emparejadas para la dimensión tiempo	101
Tabla XVII: Puntaje promedio de cada indicador de la dimensión satisfacción (pretest).	102
Tabla XVIII: Puntaje promedio de cada indicador de la dimensión satisfacción (postest). ..	103
Tabla XIX: Prueba no paramétrica para el indicador 1 de la dimensión satisfacción.....	104
Tabla XX: Prueba no paramétrica para el indicador 2 de la dimensión satisfacción.....	105
Tabla XXI: Prueba no paramétrica para el indicador 3 de la dimensión satisfacción.....	105

Tabla XXII: Prueba no paramétrica para el indicador 4 de la dimensión satisfacción.	105
Tabla XXIII: Puntaje promedio de cada indicador de la dimensión satisfacción (pretest)....	106
Tabla XXIV: Puntaje promedio de cada indicador de la dimensión satisfacción (postest)...	106
Tabla XXV: Prueba no paramétrica para el indicador 1 de la dimensión calidad.	108
Tabla XXVI: Prueba no paramétrica para el indicador 2 de la dimensión calidad.	108
Tabla XXVII: Porcentaje de reducción en la dimensión tiempo.	110
Tabla XXVIII: Porcentaje de mejora para cada indicador de la dimensión satisfacción.....	111
Tabla XXIX: Porcentaje de mejora para cada indicador de la dimensión calidad.....	111
Tabla XXX: Operacionalización de variables.....	123
Tabla XXXI: Matriz de consistencia.....	124
Tabla XXXII: Cuestionario para la variable independiente	125
Tabla XXXIII: Ficha de observación para la dimensión tiempo (pretest)	127
Tabla XXXIV: Ficha de observación para la dimensión tiempo (postest)	127
Tabla XXXV: Resumen de procesamiento de casos para cuestionario 1	135
Tabla XXXVI: Estadística de fiabilidad para cuestionario 1	135
Tabla XXXVII: Estadísticas de elementos para cuestionario 1	135
Tabla XXXVIII: Resumen de procesamiento de casos para cuestionario 2	136
Tabla XXXIX: Estadística de fiabilidad para cuestionario 2	136
Tabla XL: Estadística de elementos para cuestionario 2.....	136
Tabla XLI: Pruebas de aceptación de usuario	143
Tabla XLII: Capex y opex.....	145
Tabla XLIII: Especificación de caso de uso gestionar usuarios.....	148
Tabla XLIV: Especificación de caso de uso registrar reclamos.....	150
Tabla XLV: Especificación de caso de uso registrar producto	152

ÍNDICE DE FIGURAS

Fig. 1: Arquitectura MVC (Modelo-Vista-Controlador).....	8
Fig. 2: Arquitectura híbrida apoyada con MVVM	8
Fig. 3: Modelo de desarrollo de software en cascada	9
Fig. 4: Ubicación geográfica de Nikos Pizza	17
Fig. 5: Modelo del proceso de registro de un pedido	23
Fig. 6: Modelo del proceso de registro de un pedido (delivery)	24
Fig. 7: Modelo del proceso de generación de reportes.....	24
Fig. 8: Modelo del proceso de registro de pagos de los pedidos.....	25
Fig. 9: Actores del negocio	25
Fig. 10: Casos de uso del negocio	26
Fig. 11: Diagrama de casos de uso del negocio	26
Fig. 12: Actores de la aplicación web y móvil.....	27
Fig. 13: Casos de uso de la aplicación web y móvil	27
Fig. 14: Diagrama de casos de uso de la aplicación web y móvil.....	28
Fig. 15: Diagrama de caso de uso registrar pedido	28
Fig. 16: Diagrama de caso de uso registrar pago	30
Fig. 17: Diagrama de caso de uso generar reporte de ventas	32
Fig. 18: Diagrama de componentes aplicación web.....	35
Fig. 19: Diagrama de componentes aplicación móvil	36
Fig. 20: Diagrama de clases de la aplicación web y móvil	41
Fig. 21: Diagrama de secuencia para el dashboard	42
Fig. 22: Diagrama de secuencia para listar categorías	43

Fig. 23: Diagrama de secuencia para crear categoría.....	43
Fig. 24: Diagrama de secuencia para listar clientes	44
Fig. 25: Diagrama de secuencia para crear cliente.....	44
Fig. 26: Diagrama de secuencia para listar proveedores	45
Fig. 27: Diagrama de secuencia para crear proveedor	45
Fig. 28: Diagrama de secuencia para listar productos.....	46
Fig. 29: Diagrama de secuencia para crear producto	46
Fig. 30: Diagrama de secuencia para crear un pedido.....	47
Fig. 31: Diagrama de secuencia para listar pedidos	47
Fig. 32: Diagrama de secuencia para pagar pedido.....	48
Fig. 33: Diagrama de secuencia para listar premios.....	48
Fig. 34: Diagrama de secuencia para crear premios.....	49
Fig. 35: Diagrama de secuencia para reportes por día	49
Fig. 36: Diagrama de secuencia para reporte por rango de fechas.....	50
Fig. 37: Diagrama de secuencia para listar reclamos	50
Fig. 38: Diagrama de secuencia para registrar reclamo	51
Fig. 39: Diagrama de secuencia para modificar información de la empresa	51
Fig. 40: Diagrama de secuencia de roles y permisos	52
Fig. 41: Diseño conceptual de la base de datos.....	53
Fig. 42: Instalación de Laravel	54
Fig. 43: Ubicación de controladores en el framework Laravel	54
Fig. 44: Ubicación de modelos en el framework Laravel	55
Fig. 45: Ubicación de las vistas en el framework Laravel	55
Fig. 46: Código del controlador de categorías	56

Fig. 47: Código de crear categoría	56
Fig. 48: Vista de lista de categorías.....	57
Fig. 49: Vista de crear categoría.....	57
Fig. 50: Código de controlador de proveedores	57
Fig. 51: Código de la vista de proveedores	58
Fig. 52: Vista implementada de proveedores (listar proveedores).....	58
Fig. 53: Vista implementada de proveedores (crear proveedor).....	59
Fig. 54: Código del modelo de producto	59
Fig. 55: Código de controlador de productos	60
Fig. 56: Código de vista del index de productos	60
Fig. 57: Código de vista de crear productos.....	61
Fig. 58: Vista implementada de productos (listar productos).	62
Fig. 59: Vista implementada de productos (crear producto).....	62
Fig. 60: Código de controlador de clientes	63
Fig. 61: Código de vista de clientes	63
Fig. 62: Vista implementada de clientes (listar clientes).	64
Fig. 63: Vista implementada de clientes (crear cliente).....	64
Fig. 64: Código del modelo orden.....	65
Fig. 65: Código del controlador orden	65
Fig. 66: Vista implementada de pedidos	66
Fig. 67: Vista implementada de crear un pedido.....	66
Fig. 68: Código del modelo venta	67
Fig. 69: Código del controlador de venta	67
Fig. 70: Vista implementada de ventas (listar ventas)	68

Fig. 71: Vista implementada de ventas (crear venta)	68
Fig. 72: Vista de cobrar por mesa	68
Fig. 73: Vista de pedido encontrado para cobrar por mesa	68
Fig. 74: Modal para seleccionar el método de pago.....	69
Fig. 75: Código de la vista para la información de la empresa.	69
Fig. 76: Vista implementada para la información de la empresa.	70
Fig. 77: Vista implementada para actualizar la información de la empresa.	70
Fig. 78: Código del controlador de reclamos	71
Fig. 79: Código de la vista de reclamos	71
Fig. 80: Vista implementada para listar reclamos	72
Fig. 81: Vista implementada para crear reclamos.	72
Fig. 82: Código del modelo de premios.	73
Fig. 83: Código del controlador de premios	73
Fig. 84: Vista implementada para listar premios.....	74
Fig. 85: Vista implementada para crear premios.....	74
Fig. 86: Esquema de la base de datos.....	74
Fig. 87: Vista del index principal de la aplicación	75
Fig. 88: Vista de inicio de sesión	75
Fig. 89: Vista del dashboard, parte superior.....	76
Fig. 90: Vista del dashboard, parte inferior.....	76
Fig. 91: Vista de reporte de ventas por rango de fecha.	77
Fig. 92: Vista de roles	77
Fig. 93: Vista de usuarios de la aplicación.....	77
Fig. 94: Vista del perfil del usuario.....	78

Fig. 95: Ubicación del icono para preguntar a la I.A.	78
Fig. 96: Vista del modal para preguntar a la I.A. sobre las ventas.....	79
Fig. 97: Vista de la respuesta de la I.A. sobre las ventas	79
Fig. 98: Estructura inicial del proyecto en Android.....	80
Fig. 99: Código de splashscreen.....	81
Fig. 100: Vista final de splashscreen.....	81
Fig. 101: Código de implementación del webview	82
Fig. 102: Vista de inicio de sesión en la aplicación móvil.....	82
Fig. 103: Vista de los productos más vendidos en móvil.....	83
Fig. 104: Vista de todas las opciones en la aplicación móvil.....	83
Fig. 105: Prueba de la vista del dashboard.....	84
Fig. 106: Prueba de la vista de categorías	85
Fig. 107: Prueba de la vista de productos.....	85
Fig. 108: Prueba de la vista de ventas	85
Fig. 109: Prueba de búsqueda por mesa para cobrar en ventas.....	86
Fig. 110: Diagrama de despliegue.....	86
Fig. 111: Configuración de variables secretas.	87
Fig. 112: Resultados de GitHub Actions.....	87
Fig. 113: Vista para generar apk firmado.....	88
Fig. 114: Vista para seleccionar tipo App Bundle	88
Fig. 115: Vista de generación de Key Signed.....	89
Fig. 116: Vista final para seleccionar tipo de app a generar	89
Fig. 117: Visa de como subir app a la PlayConsole.....	90
Fig. 118: Vista de despliegue de la app en prueba interna	90

Fig. 119: Proceso modelado de registro de pedido (postest).....	93
Fig. 120: Diagrama del proceso de registro de pedido por delivery (postest)	94
Fig. 121: Proceso modelado de elaboración de reportes de ventas (postest).	94
Fig. 122: Modelado del proceso de registro de pagos (postest).....	95
Fig. 123: Porcentaje de respuestas para los indicadores de la dimensión calidad del software	109
Fig. 124: Ficha 1 de validación del instrumento (variable independiente)	131
Fig. 125: Ficha 2 de validación del instrumento (variable independiente)	132
Fig. 126: Ficha 1 de validación para el cuestionario (variable dependiente)	133
Fig. 127: Ficha 2 de validación para el cuestionario (variable dependiente)	134
Fig. 128: Ficha 1 de validación para ficha de observación	137
Fig. 129: Ficha 2 de validación para ficha de observación	138
Fig. 130: Número de lecturas según duración del ciclo	139
Fig. 131: Prueba de normalidad para registro de pedido.....	140
Fig. 132: Prueba de normalidad para reporte de ventas	140
Fig. 133: Prueba de normalidad para registro de pago.....	140
Fig. 134: Prueba de normalidad para satisfacción.....	141
Fig. 135: Prueba de normalidad para calidad.....	142
Fig. 136: Caso de uso gestionar usuarios	148
Fig. 137: Caso de uso registrar reclamos	149
Fig. 138: Caso de uso registrar productos	151

RESUMEN

La investigación abordó el problema del proceso manual en las ventas de la empresa Nikos Pizza, el cual dependía del registro en cuadernos, generando demoras en la atención de los pedidos, errores en las especificaciones de los pedidos, dificultades en la consolidación de reportes de ventas e insatisfacción en los trabajadores inmersos en el proceso de ventas. Frente a esta situación, se planteó como objetivo determinar la influencia de la implementación de una aplicación web y móvil en la mejora del proceso de ventas. Metodológicamente, se desarrolló una investigación de tipo aplicada, con diseño preexperimental, nivel explicativo y enfoque inductivo. La variable de estudio se analizó en tres dimensiones: tiempo, satisfacción y calidad, para lo cual se recopiló información antes y después de la implementación del sistema propuesto, utilizando encuestas dirigidas a los trabajadores y la técnica de observación para registrar el desempeño del proceso. Los resultados mostraron mejoras significativas: una reducción promedio del 59.48% en el tiempo empleado en el registro y control de ventas, un incremento del 88.74 % en la satisfacción percibida por los trabajadores y un aumento del 137.82 % en la calidad de los reportes generados. Estos hallazgos evidencian que la automatización de los procesos manuales mediante el uso de aplicaciones tecnológicas no solo contribuye a optimizar la eficiencia operativa, sino que también incrementa la motivación del personal que está inmerso en el proceso de ventas. En conclusión, la implementación de la aplicación web y móvil influyó positivamente en todas las dimensiones evaluadas, validando su efectividad como una solución tecnológica capaz de transformar y modernizar los procesos manuales de ventas en la empresa Nikos Pizza.

Palabras clave: aplicación web y móvil, reducción de tiempos, satisfacción laboral, calidad de reportes

ABSTRACT

This research addressed the problem of the manual sales process at Nikos Pizza, which relied on notebook recording, generating delays in order processing, errors in order specifications, difficulties in consolidating sales reports, and dissatisfaction among employees involved in the sales process. Given this situation, the objective was to determine the influence of implementing a web and mobile application on improving the sales process. Methodologically, an applied research study was developed with a pre-experimental design, an explanatory level, and an inductive approach. The study variable was analysed in three dimensions: time, satisfaction, and quality. Data was collected before and after the implementation of the proposed system, using employee surveys and observation techniques to record process performance. The results showed significant improvements: an average 59.48% reduction in the time spent recording and controlling sales, an 88.74% increase in employee satisfaction, and a 137.82% increase in the quality of the reports generated. These findings demonstrate that automating processes through the use of technological applications not only contributes to optimizing operational efficiency but also increases the motivation of the staff involved in the sales process. In conclusion, the implementation of the web and mobile application positively influenced all the evaluated dimensions, validating its effectiveness as a technological solution capable of transforming and modernizing manual sales processes at Nikos Pizza.

Key Words: Web and mobile application, time reduction, job satisfaction, report quality

CAPÍTULO I. INTRODUCCIÓN

A nivel mundial, los procesos de ventas enfrentan múltiples desafíos debido a la creciente complejidad y volumen de datos generados diariamente; y aunque, las empresas luchan para gestionar estos datos de manera adecuada no logran reducir las barreras socio-técnicas [1]. Asimismo, pocos son los gerentes de ventas que concentran esfuerzos en la digitalización y la automatización del proceso de ventas para así maximizar los ingresos [2]. Por ejemplo, en Indonesia, solo el 50% de los directores de ventas de la región de Yakarta que están inmersos en la transformación digital priorizan la efectividad en los procesos, en cambio 62.4% no apuesta por la automatización de las ventas como su principal objetivo [3].

En Latinoamérica la situación es alarmante, las pymes, que son el motor principal de las economías, enfrentan fuertes desafíos en su transformación digital, un ejemplo de ello son Chile y Costa Rica, donde aproximadamente el 40% de las pymes encuentran desafíos en los procesos de ventas, asimismo, la toma de decisiones estratégicas en el proceso de ventas en base a datos precisos es vital para el crecimiento continuo de las empresas; sin embargo, en Brasil y México, el 65% y 67% de las empresas, respectivamente, no optimizan sus datos para tener información precisa al momento de tomar decisiones [4]. Además, estar inmerso en el proceso de ventas es agotador y estresante, lo que conlleva a una alta rotación de empleados. Según lo muestra la encuesta global de actitudes de beneficios de América Latina de la consultora WTW realizada a fines de 2021 y comienzos de 2022, cerca del 44% de los trabajadores no planean quedarse en su empleo actual debido a estas condiciones adversas [5].

En el Perú, la problemática es crítica, el 98% de los restaurantes son pequeños negocios que su proceso de ventas lo realizan de forma manual y sin contar con sistema alguno, además tener información exacta y oportuna es un desafío constante para ellos afectando su capacidad para competir [6]. Las empresas que quieren ser competitivas necesitan satisfacer a sus clientes brindando atención oportuna, porque los clientes que tengan una mala experiencia no regresan, según el artículo “Calidad de servicio en la decisión de compra en centros comerciales de Perú”, el 60% de los clientes peruanos consideran que la calidad de servicios aún no es buena [7]. Además, se debe tener en cuenta que solo el 50% de los clientes usan el libro de reclamaciones frente a una discrepancia, lo que indica un bajo nivel de confianza en los mecanismos de resolución de problemas.

En la ciudad de Cajamarca, el registro de los pedidos en papel es muy común en las pequeñas empresas, tal es el caso de la empresa Nikos Pizza que se dedica al rubro restaurante-pizzería y en donde se expende pizzas y lasaña para el público en general. En la empresa antes mencionada, el registro de información en papel para gestionar el proceso de ventas genera diversos inconvenientes, tales como demora en la atención al cliente y también errores en las especificaciones de los pedidos. Además, cuando se pretende elaborar informes diarios o semanales sobre el proceso ventas, se encuentran los datos de forma desordenada e incluso faltante. Por tal motivo esta investigación buscar conocer ¿De qué manera influye la implementación de una aplicación web y móvil en el proceso de ventas en la empresa Nikos Pizza? Además, se propuso la siguiente hipótesis: La implementación de una aplicación web y móvil influye de manera positiva en el proceso de ventas de la empresa Nikos Pizza.

Esta investigación tiene una relevancia práctica en la empresa Nikos Pizza debido a que la implementación de una aplicación web y móvil para el proceso de ventas, permite a la empresa atender a sus clientes de manera más rápida al reducir el tiempo en el registro de los pedidos, en el registro de los pagos y evita conflictos con los clientes al reducir los errores en las especificaciones, a su vez, esta implementación incrementa la satisfacción de los trabajadores al disminuir su carga de trabajo manual, y también aumenta la calidad de los reportes de ventas. Por otra parte, a nivel académico, la presente investigación es importante debido a que aporta conocimiento sobre el desarrollo de una aplicación web y móvil utilizando Laravel como framework para la aplicación web y en móvil utiliza un webview para conectar la aplicación móvil con la web, asimismo, el desarrollo de la aplicación se basa en el modelo en cascada, un modelo tradicional adecuado para desarrollo de software con un resultado final bien definido.

Por otro lado, tiene como alcance reducir el tiempo de registro de un pedido como también el tiempo de registro de un pago, aumentar la satisfacción de los trabajadores involucrados en el proceso de ventas, e incrementar la calidad de los reportes sobre las ventas. Cabe mencionar que el estudio solo abarca el proceso de ventas, ya que este proceso es crucial para la empresa; la aplicación móvil se implementa únicamente para el sistema operativo Android, pero no se ha contemplado el uso directo por parte de los clientes sino solo para los trabajadores. Para el desarrollo de la aplicación web y móvil se utiliza el modelo en cascada, asimismo, la base de datos es MySQL y el framework para la aplicación web es Laravel. En la aplicación móvil se usa Kotlin y Jetpack Compose y a través de un webview se accede a la aplicación web. En el despliegue la aplicación web se sube a un hosting compartido y la aplicación móvil a la

PlayStore en una prueba interna para que esté disponible solo para los trabajadores de la empresa.

El objetivo general de esta investigación es determinar la influencia de la implementación de una aplicación web y móvil en el proceso de ventas, y para ello se plantea los objetivos específicos que son determinar la influencia de la implementación de la aplicación web y móvil en el tiempo de registro de pedidos, registro de pagos y en la elaboración de reportes, en la satisfacción de los trabajadores inmersos en el proceso de ventas y en la calidad de los reportes de ventas. Las limitaciones se presentan en los trabajadores que están acostumbrados al registro de los pedidos de forma manual, y no todos los trabajadores están dispuestos a utilizar herramientas tecnológicas para dicha actividad, lo que puede generar resistencia al cambio. Por otra parte, el despliegue de la aplicación web y móvil en internet implica un costo mensual y/o anual.

Esta investigación está organizada en capítulos de la siguiente manera; capítulo I, se encuentra la introducción en donde se describe el problema, hipótesis, justificación, el alcance, los objetivos y la descripción de los contenidos; en el capítulo II, se describe las diferentes investigaciones previas en el tema que se está investigando, asimismo se incluye los fundamentos teóricos y definición de términos básicos. En el capítulo III, se describe el procedimiento de la presente investigación siguiendo un orden y además como se trata, analiza y presenta los resultados obtenidos; en el capítulo IV, se describe y se contrasta los resultados encontrados con los datos que están en el marco teórico; en el capítulo V, se determina las conclusiones de la investigación y las recomendaciones que se considera necesarias a partir de la investigación realizada. Y finalmente, en anexos, se encuentra los formatos de las encuestas y fichas de observación con sus respectivas validaciones que se han utilizado en el desarrollo de la investigación.

CAPÍTULO II. MARCO TEÓRICO

2.1 ANTECEDENTES TEÓRICOS DE LA INVESTIGACIÓN

2.1.1 ANTECEDENTES INTERNACIONALES

Rivera [8] en su investigación titulada “modelo de datos para la gestión de ventas de la empresa MegaHome”, tuvo como objetivo general desarrollar un modelo de datos para la gestión de ventas de la empresa MegaHome. Esta investigación de tipo cuantitativa con diseño cuasiexperimental tuvo como resultados que la implementación de un modelo de datos con dashboards interactivos y reportes dinámicos permiten a los trabajadores inmersos en el proceso de ventas explorar datos de ventas de manera intuitiva y detallada, también la muestra del top 5 de productos más vendidos y esto permite visualizar rápidamente el ingreso total por cada producto. En esta investigación se concluye que el modelo de datos en el área de ventas de MegaHome es crucial y exitoso en la eficiencia de las operaciones de la empresa, con el modelo se ha logrado que los trabajadores del área de ventas cuenten con informes iterativos para gestionar y analizar datos relevantes de ventas y así teniendo la facilidad para tomar decisiones de forma más rápida y precisa. Esta investigación es de utilidad para la tesis en desarrollo porque muestra la importancia de contar con reportes de ventas de calidad.

Cayo y Seracapa [9] en su investigación titulada “desarrollo de un sistema web de gestión de ventas para la empresa EcoMejia mediante la aplicación de inteligencia de negocios”, tuvieron como objetivo principal diseñar un sistema web de gestión de ventas integrando herramientas de inteligencia de negocios en línea. Esta investigación de tipo aplicada tuvo como resultados que el 78% de los clientes perciben que el sistema web aumenta la calidad de los negocios y por ende de las ventas. Las conclusiones de esta investigación evidenciaron que los reportes y gráficos que genera el sistema brindan información clara y precisa sobre cuáles son los productos más vendidos, así como sus categorías y las fechas en que se realizaron las ventas. Esta investigación se relaciona con la tesis en desarrollo al evidenciar que los reportes de ventas son esencial en un sistema web para tener información de calidad.

Quintana y Cunuhay [10] en su investigación titulada “análisis comparativo de aplicaciones web de control de ventas en restaurantes para la toma de decisiones”, tuvieron como objetivo general la creación de una plataforma digital que permita un control más riguroso de las ventas, la generación de reportes detallados y la mejora de la productividad mediante la visualización

de datos clave. Esta investigación tuvo un enfoque mixto, combinando investigación bibliográfica, de campo y tecnológica en la cual se obtuvo como resultados la reducción del 70% en tiempos de registro, 87% de disminución en errores operativos y un 90% de satisfacción entre usuarios. A la vez, los beneficios fueron de un 85% menos en costo de material de oficina y un 60% de reducción en horas extras del personal. Se concluyó que la implementación de un sistema web representó una verdadera transformación en la gestión operativa del negocio y que el registro de ventas como la elaboración de informes superó las limitaciones identificadas en los métodos manuales. Esta investigación se relaciona con la tesis en desarrollo al evidenciar la disminución de tiempo de registro de ventas y la elaboración de informes.

2.1.2 ANTECEDENTES NACIONALES

Huamán y Alvarado [11] en su tesis de pregrado titulada “elaboración de un sistema web para la optimización de la gestión de pedidos en la empresa Chifa Kam Lun de Ica”, tuvieron como objetivo principal la elaboración de un sistema web mediante la metodología clásica para la optimización de la gestión de pedidos. Esta investigación con un diseño experimental tuvo como resultados, la reducción del tiempo de registro de los pedidos en un 3.38 minutos de tiempo promedio, es decir, al inicio se tardaba 5.07 minutos en registrar el pedido y ahora solo 1.29 minutos aproximadamente; asimismo, esta investigación mostró una reducción del porcentaje de los reclamos, pasando inicialmente de un 44% a un 8% con el sistema ya implementado. Es así que se concluyó que el sistema web repercute de forma positiva en la empresa en mención en lo que respecta a atención de los clientes, por otra parte, se concluyó que el sistema web está al 100% adecuado al entorno y necesidades de la empresa. Esta investigación se relaciona con la tesis en desarrollo al mostrar el efecto de un sistema web en el tiempo de registro de pedidos y en la disminución del número de reclamos.

Sakihara [12] en su tesis titulada “influencia de la aplicación de un sistema informático en el proceso de ventas en una empresa del sector gastronómico en el Perú”, tuvo como objetivo principal determinar la influencia de la puesta en marcha de un sistema informático en el proceso de ventas en la entidad TAM Inversiones. Esta investigación de tipo aplicada contó con una muestra de 252 clientes y tuvo como resultados la disminución del tiempo de toma de pedido en un 58.99%, la disminución del tiempo del despacho de un pedido en un 55.67% y la disminución del tiempo de entrega de los montos a pagar en un 75.76%. Se concluyó que la implementación de un sistema informático reduce significativamente en el tiempo de espera de atención de los comensales. Por otra parte, se concluyó que el sistema reduce los errores

mensuales de los pedidos que van desde un 56% hasta 79%. Esta investigación es importante con la presente tesis en desarrollo al evidenciar que un sistema reduce el tiempo de atención de los pedidos y que reduce los errores en los pedidos.

Herrera [13] en su tesis titulada “sistema web para optimizar la gestión de pedidos de la empresa agrícola Viña Vieja Viña Santa Isabel S.A.C”, tuvo como objetivo general optimizar la gestión de pedidos de la empresa mediante la implementación de un sistema web apoyado en el algoritmo AHP. Esta investigación tuvo como resultados la disminución del tiempo de atención de un pedido en un 75% aproximadamente, antes de la implementación del sistema web el tiempo fue de 5 a 7 días, luego de la implementación es de 2 a 3 días, también se logró disminuir el tiempo de envío del pedido hacia Lima pasando de 3 días a realizarlo en 1 minuto y se logró aumentar la satisfacción del cliente en un 85%. Es así que se concluyó que la disminución del tiempo de atención del pedido en un 75% contribuye a tener información inmediata de los pedidos, además, que el tiempo de envío hacia Lima se redujo en un 95%. Esta investigación es de utilidad con el trabajo en desarrollo debido a que permite comparar el tiempo que toma registrar un pedido antes y después de implementar un sistema web.

Gaona [14] en su investigación titulada “diseño e implementación de una aplicación móvil para mejorar el proceso de venta de líneas prepago en una empresa de telecomunicaciones”, tuvo como objetivo general el determinar el aumento de los ingresos en ventas a través de la implementación de la aplicación móvil. Los resultados mostraron que el tiempo de operación de un pedido se redujo de 15 a 5 minutos en promedio; además, se muestra que las ventas a través de la aplicación móvil son de 164.696 soles mientras que en un centro de atención a los clientes es de 159.574. En esta investigación concluyó que el marco de trabajo SCRUM ha sido esencial para el correcto desarrollo de la aplicación y que la aplicación está desarrollada para facilitar la adquisición de nuevas líneas Prepago. Esta investigación es importante con la tesis en desarrollo al evidenciar como una aplicación móvil influye en la reducción del tiempo de operación de una venta.

2.1.3 ANTECEDENTES LOCALES

Guerrero [15] en su tesis de pregrado titulada “efecto de la implementación de un sistema de información en los procesos de atención al cliente en el recreo campestre “La Querencia”, Cajamarca 2022”, tuvo como objetivo general determinar el efecto de la implementación de un sistema de información en los procesos de atención al cliente. Esta investigación de tipo explicativa tuvo como resultados, la reducción del tiempo de atención del cliente en un 54%, el

aumento de la calidad de atención en un 38% y el aumento de la satisfacción de los trabajadores en un 48%. Asimismo, en esta investigación se concluyó que el sistema influye de manera positiva en el proceso de atención al cliente, y con respecto a la atención de los clientes se concluyó que ha mejorado la capacidad de respuesta del restaurante y se ha logrado atender a más clientes, de igual forma se concluyó que los trabajadores están más satisfechos y se ha reducido en un 46.15% el número de quejas y/o reclamos. Esta investigación es de utilidad para la tesis en desarrollo porque muestra el porcentaje de reducción del tiempo de atención a un cliente como también el porcentaje de satisfacción de los trabajadores con el sistema.

2.2 BASES TEÓRICAS

2.2.1 APPLICACIÓN WEB Y MÓVIL

Se conoce como aplicación web al software que se ejecuta en el navegador y no es necesario instalar un software adicional en el sistema operativo para su correcto desempeño. Las aplicaciones web surgen a medida que las empresas urgen mostrar información en tiempo real como puede ser un carrito de compras, búsqueda y filtrado de productos, mensajería instantánea y redes sociales [16]. Asimismo, se conoce como aplicación móvil al software creado para ejecutarse en teléfonos o tablets y que hacen uso de los recursos de estos para realizar sus actividades para las que fueron diseñados [17]. Cabe mencionar que existe una diferencia clara entre una aplicación móvil y un sitio web móvil, resaltando que la aplicación móvil es aquella que se instala en el teléfono mientras que el sitio web móvil es aquel que se accede a través de un enlace en el navegador y fue diseñado en una modalidad responsive [18].

En el desarrollo de aplicaciones web, es fundamental conocer la estructura de cómo está organizado el código fuente, las librerías y los archivos de configuración. Para evitar complicaciones, se utilizan los llamados “frameworks”, cuyo propósito principal es estructurar el código del sistema de forma que la aplicación web sea mantenible y escalable. Además, los frameworks separan la aplicación en diferentes capas: la capa de presentación se encarga de las interacciones entre el software y el usuario; la capa de acceso de datos, que brinda la forma de acceder al agente de almacenamiento; y la capa de dominio, que se encarga de procesar los comandos recibidos desde la capa de presentación [19].

A su vez, en el desarrollo de software es indispensable utilizar una arquitectura de desarrollo de software, en el caso de una aplicación web una de las arquitecturas más conocidas es MVC (Model-View-Controller). Esta arquitectura organiza las aplicaciones en 3 componentes: el

modelo se encarga de los datos y la lógica de negocio, la vista se encarga de mostrar la información al usuario y el controlador actúa como intermediario procesando las entradas del usuario y coordinando la interacción entre el Modelo y Vista [20]. En la figura 1 [21], se muestra el modelo MVC propuesto por John Kouraklis.

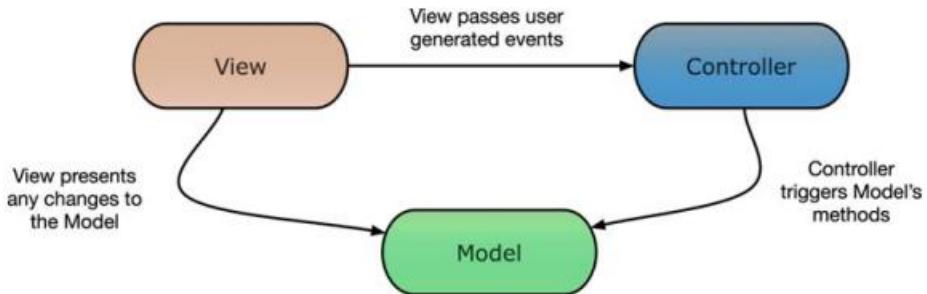


Fig. 1: Arquitectura MVC (Modelo-Vista-Controlador)

Complementariamente, para aplicaciones móviles se utiliza la arquitectura híbrida apoyada en el patrón de diseño MVVM (Model-View-ViewModel), el cual surgió como una evolución de MVC y MVP. En MVVM, el ViewModel reemplaza al Controller o Presenter, concentrando la lógica de presentación y el estado de la vista y facilitando el testing y enlace de datos bidireccional [21]. En la figura 2 [21], se muestra la arquitectura híbrida apoyada con el patrón de diseño MVVM.

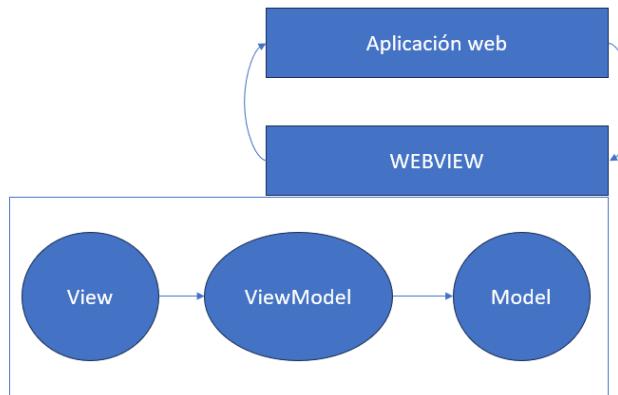


Fig. 2: Arquitectura híbrida apoyada con MVVM

Asimismo, en el desarrollo de aplicaciones móviles, una opción popular es el uso de webviews, especialmente cuando se busca reutilizar código web existente. Un WevView permite incrustar una página web dentro una aplicación nativa y esto es muy importante cuando se cuenta con una aplicación web y móvil y se requiere que las actualizaciones sean al instante sin necesidad de realizar actualizaciones en la tienda de aplicaciones que toman tiempo. Además, el uso de

WebViews reduce considerablemente el tiempo de desarrollo al aprovechar el código de la aplicación web existente [22].

Al desarrollar una aplicación, es crucial utilizar un modelo de desarrollo de software para asegurar su éxito y uno de los modelos principales es el modelo en cascada que hace referencia que los desarrolladores deben completar cada etapa en su totalidad y luego pasar a la siguiente, todo esto con el fin de evitar el solapamiento de trabajo en dos etapas. Cabe mencionar que algunos autores al modelo en cascada [24] lo conocen como metodología clásica o en cascada [23]. Existen 7 etapas del modelo en cascada; la primera etapa conocida como requisitos, en donde se establece los requisitos potenciales, el alcance y el objetivo del proyecto; la siguiente etapa es el análisis, en donde los desarrolladores determinan las especificaciones del proyecto y la función del producto en desarrollo; la tercera etapa es el diseño, en esta etapa los desarrolladores definen la arquitectura de software y otros requisitos del sistema como elección de tecnologías; en la cuarta etapa que es la implementación, los programadores desarrollan el software en unidades separadas y las van probando una a una; en la quinta que es pruebas, se combinan las unidades y prueban el sistema integrado para detectar fallos o errores; en la sexta etapa que es el despliegue, el software está disponible al público y finalmente en la séptima etapa que es el mantenimiento, los desarrolladores solucionan o reparan los problemas que se conocen durante el uso real [24]. A continuación, se muestra en la figura 3 las etapas del modelo en cascada[24].

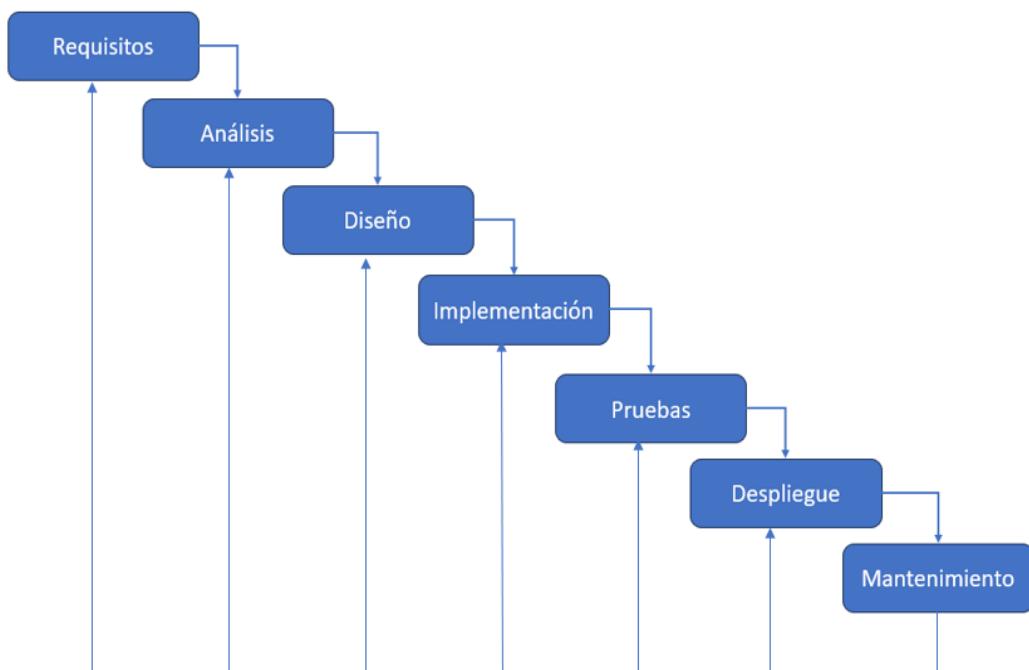


Fig. 3: Modelo de desarrollo de software en cascada

Asimismo, tanto en aplicaciones web como aplicaciones móviles, la calidad es un aspecto fundamental y para garantizarla, se puede aplicar la norma ISO/IEC 25010, que brinda las características esenciales al momento de evaluar la calidad del software [25]. En el artículo “estándares para la calidad de software” [26] se hace mención que hoy en día los usuarios son más exigentes y para ofrecer un software de calidad se debe atender las características de funcionalidad, fiabilidad, usabilidad, eficiencia y mantenibilidad. A continuación, se detalla cada uno de ellos:

- Funcionalidad: capacidad del software para satisfacer las funciones requeridas por el usuario tanto declaradas como implícitas.
- Fiabilidad: capacidad del sistema para no presentar fallos bajo condiciones normales y estar operativo en el momento que se lo requiere.
- Usabilidad: el software debe ser entendido, aprendido y operado fácilmente.
- Eficiencia: usar los recursos (CPU, memoria, almacenamiento, etc.) sin exceder de lo necesario.
- Mantenibilidad: capacidad del software para ser modificado efectiva y eficientemente debido a necesidades futuras.

Una herramienta fundamental para el control y gestión del código durante el desarrollo de aplicaciones es Git, debido a que brinda un control de versiones distribuido facilitando el seguimiento de todos los cambios realizados en el código, evitando así la pérdida de información. GitHub, por su parte, es una plataforma basada en la nube que se integra con Git y brinda almacenamiento para el código, además, permite la gestión de ramas, así como la colaboración eficiente entre miembros del equipo [27].

2.2.2 PROCESO DE VENTAS

La palabra proceso tiene su origen en el latín *processus* y significa: avance, proceso. En el ámbito empresarial se conoce como proceso al conjunto de pasos para alcanzar cierto objetivo y dentro de los diferentes tipos se encuentra el proceso productivo que consiste en el conjunto de acciones que permite convertir la materia prima, insumos y/o conocimientos en la solución final [28]. Se conoce como proceso de ventas al conjunto de pasos interrelacionados que se sigue para concretar la venta de un producto, y se inicia enfocándose en el mercado objetivo de la empresa, luego se realiza una buena presentación personal y elementos distintivos de la empresa, seguidamente se comprende las necesidades del cliente y se ofrece el producto

adecuado, luego se realiza la presentación del producto cubriendo necesidades y deseos del cliente, así se llega al cierre de ventas y finalmente al servicio post-venta [29].

Sin embargo, este proceso depende mucho de las competencias, experiencia y herramientas que tenga el trabajador que lo ejecuta. Lejos de ser un mero ejecutor de pasos, el trabajador inmerso en el proceso de ventas se convierte en el motor fundamental de cada etapa. Su capacidad para atender en menor tiempo posible, confianza durante la presentación y su capacidad para manejar objeciones determinan el éxito del cierre y la fidelización post-venta. Desde esta perspectiva, el proceso de ventas se redefine como una serie de interacciones humanas complejas mediadas por el trabajador inmerso en el proceso de ventas, por lo tanto, la satisfacción del trabajador, la disponibilidad de herramientas que agilicen su trabajo y la información de calidad se convierten en elementos indispensables para un correcto desempeño [30].

Para evaluar la variable dependiente (proceso de ventas) se ha considerado las siguientes dimensiones: tiempo, satisfacción y calidad.

En la dimensión tiempo se operacionalizó mediante el tiempo de registro de pedidos, definido como el proceso mediante el cual se ingresan los detalles de un pedido en el sistema de la empresa y el tiempo de registro de pagos, definido como el proceso mediante el cual se ingresan los detalles del pago de un pedido. Las tecnologías digitales permiten reducir significativamente este tiempo, ya que eliminan la necesidad de un registro manual y transmiten los pedidos de forma instantánea, mejorando la eficiencia operativa y la precisión del pedido, lo que a su vez mejora la experiencia del cliente [31]. Los clientes modernos están constantemente ocupados y abrumados con información, por lo que un proceso de ventas ágil no solo responde a su demanda de inmediatez, sino que también permite a las empresas maximizar su eficiencia operativa [32]. El tiempo de registro de pedidos tiene un impacto directo en la satisfacción, lealtad y retención del cliente. En el contexto del comercio electrónico, un tiempo de registro largo e inconsistente puede generar insatisfacción, quejas, cancelaciones, devoluciones y críticas negativas, afectando negativamente la experiencia del cliente y su fidelidad [33].

Asimismo, en la dimensión satisfacción se operacionalizó mediante la satisfacción laboral de los colaboradores lo cual es un factor crucial para el buen desempeño de cualquier empresa. Aunque esta puede variar significativamente de un colaborador a otro, está influenciada por las condiciones internas y la facilidad con la que los colaboradores realizan sus tareas [34]. En procesos manuales, esta satisfacción puede verse afectada negativamente, ya que la carga de

trabajo puede resultar más pesada y propensa a errores, lo que genera insatisfacción y, en última instancia, bajo rendimiento.

Para medir la dimensión calidad se consideró la calidad de los informes de ventas y que es fundamental para la gestión y desempeño comercial en cualquier empresa. La calidad en el contexto empresarial, se define como la data caracterizada por la exactitud, consistencia e integridad y que ayudan a la toma de decisiones efectivas[35]. Los reportes de ventas de calidad no solo proporcionan una visión detallada y estructurada de los datos, sino que también permiten a los gerentes tomar decisiones informadas. La calidad en la recopilación y presentación de los datos es crucial para que los gerentes puedan adaptarse a las condiciones cambiantes del mercado y las estrategias de la competencia [36]. Además, la generación de gráficos de ventas, facilita la visualización de estos datos, transformando números en representaciones visuales accesibles e intuitivas, que son fáciles de interpretar incluso para quienes no tienen experiencia analítica. La combinación de reportes detallados y gráficos interactivos mejora la capacidad de la empresa para ajustar rápidamente sus estrategias, optimizar los recursos y maximizar el rendimiento comercial [37].

Cabe mencionar que la gestión financiera distingue entre dos tipos de recursos destinados al área comercial: el CAPEX (gastos de capital) y OPEX (gastos operativos). El primero se refiere a las inversiones a largo plazo que mejoran la capacidad del equipo de ventas, el segundo se refiere a los gastos recurrentes necesarios para el funcionamiento diario [38]. En esta investigación, los reportes por día que a su vez se pueden filtrar por trabajador permiten un análisis granular de la eficiencia del OPEX, al relacionar directamente gastos operativos (horas laborales) con los resultados individuales. Simultáneamente, los reportes consolidados por rango de fechas brindan una perspectiva a mediano y largo plazo necesario para evaluar el impacto estratégico de las inversiones.

La inteligencia artificial (IA) se consolida como un aliado estratégico para el trabajador moderno. Su capacidad para analizar grandes volúmenes de datos en tiempo real permite generar recomendaciones predictivas y automatizar tareas repetitivas, además, ofrece una ventaja competitiva al permitir tener una visión más amplia en las ventas y así hacer ajustes rápidos en las estrategias de ventas. La IA no solo facilita el proceso de ventas, sino que también ofrece a los trabajadores información precisa y en tiempo real, lo que mejora la toma de decisiones y reduce el estrés laboral al disminuir la incertidumbre [39]. Además, la implementación de la inteligencia artificial ha permitido a las empresas mejorar la predicción

de la demanda, aumentar las ventas efectivas y automatizar procesos de ventas. Todo esto contribuye a un ambiente laboral más eficiente, lo que resulta en una mayor satisfacción de los empleados involucrados en el proceso de ventas. De esta manera, la integración de IA no solo beneficia a la empresa, sino también a los trabajadores [40].

2.2.3 DEFINICIÓN DE TÉRMINOS BÁSICOS

Proceso: Conjunto de pasos que transforman insumos en resultados [25].

Ventas: Intercambio de bienes o servicios por cierto valor monetario con el objetivo de satisfacer una necesidad y generar rentabilidad [41].

Influencia: Capacidad de una variable independiente para generar cambios en variables dependientes mediante diseños experimentales o cuasi-experimentales [42].

Tiempo: El tiempo que toma registrar un pedido, un pago de un pedido como también el tiempo que toma generar un reporte de ventas, este tiempo es medido en minutos y segundos.

Satisfacción: La satisfacción de los trabajadores (actitud emocional) hacia su entorno de trabajo realizando cierto proceso con herramientas y condiciones organizacionales [43].

Calidad: Grado en que los reportes cumplen con atributos de exactitud y consistencia [44].

Exactitud: Concordancia entre el valor medido y el valor real [45].

Consistencia: Uniformidad de los resultados cuando se mide varias veces [46].

Calidad del software: Grado en que un software cumple los requisitos establecidos y satisface las necesidades explícitas e implícitas [26].

Framework: Es un esquema o marco de trabajo que brinda una colección de componentes de software reutilizables que ayudan en el desarrollo de nuevas aplicaciones [47].

Laravel: Es un framework PHP de código abierto de curva de aprendizaje sencillo que brinda herramientas y recursos para crear aplicaciones web [48].

Jetstream: Es un paquete de Laravel que brinda un sistema de autenticación de usuarios y una interfaz para administrar el perfil de cada usuario. Algunas características importantes es la verificación de dos pasos, la gestión de sesiones y la protección contra ataques de fuerza bruta [49].

Kotlin: Es un lenguaje de programación creado por JetBrains y es muy popular en el desarrollo de aplicaciones Android. Es un lenguaje de tipado estático [50].

MVC: Model-View-Controller, es una propuesta de arquitectura de software que se utiliza principalmente para separar las distintas capas atendiendo a sus responsabilidades [48].

MVVM: Model-View-ViewModel, es un patrón de diseño en el desarrollo de aplicaciones móviles [21].

Jetpack Compose: Su principal característica es que cambia el paradigma del desarrollo y abre las puertas a las interfaces declarativas, es decir, la propia interfaz está conectada con un estado, y este hará que se actualice de forma automática cuando sea necesario [51].

MySQL: Es un sistema de gestión de bases de datos relacionales [52].

Eloquent: Es el ORM (Object Relational Mapping) integrado en Laravel, esto facilita la interacción con la base de datos de una manera más sencilla y elegante [53].

Webview: Es un componente que facilita la implementación de aplicaciones que necesitan mostrar contenido web y sin necesidad de abrir un navegador externo [54].

GitHub: Es un sistema de control de versiones en la nube (VCS) llamado Git, y lo más importante que es un servicio basado en la nube [55].

GitHub Actions: Es una plataforma de integración y despliegue continuo que permite compilar, hacer pruebas y/o desplegar de forma automática [56].

Hosting Compartido: Es un modelo de hosting en la nube que sirve para alojar páginas web y tienen acceso compartido a un conjunto de recursos en el servidor [57].

VPS: Servidor Privado Virtual, es una máquina que aloja todo el software y los datos necesarios para ejecutar una aplicación web. Se conoce como virtual debido a que solo consume una parte de los recursos físicos subyacentes del servidor [58].

Composer: Es un manejador de paquetes para PHP que proporciona un estándar para administrar, descargar e instalar dependencias y librerías [59].

Usuario: Es la persona que usa la aplicación web y móvil [60].

Responsive: Un enfoque del diseño web que ayuda a que una aplicación web se ajuste a los diferentes tamaños de pantalla [61].

Rol: Conjunto de permisos que definen lo que puede hacer un usuario en una aplicación [62].

CAPÍTULO III. MATERIALES Y MÉTODOS

La presente investigación se desarrolló en la empresa Nikos Pizza de la ciudad de Cajamarca, ubicada en el Jr. Reyna Farge N° 175, el tiempo de la investigación abarcó 4 meses. Asimismo, para el desarrollo de la aplicación web y móvil se utilizó el modelo en cascada.

Descripción de la empresa

Nikos Pizza es una empresa dedicada a la elaboración y venta de pizzas artesanales, fundada hace 6 años en la ciudad de Cajamarca. La empresa, con RUC 20611899018, se ha destacado en el mercado local por ofrecer productos de alta calidad, elaborados con ingredientes frescos ganando la preferencia de sus clientes. Su giro de negocio se centra en la venta de pizzas artesanales, y cuenta con un equipo de 5 empleados altamente capacitados y el número de contacto es 975180855. A lo largo de estos años, la empresa ha logrado posicionarse como un referente en el sector de la gastronomía local, brindando a sus clientes un servicio de calidad y una experiencia culinaria única.

Misión

Somos una empresa Cajamarquina que brinda servicios de comida rápida dedicada a la elaboración, venta de pizzas y pastas artesanales horneadas a la leña con ingredientes de calidad. Contamos con un sitio acogedor y agradable. Asimismo, tenemos un personal motivado y capacitado para brindar a nuestros clientes un excelente servicio.

Visión

Lograr ser una empresa líder de comida rápida en toda la región Cajamarca, haciendo diferencia en la calidad, sabor, atención y valor agregado en cada una de nuestras pizzas y pastas artesanales horneadas a leña, a través de un buen trato a nuestros colaboradores lograr superar las expectativas de nuestros clientes.

Valores

Responsabilidad: Cumplimos con estándares de higiene, puntualidad y atención para garantizar la confianza de nuestros clientes.

Calidad de servicio: Ofrecemos una experiencia excepcional, desde la preparación de la pizza hasta la entrega, superando expectativas.

Respeto: Valoramos a cada cliente y colaborador, fomentando un trato amable, inclusivo y profesional en todo momento.

Compromiso: Nos esforzamos por mejorar continuamente, manteniendo la frescura de nuestros ingredientes y la satisfacción de quienes nos eligen.

Ubicación geográfica

Ubicada en la ciudad de Cajamarca, a poca distancia del Banco BCP ubicado en el jr. Sucre, como se observa en el mapa mostrado en la figura 4.



Fig. 4: Ubicación geográfica de Nikos Pizza

Situación actual de la empresa

En la empresa Nikos Pizza, dentro del proceso de ventas algunas actividades se realizan de forma manual, como el registro de los pedidos. Se ha encontrado que el mesero, escribe el pedido en un cuaderno de apuntes o inclusive a veces solo memoriza el pedido y luego de informar a cocina lo registran en el cuaderno de ventas con el cual cuenta la empresa, pero a veces no lo realizan de manera correcta o inclusive no se registra el precio final ni el método de pago. Otra debilidad encontrada es al momento de elaborar los reportes de ventas, la búsqueda de la información tarda mucho tiempo y algunas veces no se encontraba completo, además los trabajadores no se encontraban satisfechos con el registro manual de los pedidos, para ellos escribir y detallar les generaba incomodidades.

3.1 PROCEDIMIENTO

Esta investigación fue de tipo aplicada [63], esto debido a que la investigación se enfocó en solucionar un problema concreto e inmediatamente. Asimismo, el nivel fue explicativo [42], porque se enfocó en entender la relación de causa y efecto en la implementación de la aplicación

web y móvil en el proceso de ventas de la empresa. El diseño de la investigación fue experimental – preexperimental [42] debido a que el control es mínimo y solo se realizó en un solo grupo, además se aplicó una prueba antes y otra después de la implementación de la aplicación web y móvil y luego se verificó la influencia. Cabe mencionar que el método de investigación es inductivo [64] debido a que se partió del estudio de casos particulares y se llegó a conclusiones generales. En el anexo 1 se adjunta la matriz de operacionalización de variables para tener un enfoque general de la investigación y en el anexo 2 se adjunta la matriz de consistencia para ver la alineación entre el problema, los objetivos y la hipótesis.

Por otra parte, la población de estudio estuvo representada por la totalidad de los trabajadores involucrados en el proceso de ventas de la empresa (5 trabajadores) y la muestra fue una muestra poblacional debido a que los trabajadores de la empresa son menos de 30, y la unidad de análisis es cada trabajador involucrado en el proceso de ventas.

Para la recolección de datos se utilizaron dos técnicas principales: encuesta y observación. La encuesta permite obtener y elaborar datos rápidamente [65], mientras que la observación implica adentrarse profundamente en la situación y sobre todo prestar atención a cada detalle, suceso, evento o interacción [42].

Para la variable independiente (aplicación web y móvil) se utilizó la encuesta como técnica y el cuestionario como instrumento para medir las dimensiones de funcionalidad, fiabilidad, usabilidad, eficiencia y mantenibilidad (ver anexo 3). En la variable dependiente (proceso de ventas), para la dimensión tiempo se utilizó la observación como técnica y la ficha de observación como instrumento (ver anexo 4). Para las dimensiones satisfacción y calidad se aplicó la encuesta como técnica y el cuestionario como instrumento (ver anexo 5).

En cuanto al proceso para recolectar los datos, se elaboró y se diseñó el instrumento de recolección de datos para la variable independiente (aplicación web y móvil), construyendo un cuestionario en el cual se incluyó preguntas cerradas con una escala de Likert de 5 puntos (1: Muy en desacuerdo a 5: Muy de acuerdo), garantizando así que las preguntas fueran claras y específicas para cada dimensión. Luego se procedió a realizar su validación por expertos (ver anexo 6) y se evaluó su confiabilidad mediante el cálculo del coeficiente Alfa de Cronbach, obteniendo un valor superior al 0.801 (ver anexo 8). Luego se aplicó el cuestionario a 3 expertos entre las fechas 07 y 21 de julio a través de formularios de Google, además, se le brindó acceso a la aplicación web y móvil con una cuenta de prueba para que puedan probar todas las funcionalidades.

Para la variable dependiente (proceso de ventas) se diseñó un cuestionario con preguntas cerradas en una escala de Likert de 5 puntos como en el anterior cuestionario garantizando así que las preguntas fueran claras y específicas para cada dimensión. Seguidamente se realizó su validación por expertos (ver anexo 7) para garantizar la pertinencia y claridad de las preguntas, a la vez se evaluó su confiabilidad mediante el coeficiente de Alfa de Cronbach obteniendo un valor de 0.814 (ver anexo 9). Este cuestionario fue aplicado a los trabajadores antes y después de la implementación de la aplicación web y móvil y estuvo guiado por el investigador para resolver dudas y garantizar respuestas sinceras.

Asimismo, la ficha de observación se diseñó específicamente para medir el tiempo promedio requerido en el proceso de registro de pedidos, en el registro de pagos y en la elaboración de reporte de ventas, tanto antes como después de la implementación de la aplicación web y móvil. Esta ficha fue validada por expertos (ver anexo 10), se observaron a 5 trabajadores realizar el registro de los pedidos de forma manual en un periodo de 2 semanas y se utilizó un cronómetro para medir la precisión en el tiempo. Esto también fue realizado luego de la implementación de la aplicación web y móvil bajo las mismas condiciones.

Los datos obtenidos de las encuestas y la observación fueron recopilados y registrados en una hoja de cálculo. Posteriormente, esos datos se exportaron al software IBM SPSS versión 26 (versión de prueba) para el respectivo análisis estadístico. Asimismo, en la parte de análisis se utilizó la estadística descriptiva para evaluar cada dimensión, así como también para verificar las diferencias entre las fases “pretest” y “posttest”, se aplicó pruebas de normalidad (Shapiro-Wilk) para verificar la normalidad de los datos y para contrastar la hipótesis se aplicó prueba t-Student para las muestras relacionadas con distribución normal, y se aplicó prueba de Wilcoxon para las variables que no cumplieron el supuesto de normalidad o que las muestras eran muy pequeñas. El nivel de significancia establecido fue de $\alpha = 0.05$ cuyos resultados detallados se presentan en el Capítulo IV.

Luego de tener claro la situación actual de la empresa se procedió a la implementación de la aplicación web y móvil siguiendo las etapas del modelo en cascada. A continuación, se describe cada una de las etapas.

3.1.1 REQUISITOS

Siguiendo el modelo de desarrollo de software en cascada, se inició el proyecto con la etapa de requisitos que incluyó la definición del objetivo y el alcance de la aplicación web y móvil. Para ello, se estableció un proceso estructurado de recolección y validación de requerimientos.

Técnicas de recolección de requisitos aplicadas:

Para obtener los requisitos se aplicaron las entrevistas a los trabajadores como también la observación directa.

En las entrevistas estructuradas se realizaron 3 sesiones con la representante legal, Julissa Francisca Quiroz Huamán, distribuidas de la siguiente manera: en la primera, se identificaron problemas en el proceso actual; en la segunda, se realizó una definición y especificación de las funcionalidades requeridas y en la tercera, se validó y se priorizó los requisitos identificados. A su vez, en la observación directa se observó durante 3 días diferentes el proceso actual, en el cual se registró el flujo actual de trabajo de los trabajadores, se registró el tiempo aproximado de cada actividad y se identificó las herramientas utilizadas y los problemas recurrentes.

Objetivo de la aplicación web y móvil

Desarrollar una aplicación web y móvil para el proceso de ventas de la empresa Nikos Pizza que contribuya a registrar los pedidos en menos tiempo, que automatice procesos manuales y ayude en la generación de reportes con información de calidad dando así un valor agregado.

Alcance de la aplicación web y móvil

La empresa Nikos Pizza tiene la necesidad de automatizar el proceso de ventas; para ello se desarrolla una aplicación web y móvil alojado en la web para que se tenga acceso a la información en tiempo real y desde cualquier lugar. A continuación, se muestra las opciones a integrar.

Especificación de requerimientos

A través de reuniones con los representantes de la empresa se determinó los requerimientos de la aplicación web y móvil.

Requerimientos Funcionales (RF):

RF01: La aplicación debe permitir visualizar gráficos sobre las ventas mensuales, y un gráfico sobre las ventas diarias y que sea de rápida visualización.

RF02: La aplicación debe permitir analizar la información del panel principal utilizando herramientas de inteligencia artificial cuando el usuario lo requiera.

RF03: La aplicación debe permitir mostrar la lista de productos más vendidos y visualizar el detalle de cada producto seleccionado.

RF04: La aplicación debe permitir gestionar categorías de productos indicando un nombre obligatorio y una descripción opcional.

RF05: La aplicación debe permitir gestionar clientes indicando, como mínimo, su nombre, número de DNI y fecha de nacimiento, para identificar a los clientes que cumplen años y ofrecerles promociones especiales.

RF06: La aplicación debe permitir gestionar proveedores de productos complementarios indicando obligatoriamente su nombre y dirección.

RF07: La aplicación debe permitir gestionar productos y poder cambiar el estado del producto entre activo o desactivado de forma rápida.

RF08: La aplicación debe permitir gestionar ventas con la opción de asociar o no un cliente, aplicar descuentos, identificar si corresponde a una mesa o delivery, y generar un comprobante en formato PDF.

RF09: La aplicación debe permitir gestionar premios indicando su nombre, descripción y cantidad de compras necesarias, generando alertas cuando un cliente alcance o esté próximo a alcanzar un premio.

RF10: La aplicación debe permitir generar reportes de ventas diarias mostrando el total de ingresos, número de ventas, ventas realizadas en mesa y/o delivery, ventas pagadas en yape y/o efectivo y pedidos cancelados, con opciones de filtrado por producto, vendedor y cliente.

RF11: La aplicación debe permitir generar reportes de ventas por rango de fechas con opciones de filtrado por producto, vendedor o cliente.

RF12: La aplicación debe permitir gestionar reclamos clasificándolos por estado (pendiente, en atención o atendido) y prioridad (baja, media o alta).

RF13: La aplicación debe permitir registrar y actualizar los datos generales de la empresa.

RF14: La aplicación debe permitir modificar el umbral mínimo de ventas diarias mostrado en el panel principal.

RF15: La aplicación debe permitir controlar el acceso a las diferentes secciones de la aplicación.

RF16: La aplicación debe permitir visualizar y modificar la información personal de la cuenta de usuario.

RF17: La aplicación debe permitir utilizar las mismas funcionalidades de la versión web en la aplicación móvil y acceder a esta desde la Google Play Store.

Requerimientos No Funcionales (RNF):

RNF01: La aplicación web debe garantizar una estructura modular que facilite la separación de responsabilidades y permita mantener, actualizar y escalar el software de manera eficiente a lo largo del tiempo.

RNF02: La aplicación móvil debe asegurar una adecuada organización del código que promueva la mantenibilidad, la reutilización de componentes y la claridad en la interacción entre los distintos módulos de la aplicación.

RNF03: La aplicación debe garantizar una alta disponibilidad, permitiendo el acceso a sus funcionalidades desde diversos dispositivos y ubicaciones, en cualquier momento del día.

RNF04: La aplicación debe posibilitar un proceso de despliegue ágil y centralizado que minimice el tiempo y la complejidad al momento de realizar actualizaciones o publicar nuevas versiones, tanto en la aplicación web como en la aplicación móvil.

RNF05: La aplicación debe asegurar la protección de los datos y las comunicaciones mediante el uso de mecanismos que salvaguarden la confidencialidad, integridad y autenticidad de la información intercambiada.

RNF06: La aplicación debe almacenar la información en una base de datos confiable y segura, garantizando la integridad, disponibilidad y rendimiento adecuado de los datos necesarios para las operaciones la aplicación.

3.1.2 ANÁLISIS

Se realizó el análisis y evaluación del proceso actual en la pizzería Nikos Pizza y en donde se identificaron las etapas principales del flujo de trabajo. Asimismo, como parte del estudio, se diagramó el proceso utilizando la metodología BPM (Business Process Management) con la notación BPMN (Business Process Model and Notation), empleando la plataforma Bizagi Modeler.

En la figura 5, se observa el flujo actual del proceso de registro de pedidos (registro en mesa).

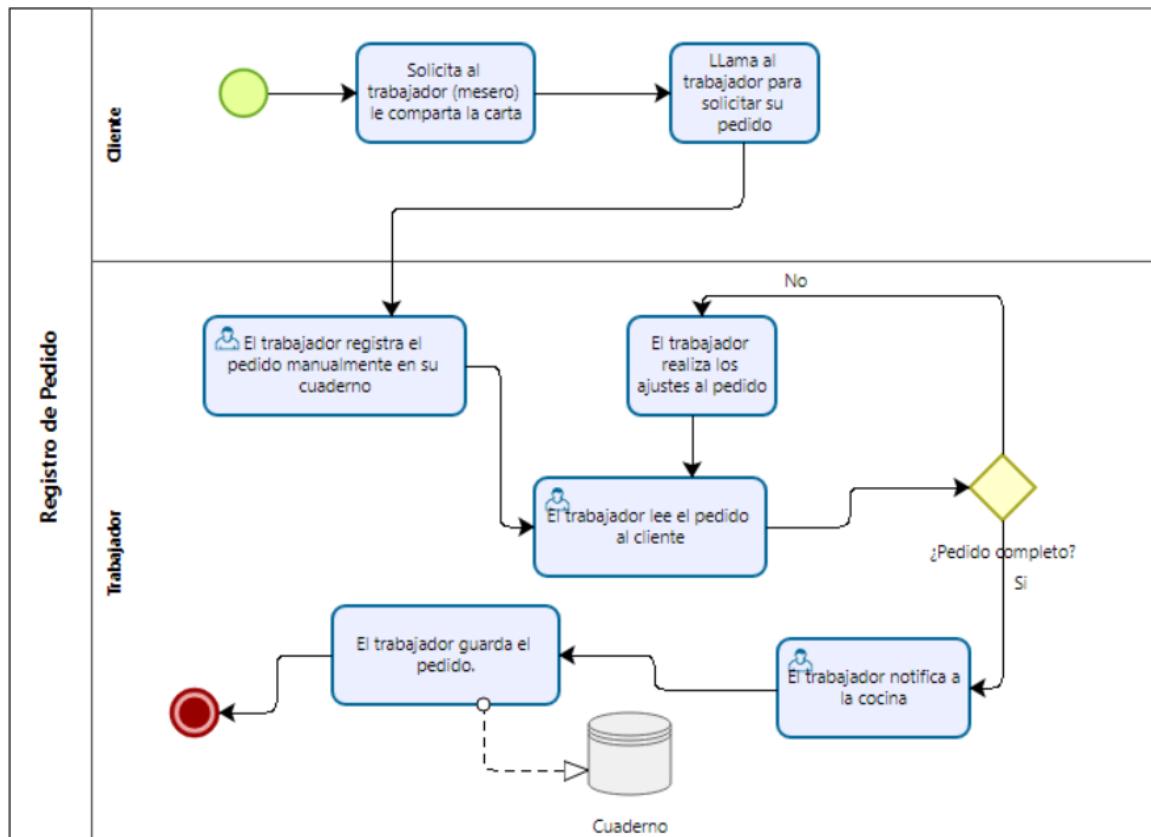


Fig. 5: Modelo del proceso de registro de un pedido

En la figura 6, se muestra el flujo del proceso de registro de un pedido (delivery).

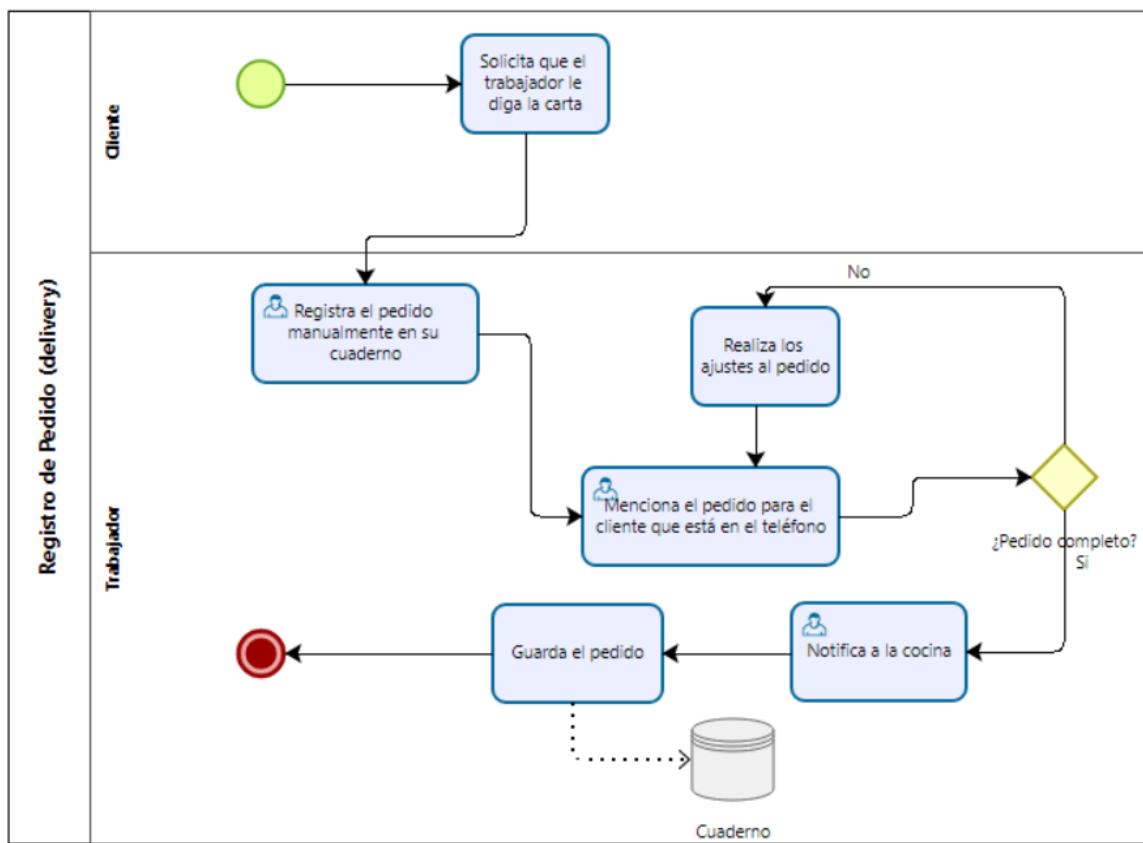


Fig. 6: Modelo del proceso de registro de un pedido (delivery)

En la figura 7, se muestra el flujo del proceso de generación de reportes de ventas.

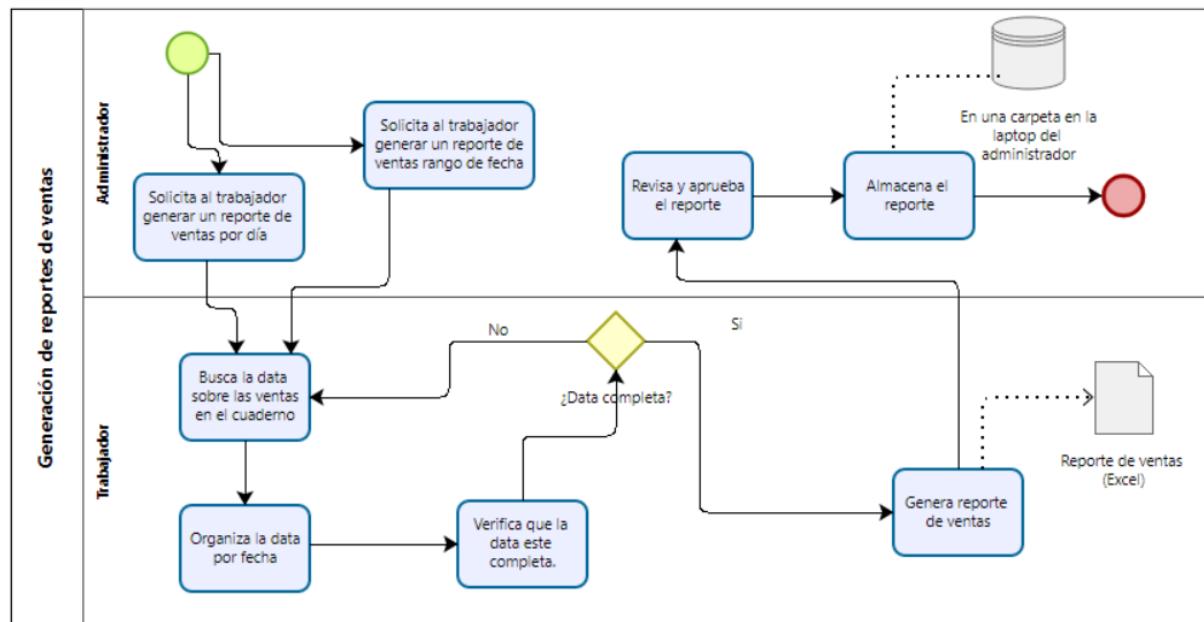


Fig. 7: Modelo del proceso de generación de reportes

En la figura 8, se muestra el proceso de registro de pagos de los pedidos en el proceso de ventas.

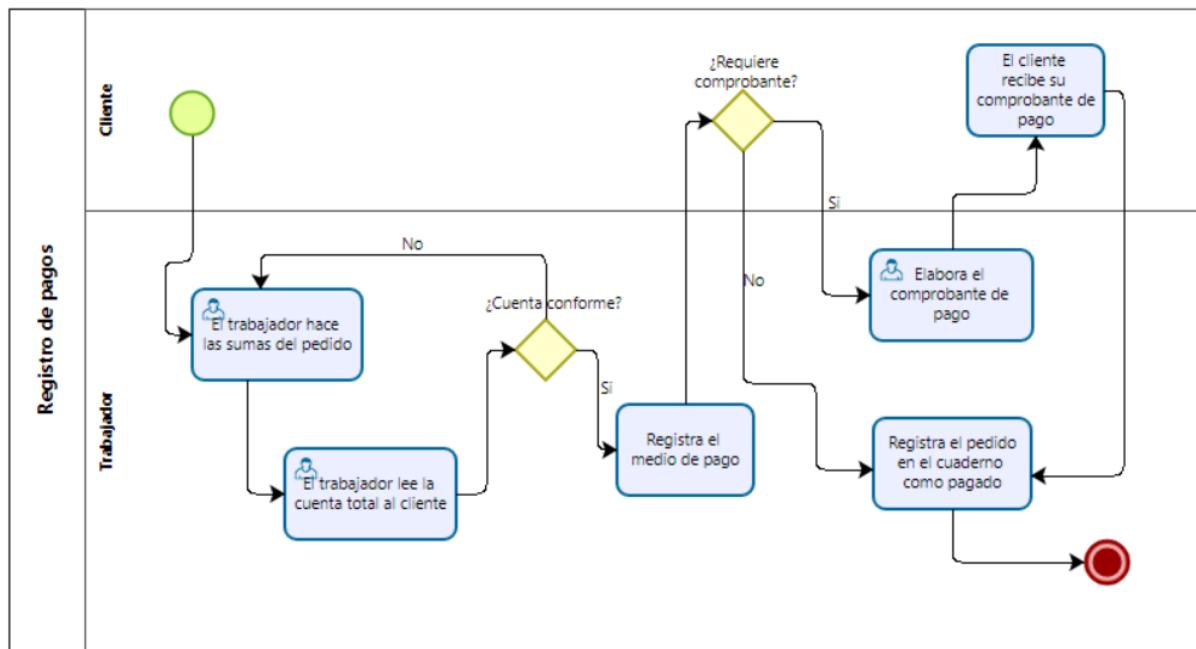


Fig. 8: Modelo del proceso de registro de pagos de los pedidos

Modelado de casos de uso

Para un entendimiento completo de los requisitos, se elaboran casos de uso del negocio que describen los procesos desde la perspectiva de los actores del negocio y casos de uso del sistema que detallan la interacción con la aplicación web y móvil.

Casos de uso del negocio

En la figura 9, se muestra los actores del negocio.

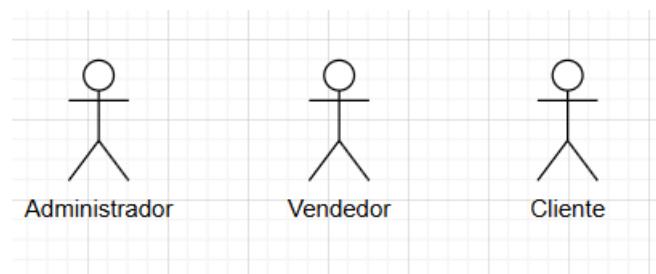


Fig. 9: Actores del negocio

En la figura 10, se muestra los casos de uso del negocio



Fig. 10: Casos de uso del negocio

En la figura 11, se muestra el diagrama de casos de uso del negocio

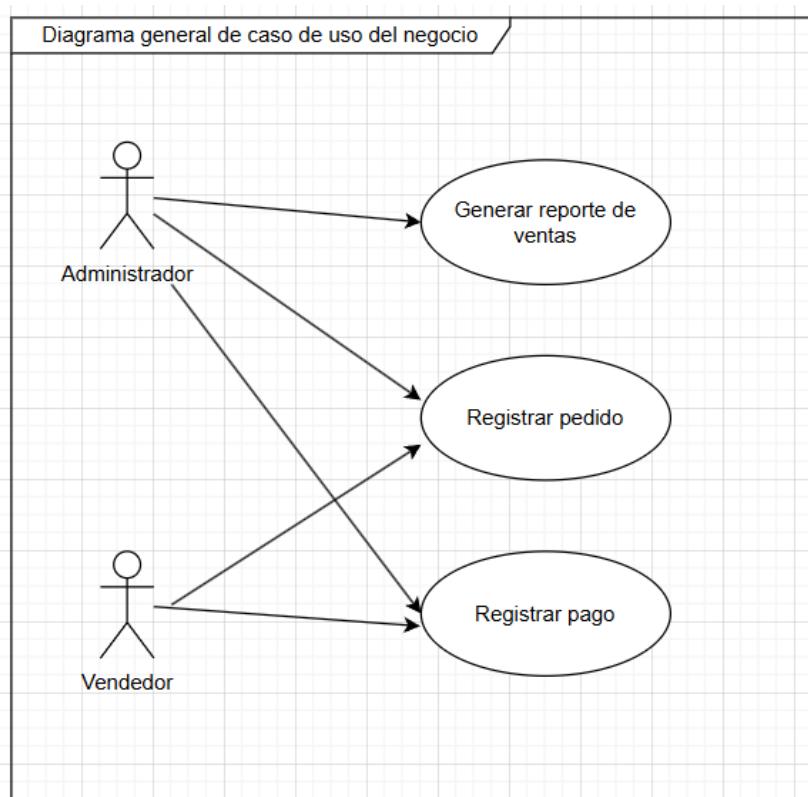


Fig. 11: Diagrama de casos de uso del negocio

Casos de uso de la aplicación web y móvil

En la figura 12, se muestra los actores de la aplicación web y móvil.

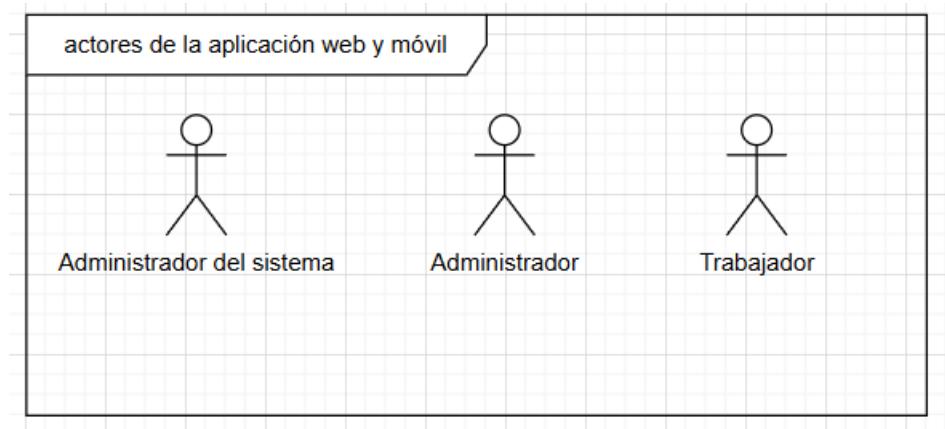


Fig. 12: Actores de la aplicación web y móvil

En la figura 13, se muestra los casos de uso de la aplicación web y móvil

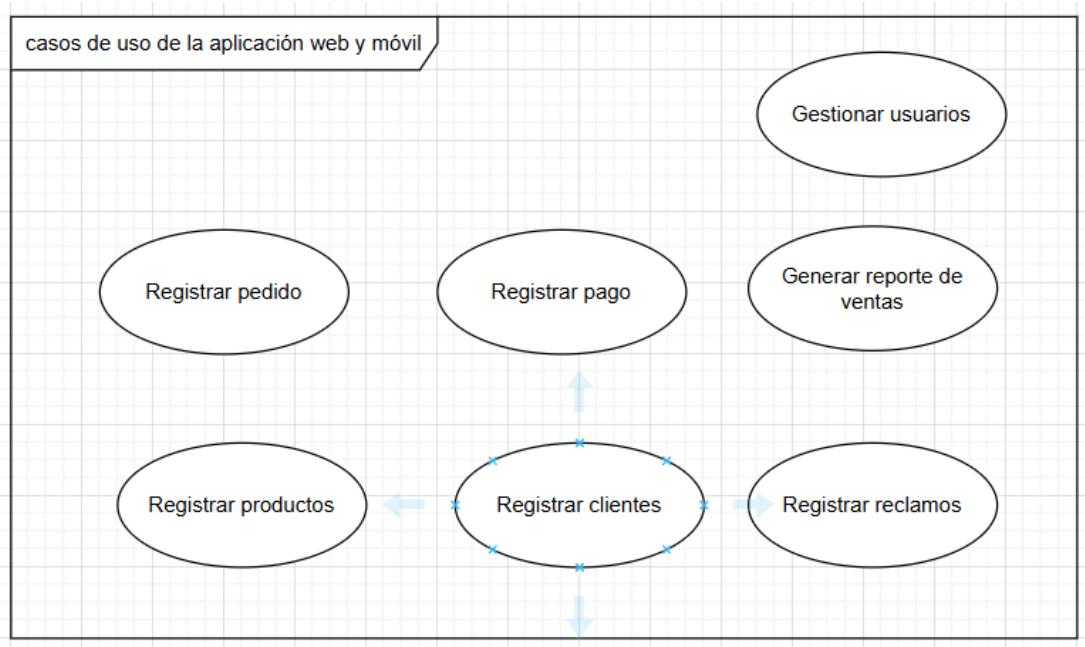


Fig. 13: Casos de uso de la aplicación web y móvil

En la figura 14, se muestra el diagrama de casos de uso de la aplicación web y móvil.

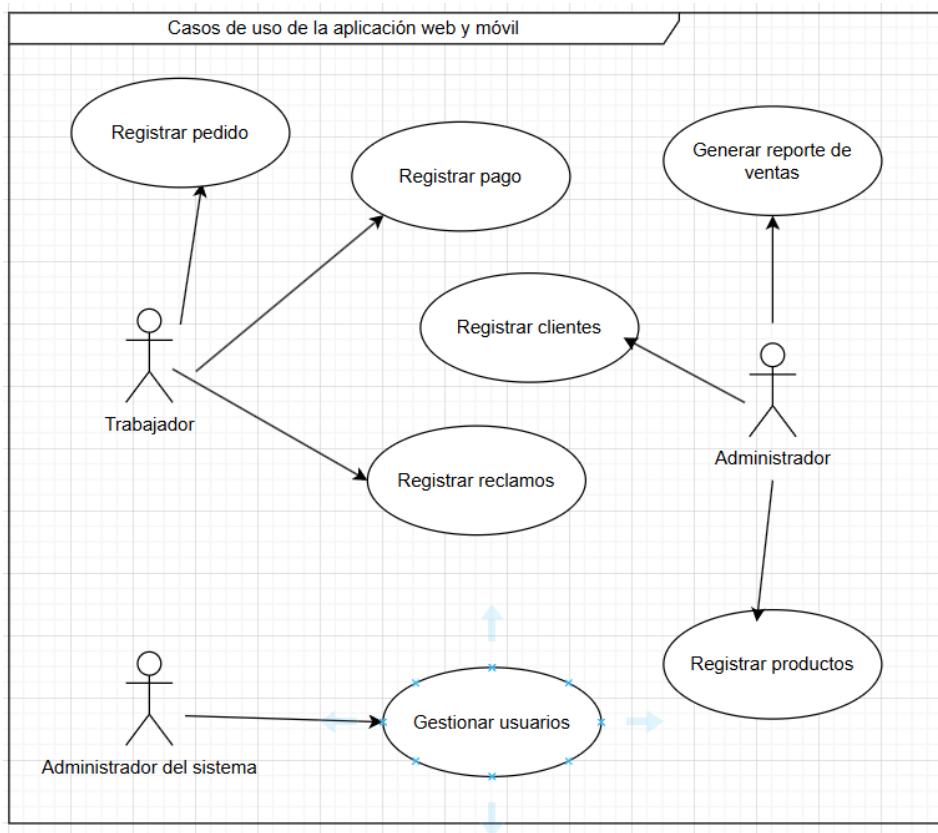


Fig. 14: Diagrama de casos de uso de la aplicación web y móvil.

Especificación de los casos de uso

CU 01: Registrar pedido

En la figura 15, se muestra el diagrama del caso de uso registrar pedido

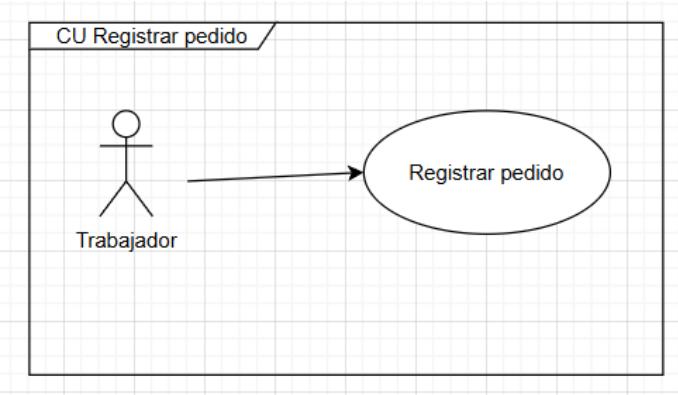


Fig. 15: Diagrama de caso de uso registrar pedido

Especificación de caso de uso

En la tabla I se muestra la especificación del caso de uso registrar pedido.

Tabla I: Especificación de caso de uso registrar pedido

Nombre	Registrar pedido
Autor	Tafur Valdivia, Roberto Jhuliño
Fecha	17/04/2025
Descripción	Este caso de uso permite registrar, editar y cancelar pedidos dentro de la aplicación web y móvil.
Actores	Actor principal: El trabajador encargado de registrar los pedidos (mesero). Actor secundario: El administrador
Precondiciones	El actor debe haber iniciado sesión en la aplicación web y móvil Productos deben estar registrados en la aplicación web y móvil
Flujo normal	<p>El actor accede al panel de control y selecciona la opción 'Pedidos'. El sistema muestra una interfaz que incluye:</p> <ul style="list-style-type: none"> - Botón 'Aregar nuevo pedido' - Campo de búsqueda para pedidos existentes - Lista paginada de pedidos ordenadas descendente por fecha <p>Según la acción seleccionada:</p> <ul style="list-style-type: none"> - Si elige agregar nuevo pedido: proceder al Subflujo 1 - Si elige editar pedido: proceder al Subflujo 2 - Si elige cambiar estado de pedido: proceder al Subflujo 3
Subflujos	<p>Subflujo1:</p> <ul style="list-style-type: none"> - El actor selecciona "agregar nuevo pedido" y se muestra el formulario correspondiente. - El actor completa los campos: cliente, producto, cantidad, precio de venta, tipo (mesa/delivery). - Para pedidos delivery, se especifica el costo adicional - El método de pago por defecto se establece como "Pendiente" - La fecha se autocompleta con la fecha actual - Al presionar "Guardar", el sistema redirige a la vista anterior mostrando mensaje de confirmación <p>Subflujo2:</p>

	<ul style="list-style-type: none"> - El actor busca el pedido mediante el buscador y selecciona el icono de editar. - La aplicación muestra el formulario con los datos actuales, permitiendo modificar: cliente, mesa, costo delivery, método de pago y fecha. - Tras realizar los cambios, el actor presiona "Actualizar", redirigiéndose a la lista de pedidos con un mensaje de pedido actualizado. <p>Subflujo3:</p> <ul style="list-style-type: none"> - El actor busca el pedido objetivo mediante el buscador. - El actor presiona el indicador de estado ("activo") y la aplicación alterna entre "activo" y "cancelado" automáticamente.
Flujo alternativo	En el paso 2 del subflujo 1. Si el actor intenta agregar un detalle de pedido sin completar los campos obligatorios (producto, cantidad, mesa), el sistema muestra una alerta solicitando ingresar toda la información requerida.
Post condiciones	En la lista de pedidos, la aplicación muestra para cada registro: código autogenerado, cliente, monto total y opciones de gestión (cambio de estado, tipo de pago y edición).

CU 02: Registrar pago

En la figura 16, se muestra el diagrama del caso de uso registrar pago.

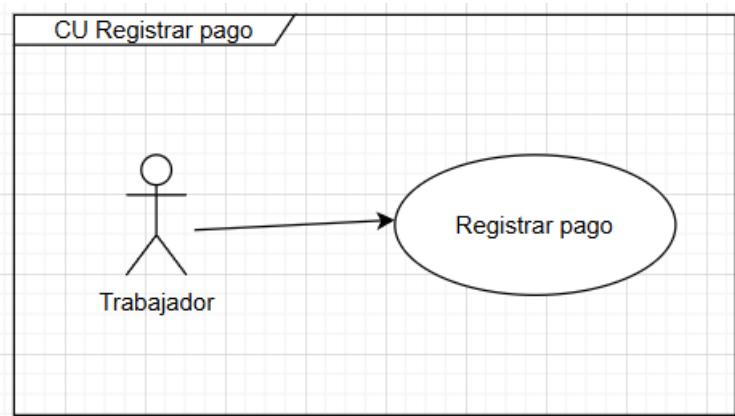


Fig. 16: Diagrama de caso de uso registrar pago

Especificación de caso de uso

En la tabla II se muestra la especificación del caso de uso registrar pago

Tabla II: Especificación de caso de uso registrar pago

Nombre	Registrar pago
Autor	Tafur Valdivia, Roberto Jhuliño
Fecha	20/04/2025
Descripción	Este caso de uso permite registrar el tipo de pago de un pedido, marcando así su finalización en la aplicación web y móvil y concretándose así la venta.
Actores	Actor principal: Trabajador (mesero encargado de registrar pagos). Actor secundario: Administrador.
Precondiciones	<ul style="list-style-type: none"> - El actor debe haber iniciado sesión en la aplicación web y móvil. - Debe existir al menos un pedido registrado con estado “pendiente de pago”.
Flujo normal	<p>El actor se ubica en el panel de control y selecciona pedidos.</p> <p>La aplicación muestra: buscador, lista paginada de pedidos y botón “cobrar por mesa”.</p> <p>El actor selecciona una de las opciones:</p> <ul style="list-style-type: none"> - Buscar pedido específico: ver subflujo1 - Cobrar por mesa: ver subflujo2
Subflujos	<p>Subflujo1:</p> <ul style="list-style-type: none"> - El actor localiza el pedido mediante el buscador. - Selecciona la opción “cambiar tipo de pago”. - La aplicación muestra modal con métodos de pago disponibles. - El actor selecciona método y confirma. - La aplicación actualiza estado y cierra modal. <p>Subflujo2:</p> <ul style="list-style-type: none"> - El actor seleccionar cobrar por mesa. - El actor selecciona la mesa y presiona el botón de buscar. - La aplicación muestra pedidos pendientes de pago. - El actor selecciona “pagar” y elige método de pago. - La aplicación registra pago y cierra modal.
Flujo alternativo	En el paso 1 del subflujo 1. El actor al presionar el botón de “cancelar” se cierra el modal y no se realizan cambios.

	En el paso 2 del subflujo 2: El actor al presionar buscar y si no existe pedidos pendientes la lista se muestra vacía.
Post condiciones	<ul style="list-style-type: none"> - El estado de pago se actualiza en tiempo real. - El pedido cambia a estado “completado”.

CU 02: Generar reporte de ventas

En la figura 17, se muestra el diagrama del caso de uso generar reporte de ventas.

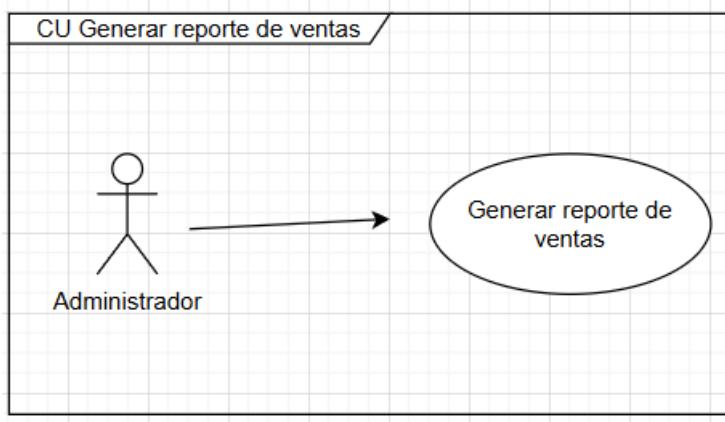


Fig. 17: Diagrama de caso de uso generar reporte de ventas

Especificación de caso de uso

En la tabla III, se muestra la especificación del caso de uso generar reporte de ventas.

Tabla III: Especificación de caso de uso generar reporte de ventas

Nombre	Generar reporte de ventas
Autor	Tafur Valdivia, Roberto Jhuliño
Fecha	25/04/2025
Descripción	Este caso de uso permite generar reportes de ventas con diferentes criterios de filtrado y la opción de consultar información mediante inteligencia artificial. Se presenta dos reportes, por día y por rango de fecha, cada uno muestra las el ingreso total de las ventas, el medio de pago (cantidad y total de ingresos), el lugar (en mesa y/o delivery) y también se muestra los pedidos cancelados.
Actores	<p>Actor principal: El administrador.</p> <p>Actor secundario: Trabajador (según permisos).</p>
Precondiciones	<ul style="list-style-type: none"> - El actor debe haber iniciado sesión en la aplicación web y móvil.

	<ul style="list-style-type: none"> - Debe existir al menos una venta registrada en la aplicación web y móvil.
Flujo normal	<p>El actor se ubica en el panel de control y selecciona reportes.</p> <p>La aplicación muestra dos opciones: “reportes de ventas por día” y “reporte de ventas por fecha”.</p> <p>El actor selecciona una opción:</p> <ul style="list-style-type: none"> - Reporte de ventas por día: ver subflujo1 - Reporte de ventas por fecha: ver subflujo2
Subflujos	<p>Subflujo1:</p> <ul style="list-style-type: none"> - La aplicación muestra automáticamente el reporte del día actual con el total de ventas, ventas clasificadas por su tipo de pago y ventas clasificadas por ventas en mesa y/o delivery. - El actor puede aplicar filtros por producto, vendedor y/o cliente. - Los resultados se actualizan automáticamente al seleccionar filtros. - El actor puede descargar el reporte en formato Excel. - El actor puede realizar consultas a la IA sobre los datos mostrados. <p>Subflujo2:</p> <ul style="list-style-type: none"> - El actor selecciona un rango de fechas (fecha inicio y fecha fin). - El actor presiona el botón “buscar” para ejecutar la consulta. - La aplicación muestra las ventas correspondientes al periodo seleccionado con sus respectivas clasificaciones. - El actor puede aplicar filtros adicionales (producto, vendedor, cliente). - El actor debe presionar “buscar” nuevamente para aplicar los filtros. - El actor puede descargar el reporte en Excel. - El actor puede realizar consultas a la IA sobre los datos.
Flujo alternativo	<p>En el subflujo 1 y 2 cuando se pregunta a la IA:</p> <ul style="list-style-type: none"> - Al seleccionar la opción de consulta a la IA (Gemini), la aplicación muestra un modal. - El actor ingresa su pregunta en texto y presiona “enviar”.

	<ul style="list-style-type: none"> - La aplicación procesa la consulta y muestra la respuesta en el mismo modal. - El actor puede realizar nuevas consultas o cerrar el modal.
Post condiciones	<ul style="list-style-type: none"> - Para utilizar la funcionalidad de IA, debe existir un token válido configurado en las variables de entorno del proyecto. - El administrador del sistema es responsable de gestionar la configuración del token. - En caso de agotar el límite de tokens, las consultas se reanudarán al siguiente día. - Los reportes generados incluyen todos los filtros aplicados durante la consulta.

Para consultar la especificación completa de los demás casos de uso identificados en el sistema, se ha incluido el **Anexo 17** que contiene el detalle de cada uno de ellos.

3.1.3 DISEÑO

Para garantizar el éxito del desarrollo de la aplicación, se llevó a cabo una cuidadosa selección de tecnologías que sobre todo ayuden a construir una aplicación robusta, escalable y de fácil mantenimiento. Asimismo, se realizaron diagramas de componentes y diagramas de clases para tener un entendimiento claro de la aplicación web y móvil. A su vez se elaboró diagramas de secuencia para comprender la secuencia que debe seguir el usuario al utilizar el sistema.

Arquitectura de la aplicación web y móvil

Para la aplicación web se implementa la arquitectura MVC (Modelo-Vista-Controlador) debido a que permite separar responsabilidades, facilitando el mantenimiento y la escalabilidad de la aplicación web. El Modelo gestiona los datos, la Vista se encarga de la interfaz de usuario y el Controlador maneja las peticiones y coordina ambos componentes. Para la aplicación móvil se utiliza una arquitectura híbrida apoyada en el patrón de diseño MVVM (Modelo-Vista-Modelo de Vista), donde el ViewModel actúa como intermediario entre la Vista y el Modelo, gestionando el estado de la interfaz y la lógica de presentación, garantizando consistencia en los datos y lógica de negocio entre ambas plataformas.

Diagrama de componentes

En la figura 18, se muestra el diagrama de componentes de la aplicación web, siguiendo la arquitectura MVC.

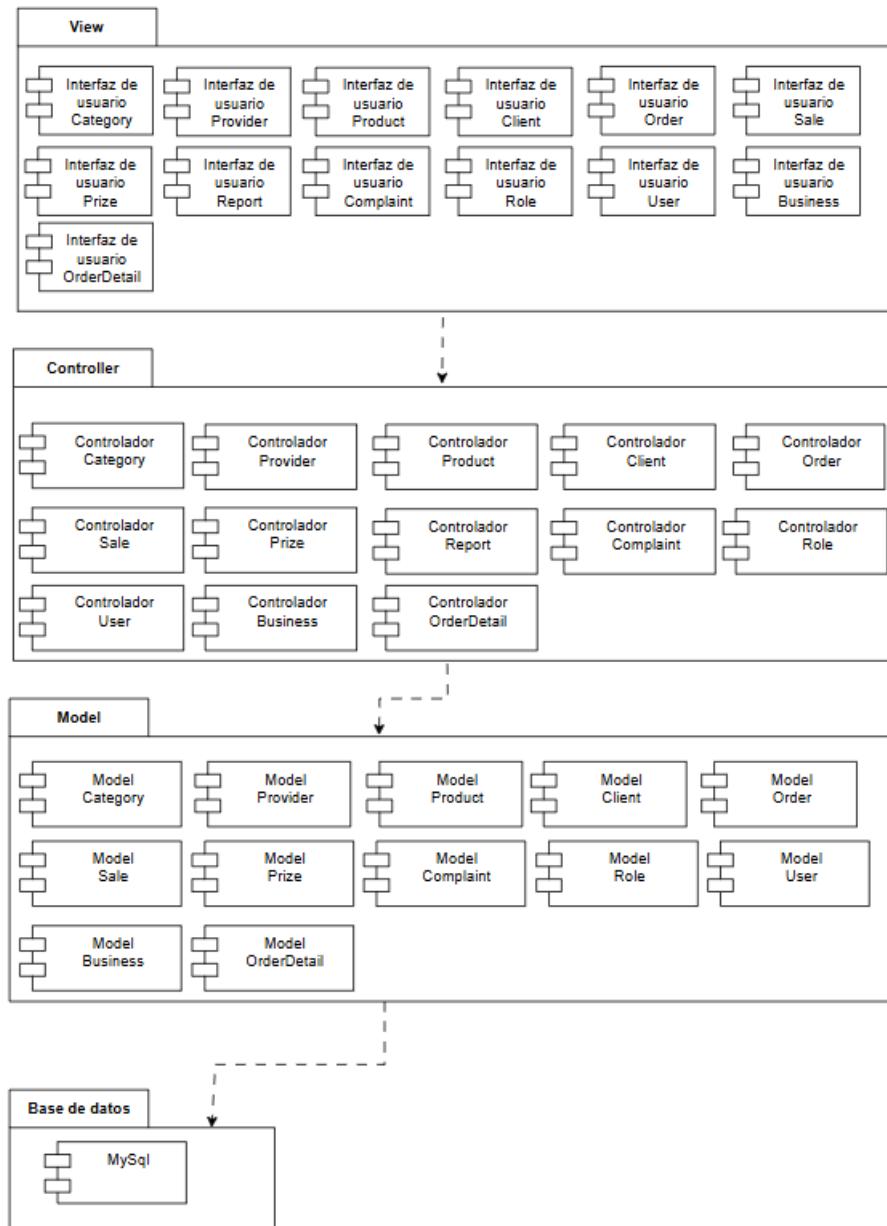


Fig. 18: Diagrama de componentes aplicación web

En la figura 19, se muestra el diagrama de componentes de la aplicación móvil, que a través de un webview se conecta a la aplicación web.

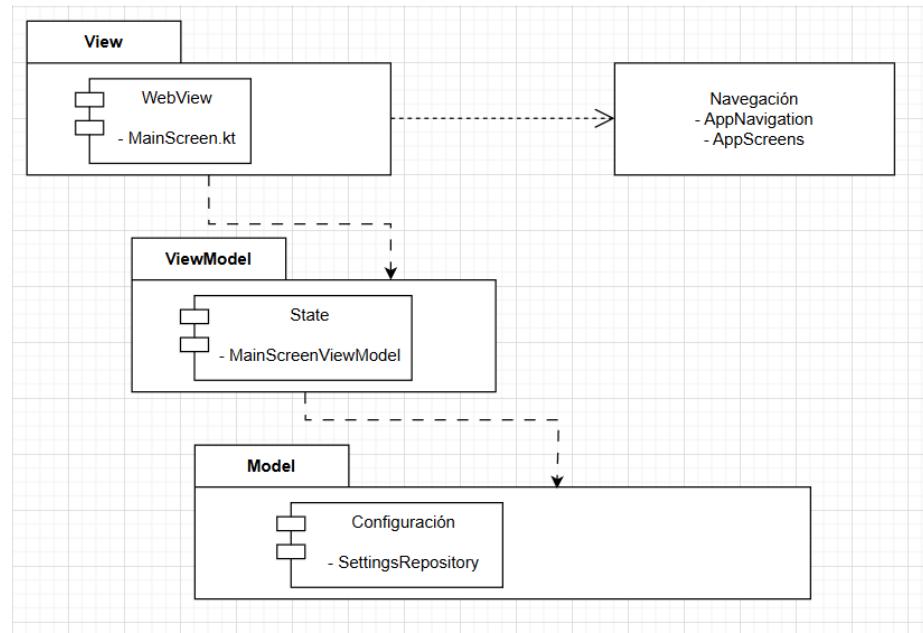


Fig. 19: Diagrama de componentes aplicación móvil.

Elección de tecnologías

Para elegir el framework adecuado para el desarrollo de la aplicación se realizó un análisis comparativo entre los más comunes para el desarrollo de aplicaciones web como se muestra en la tabla IV. Los frameworks fueron seleccionados debido a su amplia adopción en proyectos web, su versatilidad y su capacidad para abordar diferentes necesidades de desarrollo.

Tabla IV: Categorización de frameworks por criterios

Criterio	Laravel	Django	Ruby on Rails	Express.js
Facilidad de uso	Tiene una estructura clara y organizada	Sencillo, pero puede parecer más rígido por sus convenciones estrictas.	Rápido para prototipos	Flexible, pero requiere más trabajo manual.
Curva de aprendizaje	Relativamente baja	Relativamente moderada	Suave, pero se debe conocer Ruby	Fácil si se conoce JavaScript
Costo	Gratis	Gratis	Gratis	Gratis

Criterio	Laravel	Django	Ruby on Rails	Express.js
Integración con base de datos	Se integra muy bien con MySQL	Compatible pero no la más optimizada	Funciona bien, pero requiere configuraciones adicionales	Depende de paquetes adicionales.
ORM	Eloquent	Django ORM	ActiveRecord	No incluye ORM
Comunidad	Muy grande y activa	Grande, con buen soporte	Disminuyendo en popularidad	En crecimiento, pero más pequeña

Para evaluar los frameworks seleccionados, se asignaron puntajes a cada uno de los criterios en una escala de 1 a 5 puntos, donde 1 representa el desempeño más bajo y 5 el más alto. La escala de puntuación del 1 al 5, fue establecida mediante consenso entre el representante legal de Nikos Pizza y el desarrollador. Los criterios acordados mutuamente son: 5 puntos, cumple excelentemente con el criterio y supera expectativas; 4 puntos, cumple muy bien con el criterio; 3 puntos, cumple adecuadamente con el criterio; 2 puntos, cumple parcialmente con el criterio; 1 punto, no cumple o cumple mínimamente con el criterio. Cabe mencionar que esto se alinea con los principios de comprensión de requerimientos donde el cliente participa activamente en la evaluación de alternativas garantizando así que la selección tecnológica responda a las necesidades reales del negocio [66].

En la tabla V, se muestra los puntajes asignados a cada framework asignados mediante consenso entre el representante de la empresa y el desarrollador.

Tabla V: Puntajes de cada framework por criterio

Criterio	Laravel	Django	Ruby on Rails	Express.js
Facilidad de uso	5	4	4	3
Curva de aprendizaje	5	4	3	5
Integración con base de datos	5	4	4	3
Comunidad	5	4	3	4
ORM	5	4	4	2
Costo	5	5	5	5

Para asignar el puntaje final a cada framework se utilizó la fórmula de sumatoria simple del puntaje de cada criterio, como se muestra a continuación.

$$\text{Puntaje Final} = \Sigma(\text{Puntaje del criterio})$$

Es así como se obtuvo los siguientes valores para cada tecnología. Laravel obtuvo un puntaje final de 30, Django un puntaje de 25, Ruby on Rails un puntaje de 23 y Express.js un puntaje de 22. El framework a utilizar fue Laravel debido a que obtuvo el puntaje más alto debido a su consistencia en los criterios evaluados, especialmente en la facilidad de uso, integración de base de datos y costo.

Para elegir el mejor hosting compartido para desplegar la aplicación web se realizó un análisis comparativo que se detalla en la tabla VI.

Tabla VI: Categorización de hosting por criterio

Criterio	InfinityFree	Hostinger	Bluehost	IONOS
Costo	Gratis	Desde \$1.99/mes	Desde \$2.95/mes	Desde \$1/mes
Almacenamiento	Limitado (5GB)	Entre 30GB y 200GB	Almacenamiento ilimitado	100GB en el plan básico
SSL	Certificados SSL gratuitos	Certificados SSL gratuitos	Certificados SSL gratuitos	Certificados SSL gratuitos
Soporte	Solo vía foros y comunidad	Soporte 24/7 en varios idiomas	Soporte 24/4 con buena reputación	Soporte disponible, respuesta variable
Soporte para Laravel	Fácil integración de Laravel	Soporte completo para Laravel	Soporte completo para Laravel	Soporte para Laravel, pero puede requerir configuraciones adicionales.
Despliegue con GitHub Actions	Despliegue fácil a través de FTP	Soporte parcial en algunos planes avanzados	Compatible con GitHub Actions	Soporte limitado para GitHub Actions en algunos planes
Dominio	Subdominio gratuito	Dominio gratuito en algunos planes	Dominio gratuito el primer año	Dominio gratuito en algunos planes

En la tabla VII se muestran los puntajes asignados a cada proveedor de hosting compartido, evaluados según los criterios establecidos. Estos puntajes se califican en una escala del 1 al 5, de un consenso entre el representante legal y el desarrollador detallado anteriormente.

Tabla VII: Puntajes para cada hosting por criterio

Criterio	InfinityFree	Hostinger	BlueHost	IONOS
Costo	5	3	3	4
Almacenamiento	5	2	3	3
SSL	5	5	5	5
Soporte	2	4	3	2
Soporte para Laravel	4	4	3	3
Despliegue con GitHub Actions	4	4	3	3
Dominio	4	2	2	2

Luego de realizar el análisis ponderado de acuerdo a los puntajes asignados a cada hosting, InfinityFree obtuvo un puntaje final de 29, Hostinger un puntaje de 24, BlueHost de 22 y el hosting de IONOS un puntaje de 22. InfinityFree obtuvo el puntaje más alto y se tomó como la mejor opción, destacando principalmente por su costo gratuito, almacenamiento ilimitado y la disponibilidad de dominio gratuito.

Para el servidor VPS para el uso de la base de datos se analizó los más usados por la comunidad de desarrollos, en la tabla VIII se detallan las características principales de ambos proveedores.

Tabla VIII: Categorización del servidor (VPS) por criterio

Criterio	CloudCone	DigitalOcean
Costo	Desde \$3.50/mes	Desde \$5/mes
Almacenamiento	20GB SSD	25GB SSD
Rendimiento	Muy bueno, ajustable según el plan	Excelente
Facilidad de uso	Interfaz intuitiva y panel de control	Interfaz sencilla
Soporte	Soporte 24/7.	Buen soporte con documentación extensa
Escalabilidad	Escalabilidad fácil según las necesidades	Fácil de escalar con solo unos clics
Compatibilidad con MySQL	Soporte completo y fácil instalación de MySQL	Compatibilidad total, instalación rápida de MySQL

Posteriormente, en la tabla IX se asignan puntajes a cada criterio evaluado, asignando un puntaje del 1 al 5 de acuerdo a lo detallado anteriormente sobre el consenso entre el representante legal y el desarrollador, permitiendo obtener un análisis cuantitativo que respalda la decisión final.

Tabla IX: Pesos de criterio los servidores

Criterio	CloudCone	DigitalOcean
Costo	5	4
Almacenamiento	4	5
Rendimiento	4	5
Facilidad de uso	5	4
Soporte	5	4
Escalabilidad	5	4
Compatibilidad con MySQL	5	4

Al evaluar ambos servidores, CloudCone obtuvo un puntaje total de 33, superando a DigitalOcean con un puntaje de 30. Esto se debe principalmente a su costo más accesible, lo que lo convierte en la mejor opción para alojar la aplicación web.

Para la aplicación móvil se analizó las recomendaciones que brindan expertos y organizaciones relacionadas al desarrollo móvil. El lenguaje se optó por Kotlin debido a que este lo sugiere

Google, además se sabe que más del 67% de los desarrolladores profesionales que usan Kotlin afirman que aumenta su productividad debido a que es un lenguaje conciso, interoperable, fácil de aprender y menos propenso a errores [67]. Por otra parte, Google recomienda usar Jetpack Compose debido a que es un kit de herramientas moderno de Android para compilar IU nativas [68].

Diagrama de clases

En la figura 20, se muestra el diagrama de clases de la aplicación web y móvil.

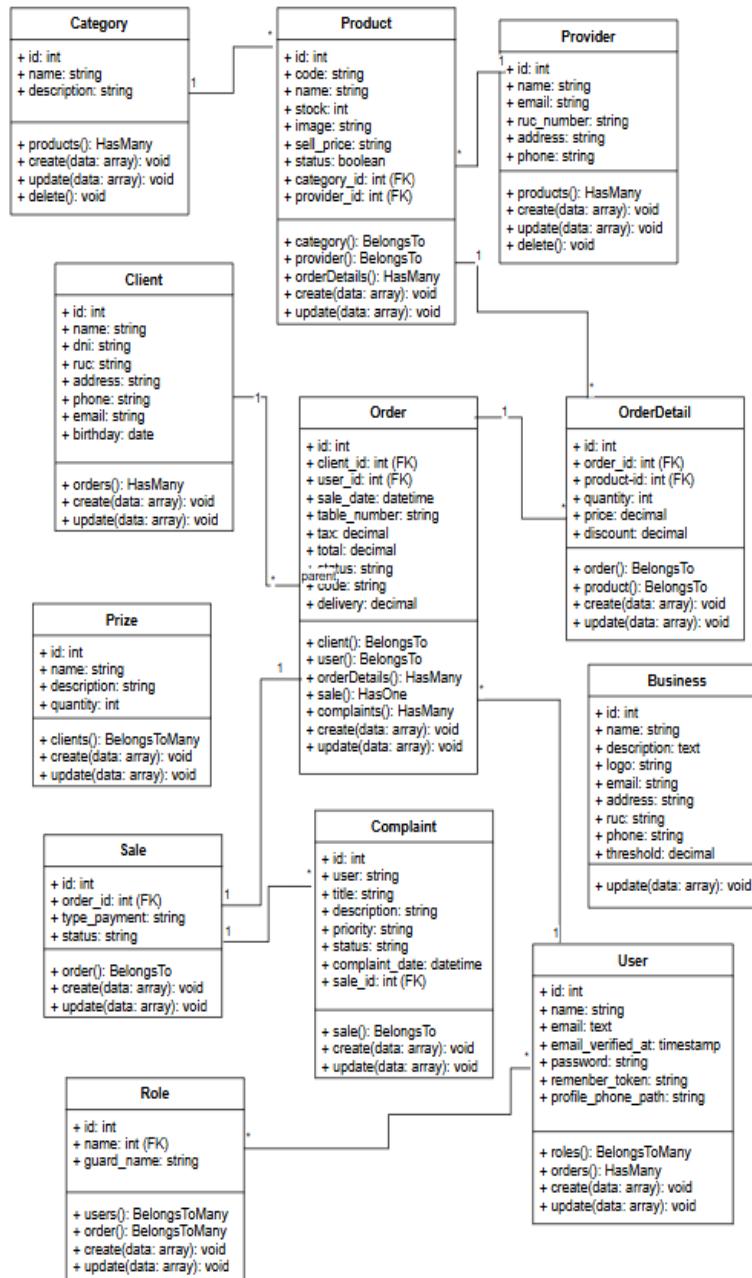


Fig. 20: Diagrama de clases de la aplicación web y móvil

Diagramas de secuencia

Luego de hacer el análisis y tener claro con los casos de uso se procedió a elaborar los diagramas de secuencia como se muestra a continuación.

En la figura 21, se muestra el diagrama de secuencia para el dashboard. El dashboard carga luego del correcto inicio de sesión.

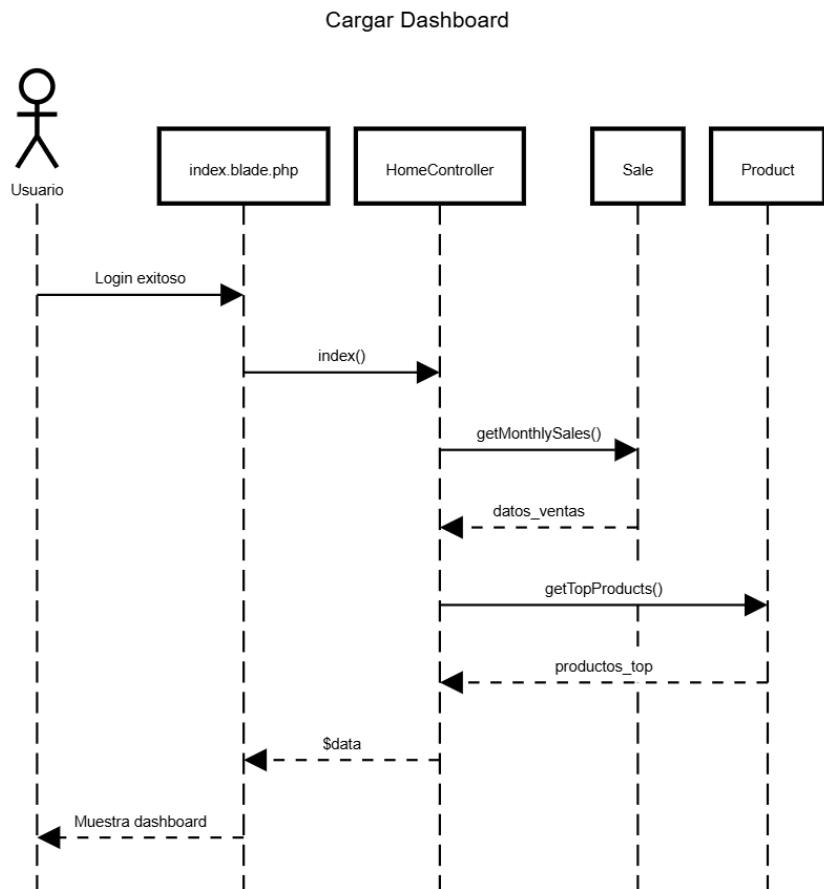


Fig. 21: Diagrama de secuencia para el dashboard

Para complementar el diseño se elaboró un diagrama de secuencia para mostrar el flujo de secuencia cuando el usuario presiona categorías en el panel administrativo. En la figura 22, se muestra el diagrama de secuencia para listar categorías.

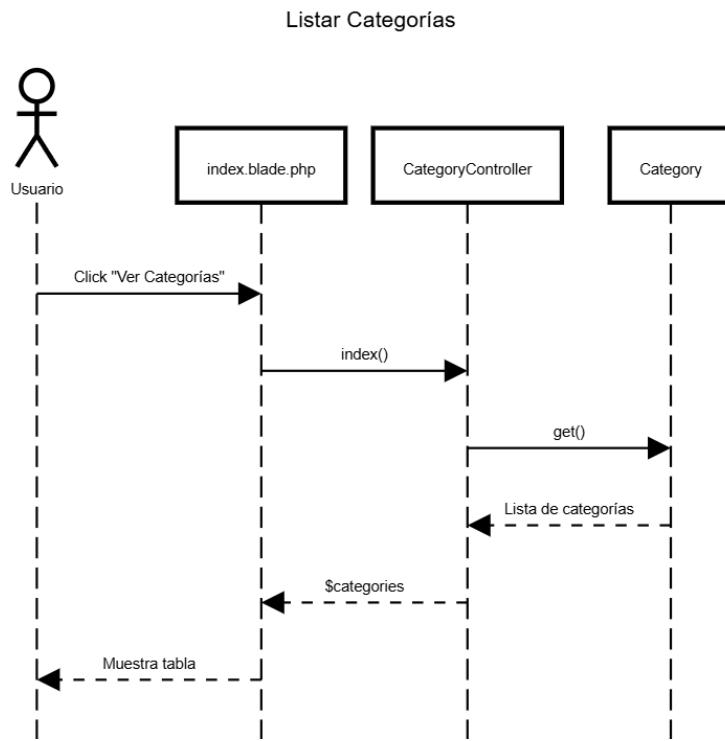


Fig. 22: Diagrama de secuencia para listar categorías

En la figura 23, se muestra el diagrama de secuencia para crear una categoría.

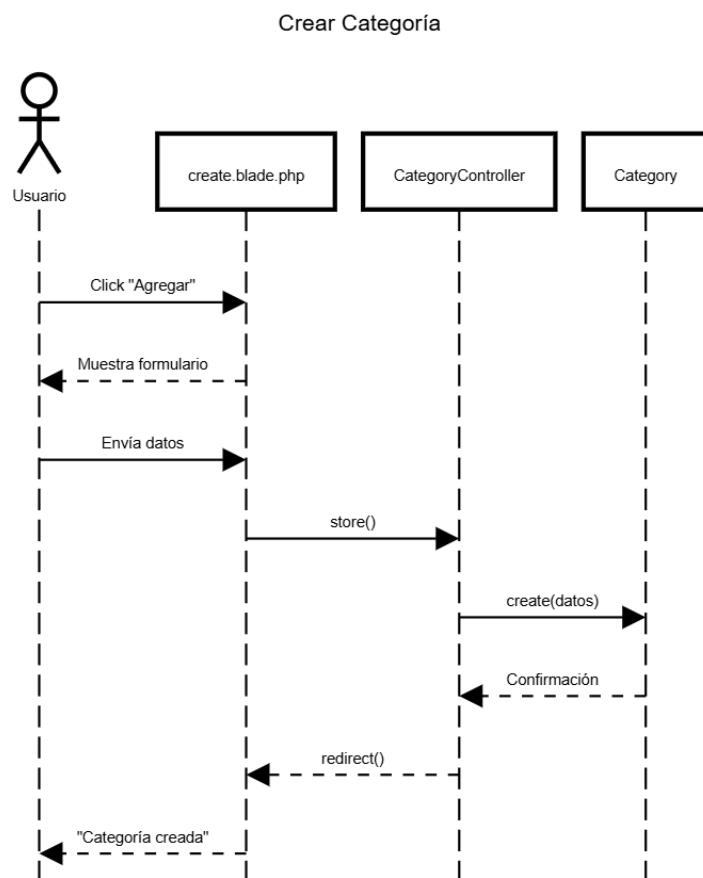


Fig. 23: Diagrama de secuencia para crear categoría

En la figura 24, se muestra el diagrama de secuencia para mostrar la lista de clientes luego de que el usuario presione clientes en el panel de control.

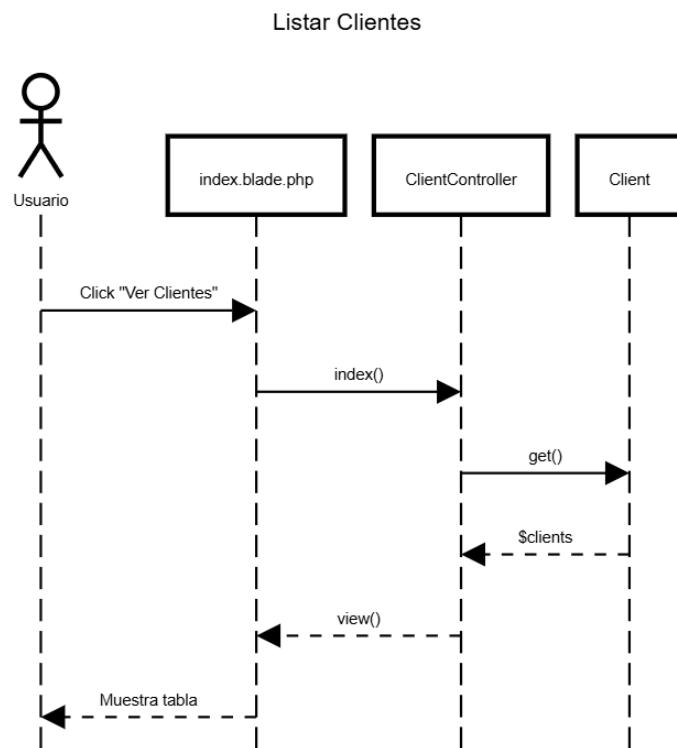


Fig. 24: Diagrama de secuencia para listar clientes

En la figura 25, se muestra el diagrama de secuencia para crear cliente.

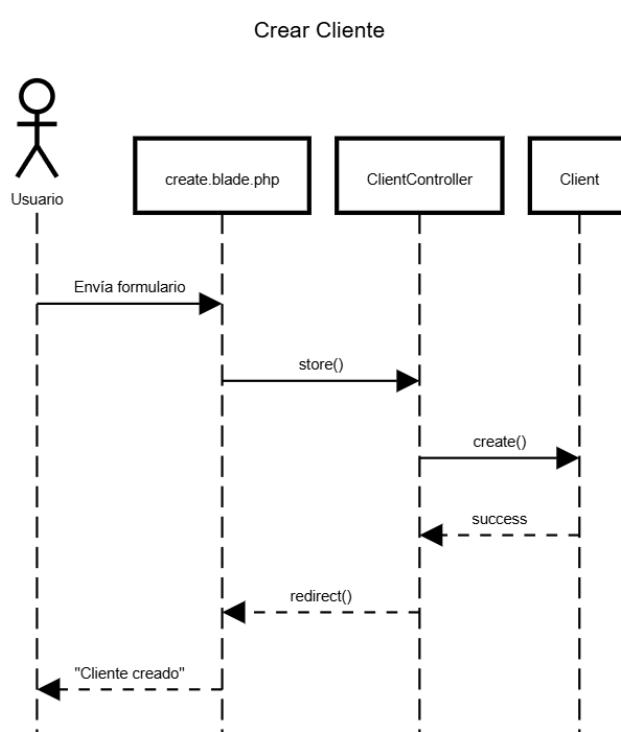


Fig. 25: Diagrama de secuencia para crear cliente

En la figura 26, se muestra el diagrama de secuencia para mostrar los proveedores luego de que el usuario presione en proveedores en el panel de control.

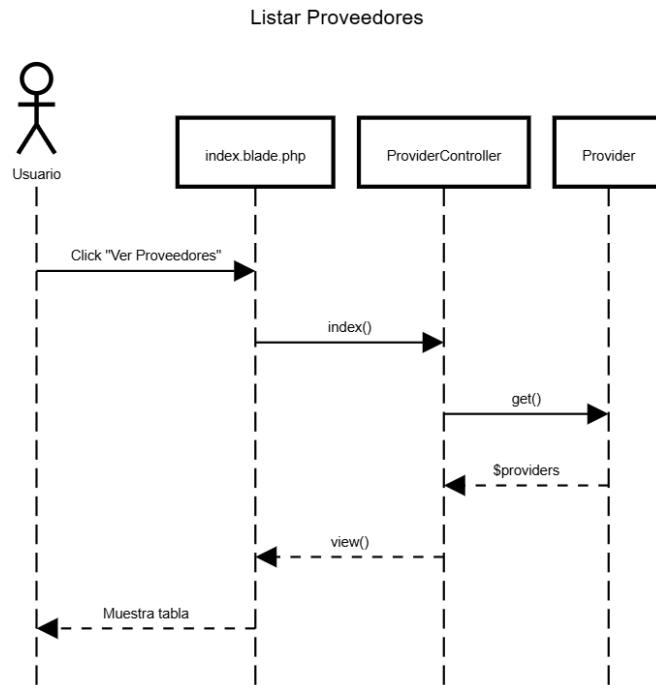


Fig. 26: Diagrama de secuencia para listar proveedores

En la figura 27, se muestra el diagrama de secuencia para crear proveedor.

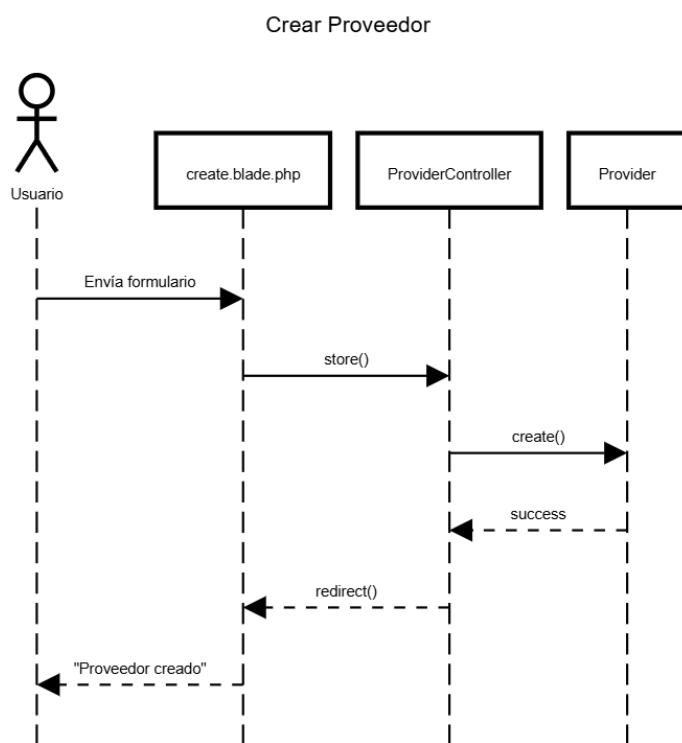


Fig. 27: Diagrama de secuencia para crear proveedor

En la figura 28, se muestra el diagrama de secuencia para listar productos.

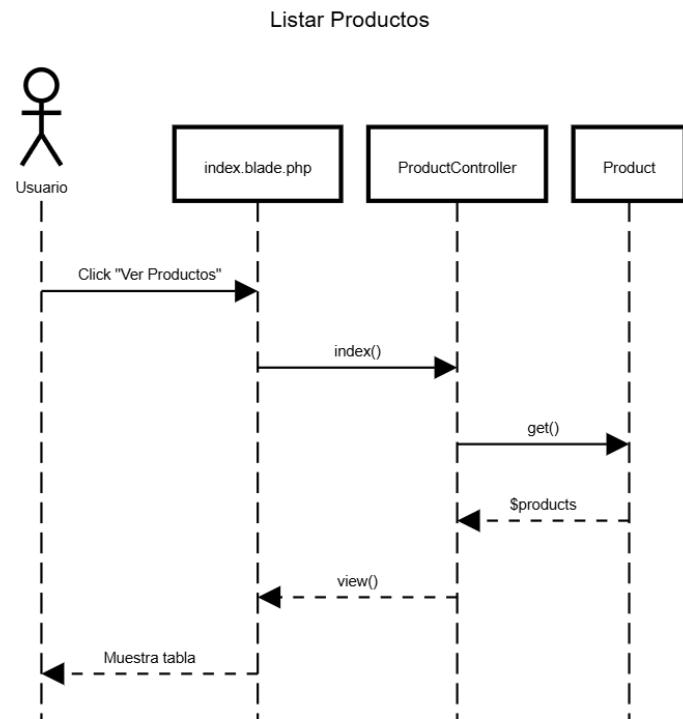


Fig. 28: Diagrama de secuencia para listar productos

En la figura 29, se muestra el diagrama de secuencia para crear un producto

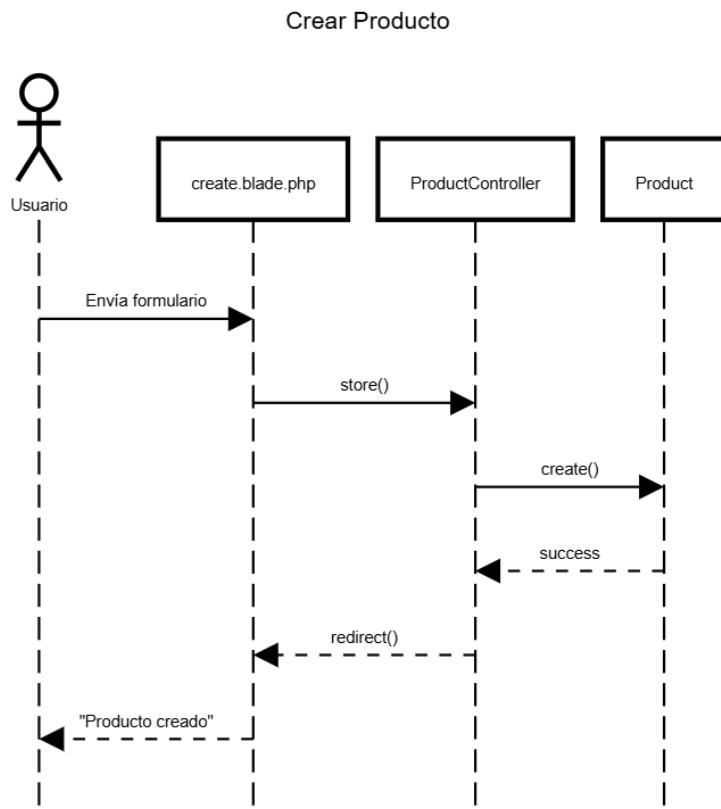


Fig. 29: Diagrama de secuencia para crear producto

En la figura 30, se muestra el diagrama de secuencia para crear un pedido.

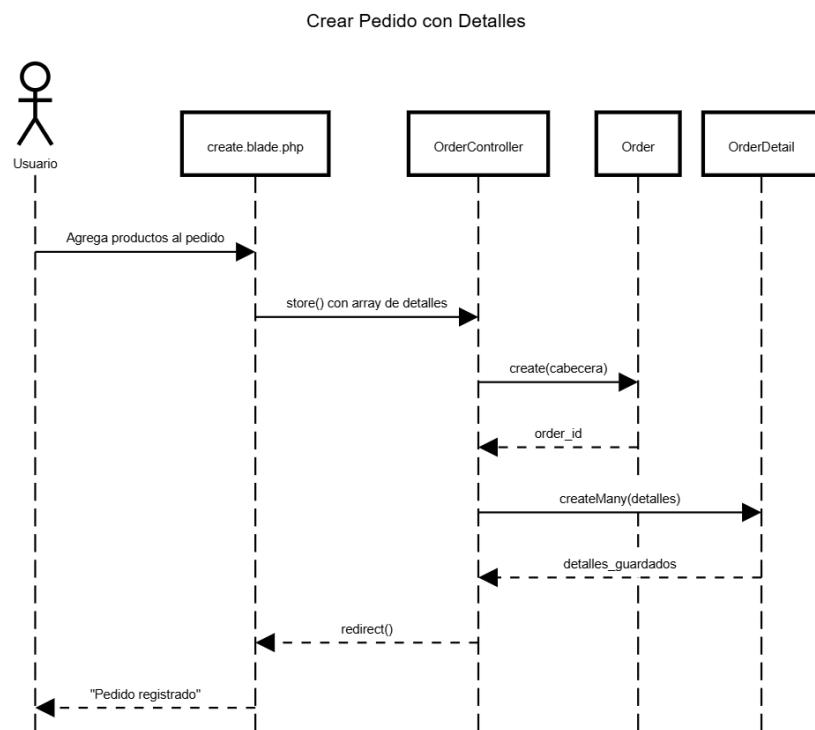


Fig. 30: Diagrama de secuencia para crear un pedido

En la figura 31, se muestra el diagrama de secuencia para listar pedidos.

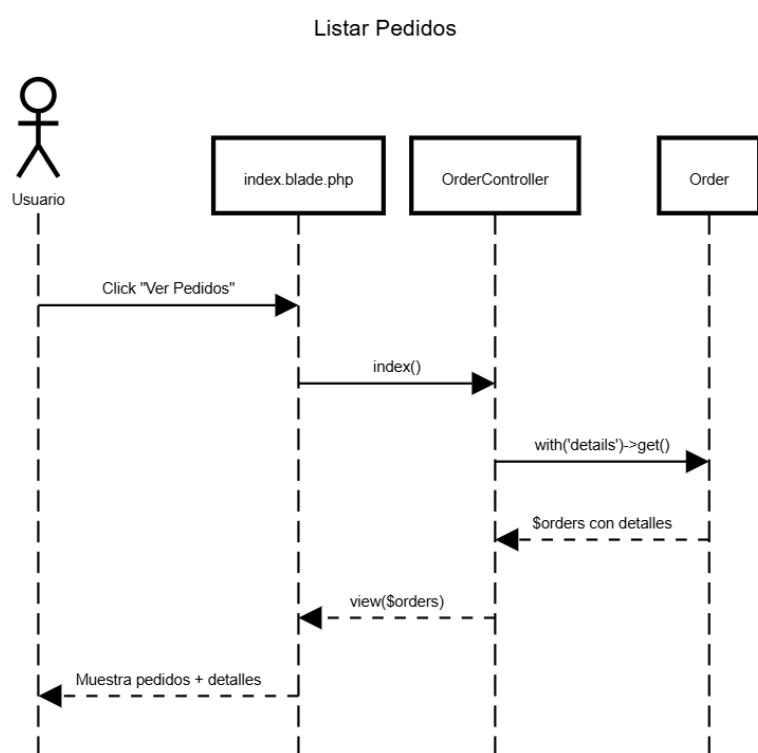


Fig. 31: Diagrama de secuencia para listar pedidos

En la figura 32, se muestra el diagrama de secuencia para pagar un pedido.

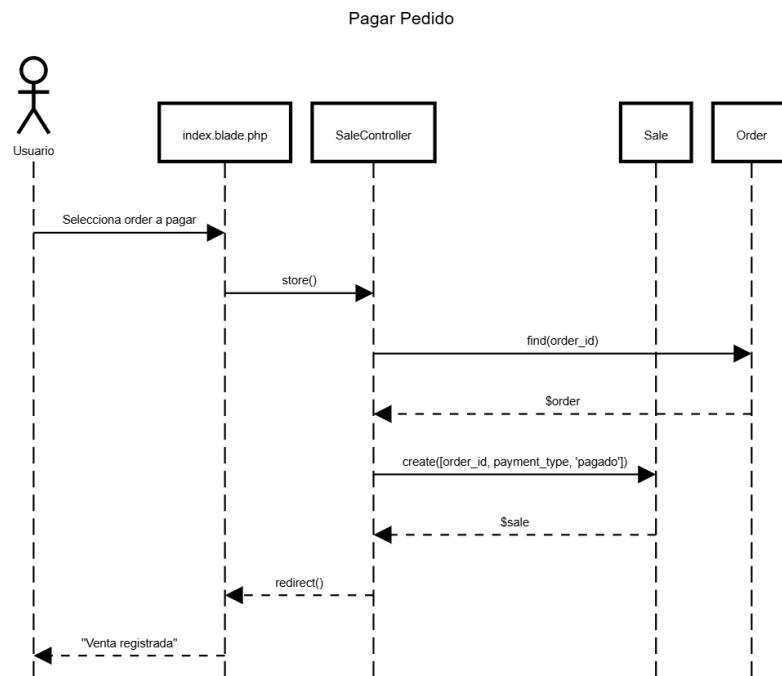


Fig. 32: Diagrama de secuencia para pagar pedido

En la figura 33, se muestra el diagrama de secuencia para listar los premios.

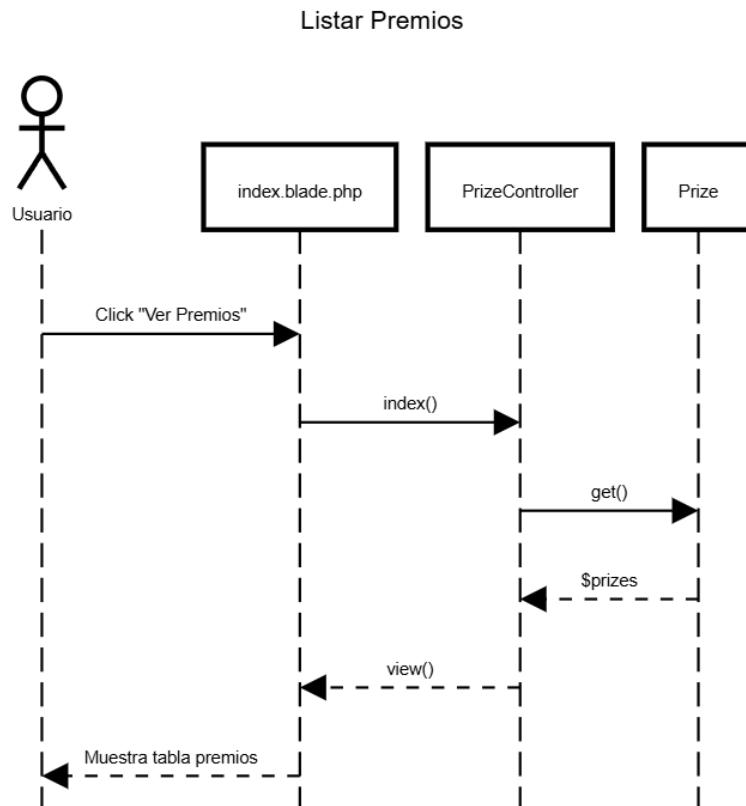


Fig. 33: Diagrama de secuencia para listar premios

En la figura 34, se muestra el diagrama de secuencia para crear premios.

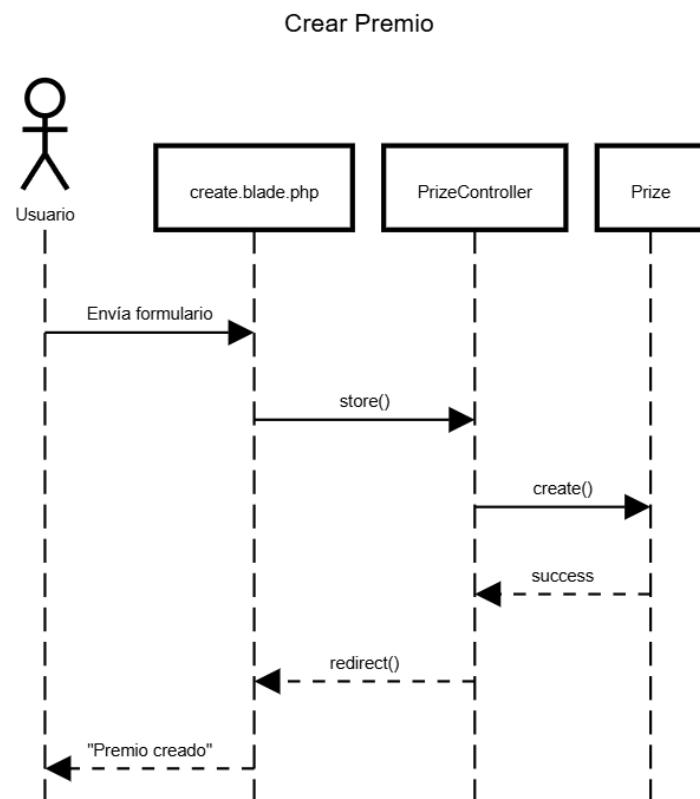


Fig. 34: Diagrama de secuencia para crear premios

En la figura 35, se muestra el diagrama de secuencia para reportes por día.

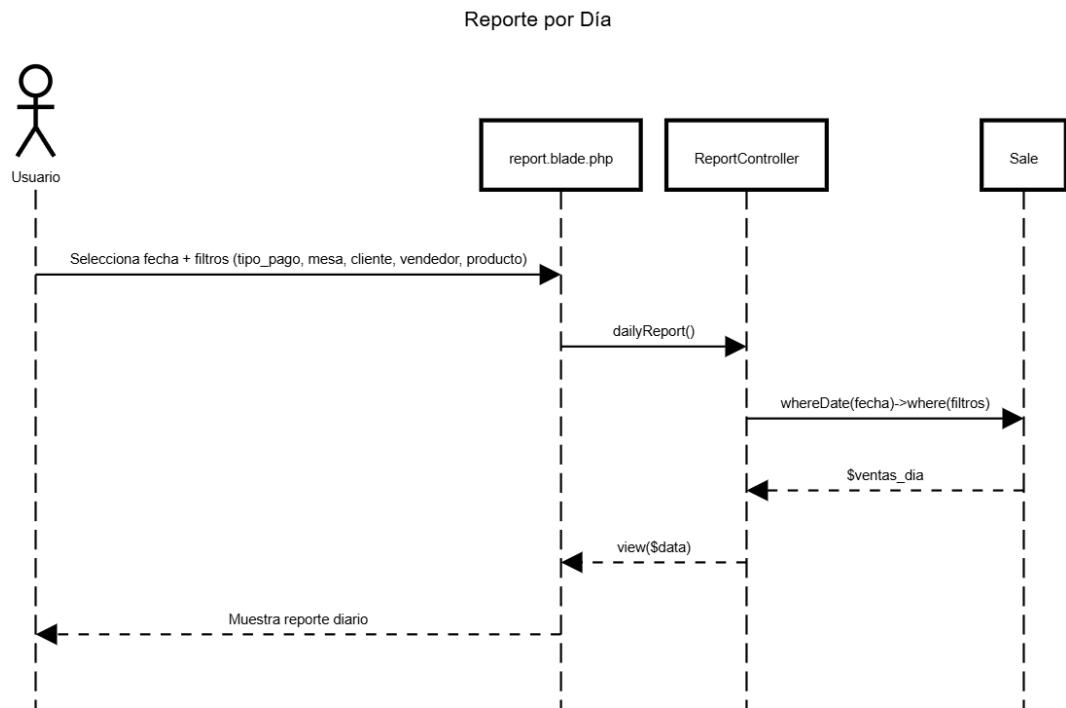


Fig. 35: Diagrama de secuencia para reportes por día

En la figura 36, se muestra el diagrama de secuencia para el reporte por rango de fechas.

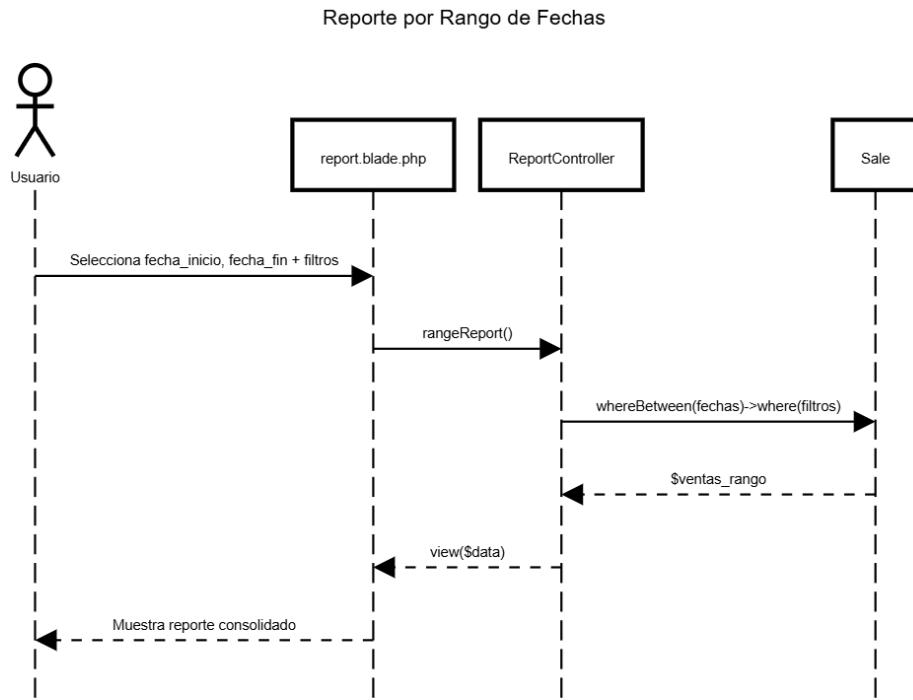


Fig. 36: Diagrama de secuencia para reporte por rango de fechas

En la figura 37, se muestra el diagrama para la lista de reclamos.

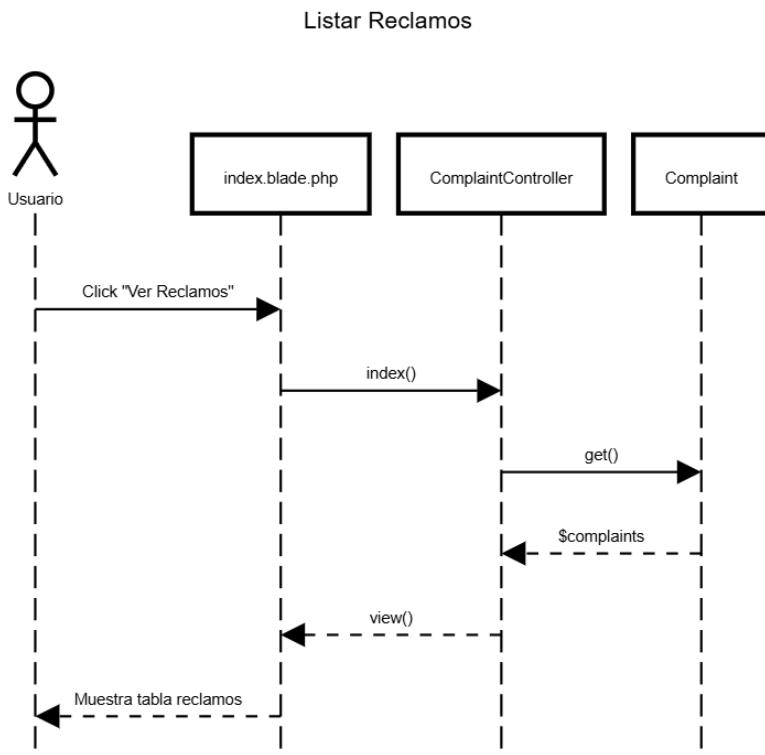


Fig. 37: Diagrama de secuencia para listar reclamos

En la figura 38, se muestra el diagrama de secuencia para registrar un reclamo.

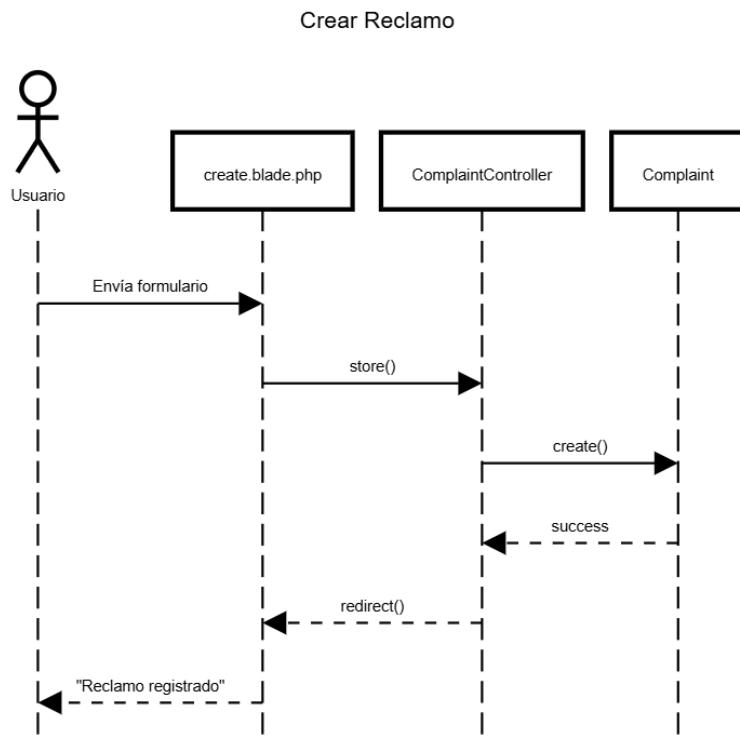


Fig. 38: Diagrama de secuencia para registrar reclamo

En la figura 39, se muestra el diagrama de secuencia para modificar la información de la empresa.

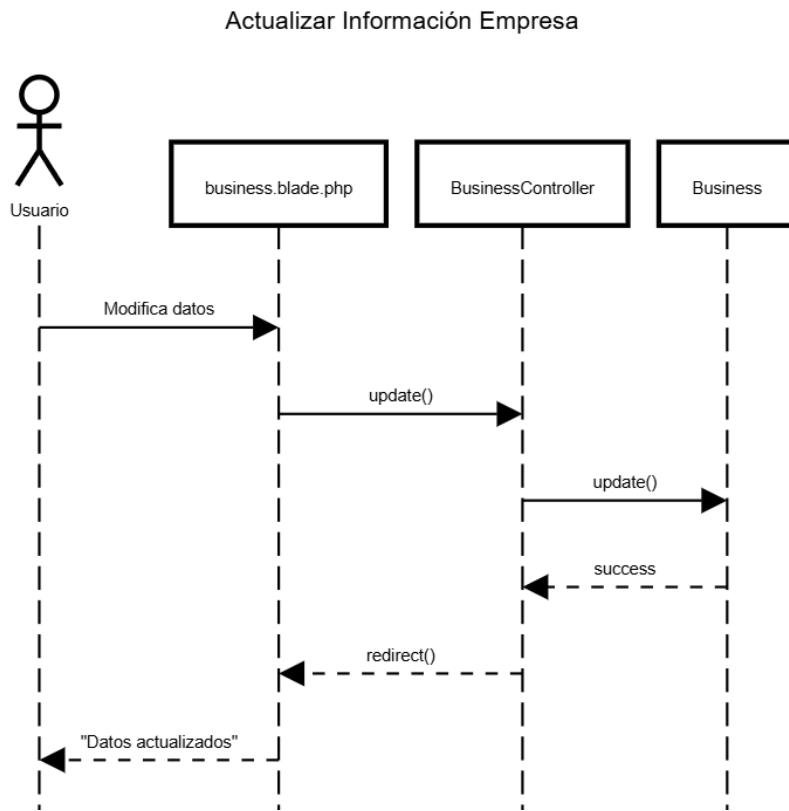


Fig. 39: Diagrama de secuencia para modificar información de la empresa

En la figura 40, se muestra el diagrama de secuencia de roles y permisos.

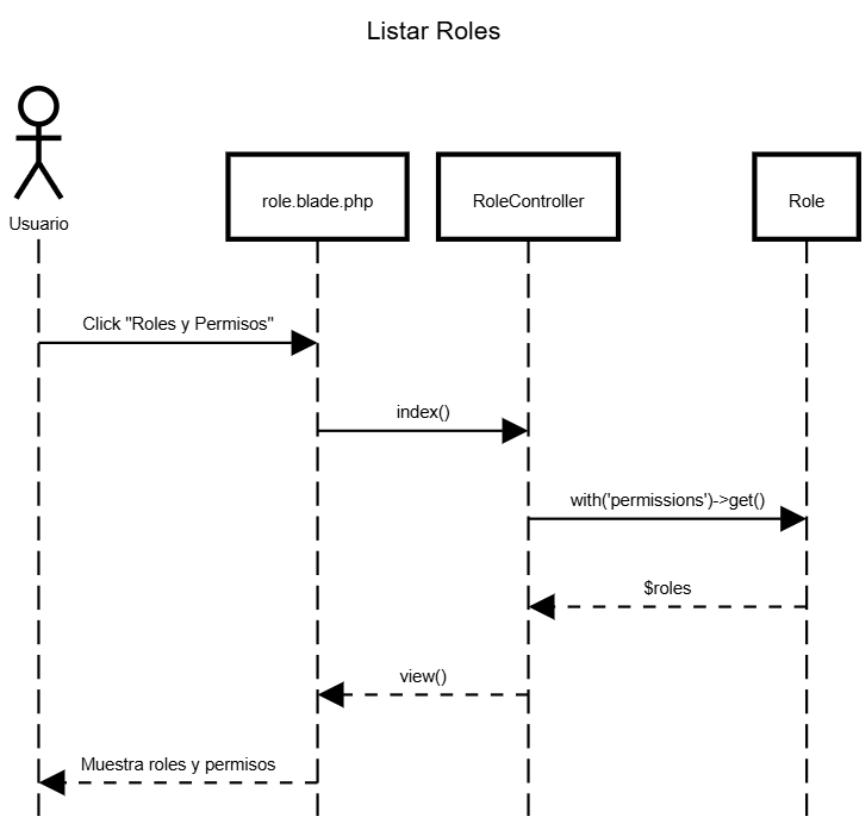


Fig. 40: Diagrama de secuencia de roles y permisos

Diseño conceptual de la base de datos

Para el diseño conceptual de la base de datos se identificaron las entidades clave que necesita la aplicación web y móvil mediante reuniones con los representantes de la empresa. En conjunto, se determinaron 11 entidades principales, que abarcan desde lo básico como los clientes y productos, hasta transacciones de las ventas. A continuación, la figura 20 muestra el modelo entidad-relación con todas las entidades y sus relaciones. Cabe mencionar que para “User”, “Role” y “Permission”, no se muestra debido a que esto es manejado por el propio framework Laravel.

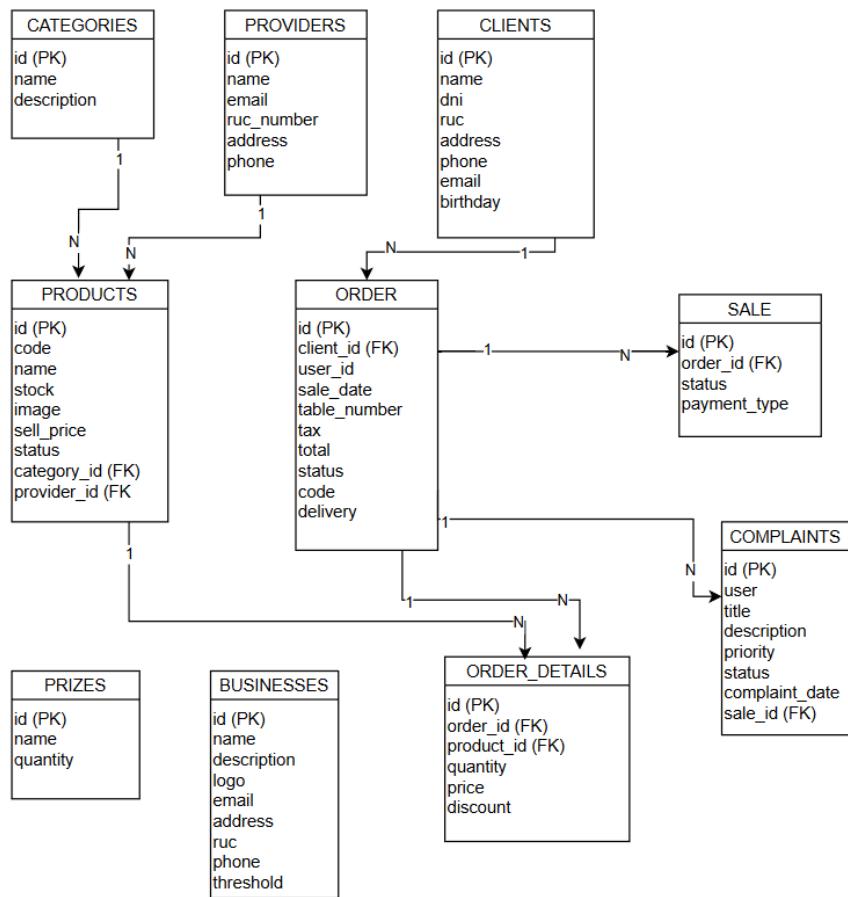


Fig. 41: Diseño conceptual de la base de datos

3.1.4 IMPLEMENTACIÓN

Instalación y configuración del framework

Luego de elegir las tecnologías a utilizar, se procedió a investigar en la documentación oficial la instalación de Laravel, para ello primero se instaló el IDE para el desarrollo del sistema, en este caso se utilizó Visual Studio Code, además se instaló Composer y Git.

En la figura 42, se muestra el comando para crear el proyecto con Laravel y en donde se va seleccionando las demás tecnologías a utilizar.

```

valdorio MINGW64 ~/Desktop
$ laravel new NikosPizza
[laravel]

Would you like to install a starter kit? [No starter kit]:
[none] No starter kit
[breeze] Laravel Breeze
[jetstream] Laravel Jetstream
> jetstream

Which Jetstream stack would you like to install? [Livewire]:
[livewire] Livewire
[inertia] Vue with Inertia
> livewire

Would you like any optional features? [None]:
[none] None
[api] API support
[dark] Dark mode
[verification] Email verification
[teams] Team support
> verification

Which testing framework do you prefer? [Pest]:
[0] Pest
[1] PHPUnit
>

Would you like to initialize a Git repository? (yes/no) [no]:
> yes
Creating a "laravel/laravel" project at "./NikosPizza"

```

Fig. 42: Instalación de Laravel

Implementación de la aplicación web

En la figura 43, se muestra la ubicación de los controladores en el proyecto NIKOSPIZZA en el framework Laravel.

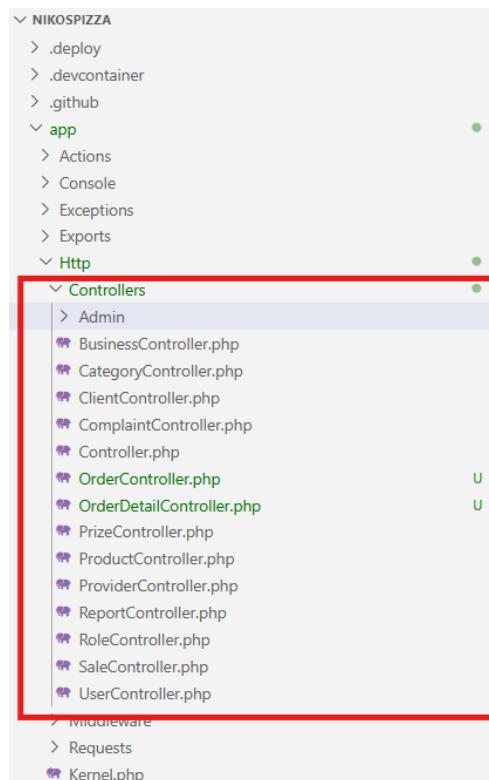


Fig. 43: Ubicación de controladores en el framework Laravel

En la figura 44, se muestra la ubicación de los modelos dentro del framework Laravel.

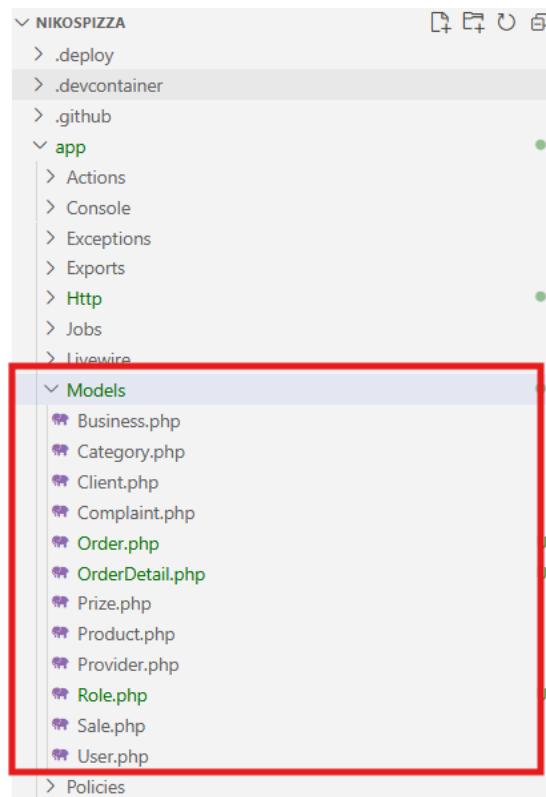


Fig. 44: Ubicación de modelos en el framework Laravel

En la figura 45, se muestra la ubicación de las vistas dentro del framework Laravel.

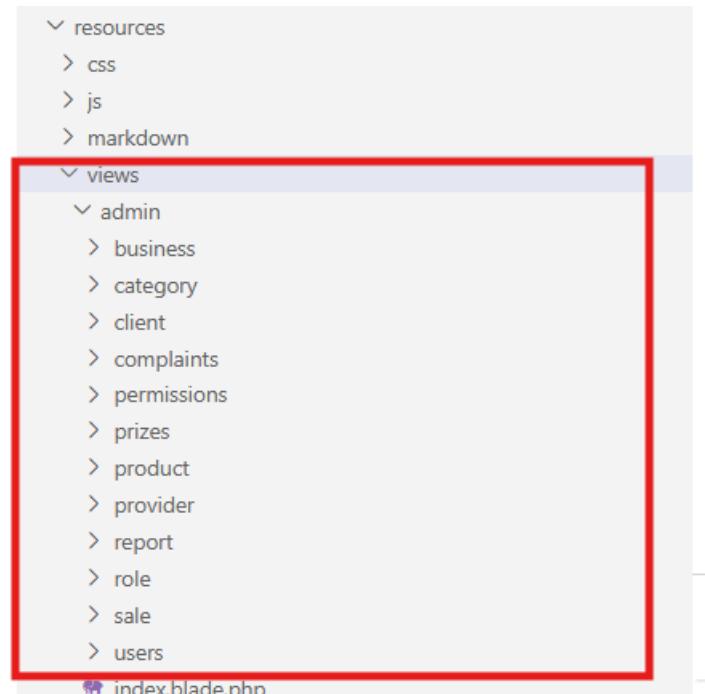


Fig. 45: Ubicación de las vistas en el framework Laravel

En la figura 46, se muestra parte del código del controlador correspondiente al index y el create de categorías.

```

public function index()
{
    $categories = Category::get();
    return view('admin.category.index', compact('categories'));
}

/**
 * Show the form for creating a new resource.
 */
public function create()
{
    return view('admin.category.create');
}

/**
 * Store a newly created resource in storage.
 */
public function store(StoreRequest $request)
{
    Category::create($request→all());
    return redirect()→route('categories.index')→with('info', 'Categoría creada correctamente');
}

```

Fig. 46: Código del controlador de categorías

En la figura 47, se muestra el código en la parte de crear una categoría.

```

@section('content')
<div class="row">
    <div class="col lg:12 grid-margin stretch-card">
        <div class="card">
            <div class="card-body">

                {!! Form::open(['route' => 'categories.store', 'method' => 'POST']) !!}

                @include('admin.category._form')

                <button type="submit" class="btn btn-primary mr-2 text-blue-100">
                    Registrar
                </button>
                <a href="{{route('categories.index')}}" class="btn btn-light">Cancelar</a>

                {!! Form::close() !!}

            </div>
        </div>
    </div>
</div>
@stop

```

Fig. 47: Código de crear categoría

En la figura 67 y 68 se muestra la vista correspondiente a categorías ya implementado.



ID	Nombre	Descripción	Acciones

Fig. 48: Vista de lista de categorías

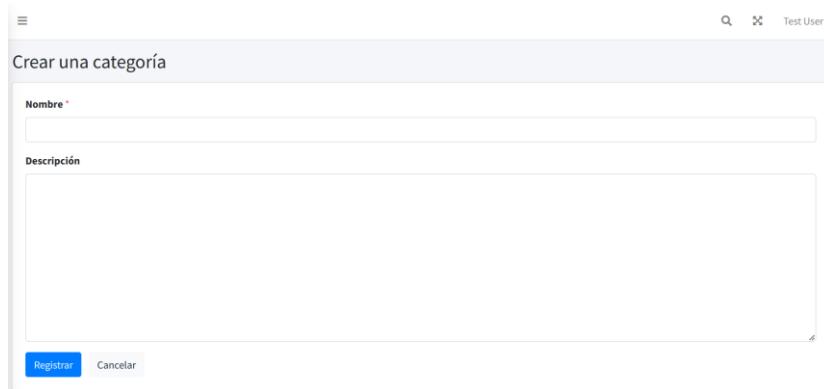


Fig. 49: Vista de crear categoría

En la figura 50 se muestra el controlador de proveedores y en la figura 51 se muestra el código de la vista de proveedores.

```
public function index()
{
    return view('admin.provider.index');
}

public function create()
{
    return view('admin.provider.create');
}

public function store(StoreRequest $request)
{
    Provider::create($request->all());
    return redirect()->route('providers.index')->with('info', 'Proveedor creado correctamente');
}

public function show(Provider $provider)
{
    return view('admin.provider.show', compact('provider'));
}
```

Fig. 50: Código de controlador de proveedores

```

<div class="card-body">

    { !! Form::open(['route' => 'providers.store', 'method' => 'POST']) !! }

    { !! Form::label('name', "Nombre") !!}
    <span class="text-red-500" title="Obligatorio">*</span>
    { !! Form::text('name', null, ['class' => 'form-control']) !!}
    @error('name')
        <span class="text-red-500 italic text-sm">{{$message}}</span>
        <br>
    @enderror

    { !! Form::label('email', "Email") !!}
    { !! Form::email('email', null, ['class' => 'form-control']) !!}
    @error('email')
        <span class="text-red-500 italic text-sm">{{$message}}</span>
        <br>
    @enderror

    { !! Form::label('ruc_number', 'Número de RUC') !!}
    { !! Form::number('ruc_number', null, ['class' => 'form-control']) !!}
    @error('ruc_number')
        <span class="text-red-500 italic text-sm">{{$message}}</span>
        <br>
    @enderror

    { !! Form::label('address', 'Dirección') !!}
    <span class="text-red-500" title="Obligatorio">*</span>
    { !! Form::text('address', null, ['class' => 'form-control']) !!}
    @error('address')
        <span class="text-red-500 italic text-sm">{{$message}}</span>
        <br>
    @enderror

    { !! Form::label('phone', 'Celular') !!}
    { !! Form::number('phone', null, ['class' => 'form-control']) !!}
    @error('phone')
        <span class="text-red-500 italic text-sm">{{$message}}</span>
        <br>
    @enderror

    <button type="submit" class="btn btn-primary mr-2 text-blue-300 mt-2">
        Registrar
    </button>
    <a href="{{route('providers.index')}}" class="btn btn-light mt-2">Cancelar</a>

    { !! Form::close() !!}

</div>

```

Fig. 51: Código de la vista de proveedores

En las figuras 52 y 53 se muestra la vista correspondiente a proveedores luego de codificar el controlador y la vista.

ID	Nombre	Correo electrónico	Teléfono/Celular	Acciones

Fig. 52: Vista implementada de proveedores (listar proveedores).

Crear un proveedor

Nombre *

Email

Número de RUC

Dirección *

Celular

Registrar Cancelar

Fig. 53: Vista implementada de proveedores (crear proveedor).

En la figura 54, se muestra el código del modelo correspondientes a productos.

```
app > Models > Product.php > ...
1 <?php
2
3 namespace App\Models;
4
5 use Illuminate\Database\Eloquent\Factories\HasFactory;
6 use Illuminate\Database\Eloquent\Model;
7
8 class Product extends Model
9 {
10     protected $guarded = ['id'];
11
12     use HasFactory;
13
14     public function category()
15     {
16         return $this->belongsTo(Category::class);
17     }
18
19     public function provider()
20     {
21         return $this->belongsTo(Provider::class);
22     }
23 }
24
```

Fig. 54: Código del modelo de producto

En la figura 55, se muestra parte del código del controlador correspondiente a productos.

```

public function index()
{
    return view('admin.product.index');
}

public function create()
{
    $categories = Category::all()→pluck('name', 'id');
    $providers = Provider::all()→pluck('name', 'id')→prepend('Sin proveedor', '');
    return view('admin.product.create', compact('categories', 'providers'));
}

public function store(StoreProductRequest $request)
{
    $url = "";
    if($request→file('image')){
        $url = Storage::put('public/products', $request→file('image'));
    }

    $product = Product::create($request→all());
    $product→update([
        'image' => $url
    ]);
    return redirect()→route('products.index')→with('info', 'Producto creado correctamente');
}

```

Fig. 55: Código de controlador de productos

En la figura 56, se muestra el código de la vista correspondiente a la lista de productos en el index y en la figura 57 se muestra el código de la vista de crear un producto.

```

@extends('adminlte::page')

@section('title', 'Productos')

@section('content_header')
    <h1>Lista de Productos</h1>
@stop

@section('content')
@if (session('info'))
    <div class="alert alert-info">
        {{session('info')}}
    </div>
@endif


<a href="{{route('products.create')}}" class="bg-green-500 hover:bg-green-700 text-white font-bold py-2 px-4 rounded-full">Agregar nuevo producto</a>


@livewire('admin.product-index')
@stop

@section('css')
    <link rel="stylesheet" href="/css/admin_custom.css">
    <script src="https://cdn.tailwindcss.com"></script>

```

```

@section('js')
<script>
    function confirmDelete(id) {
        if (confirm('¿Estás seguro de que quieres eliminar este elemento?')) {
            document.getElementById('delete-form-' + id).submit();
        }
    }
</script>
@stop

```

Fig. 56: Código de vista del index de productos

```

<div class="card-body">

{ !! Form::open(['route' => 'products.store', 'method' => 'POST', 'files' => true]) !!}

{ !! Form::label('name', "Nombre") !!}
<span class="text-red-500" title="Obligatorio">*</span>
{ !! Form::text('name', null, ['class' => 'form-control']) !!}
@error('name')
<span class="text-red-500 italic text-sm">{{ $message }}</span>
<br>
@enderror

{ !! Form::label('code', "Código") !!}
<span class="text-red-500" title="Obligatorio">*</span>
{ !! Form::text('code', null, ['class' => 'form-control']) !!}
@error('code')
<span class="text-red-500 italic text-sm">{{ $message }}</span>
<br>
@enderror

{ !! Form::label('stock', "Stock") !!}
<span class="text-red-500" title="Obligatorio">*</span>
{ !! Form::number('stock', null, ['class' => 'form-control']) !!}
@error('stock')
<span class="text-red-500 italic text-sm">{{ $message }}</span>
<br>
@enderror

{ !! Form::label('sell_price', 'Precio de Venta') !!}
<span class="text-red-500" title="Obligatorio">*</span>
{ !! Form::number('sell_price', null, ['class' => 'form-control', 'step' => 'any']) !!}
@error('sell_price')
<span class="text-red-500 italic text-sm">{{ $message }}</span>
<br>
@enderror

{ !! Form::label('category_id', 'Categoria') !!}
<span class="text-red-500" title="Obligatorio">*</span>
{ !! Form::select('category_id', $categories, null, ['class' => 'form-control']) !!}
@error('category_id')
<span class="text-red-500 italic text-sm">{{ $message }}</span>
<br>
@enderror

{ !! Form::label('provider_id', 'Proveedor') !!}
{ !! Form::select('provider_id', $providers, null, ['class' => 'form-control mb-2']) !!}
@error('provider_id')
<span class="text-red-500 italic text-sm">{{ $message }}</span>
<br>
@enderror

<button type="submit" class="btn btn-primary mr-2 text-blue-300 mt-2">
    Registrar
</button>
<a href="{{ route('products.index') }}" class="btn btn-light mt-2">Cancelar</a>

{ !! Form::close() !!}

</div>

```

Fig. 57: Código de vista de crear productos

En la figura 58 y la figura 59 se muestra la vista correspondiente a productos ya implementado.

ID	Nombre	Stock	Estado	Categoría	Acciones
----	--------	-------	--------	-----------	----------

Fig. 58: Vista implementada de productos (listar productos).

Nombre *

Código *

Stock *

Precio de Venta *

Categoría *

Pizzas

Proveedor

Sin proveedor

Registrar Cancelar

Fig. 59: Vista implementada de productos (crear producto).

En la figura 60, se muestra parte del código del controlador correspondiente a clientes.

```

public function index()
{
    // Obtener la fecha actual
    $dateNow = Carbon::now();
    // Obtener el día y el mes actual
    $day = $dateNow→day;
    $month = $dateNow→month;
    // Buscar clientes cuya fecha de cumpleaños coincide con el día y mes actual
    $clientsBirthday = Client::whereDay('birthday', $day)
        →whereMonth('birthday', $month)
        →get();
    $clients = Client::get();
    return view('admin.client.index', compact('clients', 'clientsBirthday'));
}

public function create()
{
    return view('admin.client.create');
}

public function store(StoreClientRequest $request)
{
    Client::create($request→all());
    return redirect()→route('clients.index')→with('info', 'Cliente creado correctamente');
}

```

Fig. 60: Código de controlador de clientes

En la figura 61, se muestra el código de la lista de clientes que se muestra en el index.

```

@section('content')
@if (session('info'))
    <div class="alert alert-info">
        {{ session('info') }}
    </div>
@endif

@if ($clientsBirthday→isNotEmpty())
    
    <script>
        document.addEventListener('DOMContentLoaded', function() {
            var message = "<ul>"; <!-- Iniciamos la lista de clientes-- >
            @foreach ($clientsBirthday as $client)
                message += "<li class='bg-gray-300 mb-1 rounded-md'>{{ $client→name }}</li>"; <!--
                    Agregamos cada cliente a la lista-- >
            @endforeach
            message += "</ul>"; <!-- Cerramos la lista de clientes-- >

            Swal.fire({
                title: 'Clientes con cumpleaños',
                html: message,
                icon: 'info',
                confirmButtonText: 'OK'
            });
        });
    </script>
@endif

<div class="my-4">
    <a href="{{ route('clients.create') }}"
        class="bg-green-500 hover:bg-green-700 text-white font-bold py-2 px-4 rounded-full">Agregar nuevo cliente</a>
</div>
@livewire('admin.client-index')
@stop

```

Fig. 61: Código de vista de clientes

En las figuras 62 y 63 se muestra la vista correspondiente a clientes ya implementado.

ID	Nombre	DNI	Dirección	Teléfono	Opciones
----	--------	-----	-----------	----------	----------

Fig. 62: Vista implementada de clientes (listar clientes).

Crear un cliente

Nombre *

DNI

Ruc

Dirección

Celular

Email

Fecha de nacimiento

dd/mm/aaaa

Registrar Cancelar

Fig. 63: Vista implementada de clientes (crear cliente).

En la figura 64, se muestra el código del modelo order.

```

app > Models > Order.php > ...
1  <?php
2
3  namespace App\Models;
4
5  use Carbon\Carbon;
6  use Illuminate\Database\Eloquent\Factories\HasFactory;
7  use Illuminate\Database\Eloquent\Model;
8
9  class Order extends Model
10 {
11     protected $table = 'sales'; // This model uses the 'sales' table
12     protected $guarded = ['id'];
13
14     use HasFactory;
15
16     protected static function booted()
17     {
18         static::creating(function ($order) {
19             // The column in the database is still 'sale_date'
20             $order->code = 'SALE-' . Carbon::parse($order->sale_date)->format('Ymd').Carbon::now()->format('His');
21         });
22     }
23
24     public function user()
25     {
26         return $this->belongsTo(User::class);
27     }
28
29     public function client()
30     {
31         return $this->belongsTo(Client::class);
32     }
33
34     // Renamed relationship to reflect the new model name
35     public function orderDetails()
36     {
37         // The foreign key in the sale_details table is still 'sale_id'
38         return $this->hasMany(OrderDetail::class, 'sale_id');
39     }
40
41     public function complaints()
42     {
43         return $this->hasMany(Complaint::class);
44     }
45 }

```

Fig. 64: Código del modelo orden

En la figura 65, se muestra el código del controlador Order.

```

21  class OrderController extends Controller
22
23      public function index()
24      {
25          return view('admin.order.index');
26      }
27
28      public function create()
29      {
30          $clients = Client::get()->pluck('name', 'id')->prepend('Sin cliente', '');
31          $products = Product::where('status', 'ACTIVE')->get();
32
33          return view('admin.order.create', compact('clients', 'products'));
34      }
35
36      public function store(StoreSaleRequest $request)
37      {
38          $orden = Order::create($request->all()) + [
39              'user_id' => Auth::user()->id,
40              'sale_date' => Carbon::parse($request->sale_date)->setTimezone('America/Lima'),
41          ];
42
43          $results = [];
44          foreach ($request->product_id as $key => $product) {
45              $results[] = [
46                  'product_id' => $request->product_id[$key],
47                  'quantity' => $request->quantity[$key],
48                  'price' => $request->price[$key],
49                  'discount' => $request->discount[$key],
50              ];
51          }
52          $orden->orderDetails()->createMany($results);
53
54          $info = '';
55          $title = '';
56
57          if ($request->client_id) {
58              $customerPizzas = OrderDetail::whereHas('order', function ($query) use ($request) {
59                  $query->where('client_id', $request->client_id);
60              })
61              ->join('products', 'order_details.product_id', '=', 'products.id')
62              ->where('products.category_id', '1')
63              ->sum('quantity');
64
65              $closestPrize = Prize::where('quantity', '>=', $customerPizzas)
66              ->orderBy('quantity')
67              ->first();
68
69              if ($closestPrize) {
70                  $closestPrizeName = $closestPrize->name;
71                  $client = Client::find($request->client_id);
72                  if ($closestPrize->quantity == $customerPizzas) {
73                      $info = 'El cliente '.$client->name.' ha ganado el premio: '.$closestPrizeName;
74                      $title = 'Felicitaciones!!!';
75                  } else {
76
77
78
79
80
81
82
83
84
85
86
87
88

```

Fig. 65: Código del controlador orden

En la figura 66, se muestra la vista de la lista de pedidos y en la figura 67 se muestra la vista con el formulario para crear un pedido.

ID	Código	Cliente	Total	Estado	Tipo de Pago	Acciones

Fig. 66: Vista implementada de pedidos

Eliminar	Producto	Precio de venta	Descuento	Cantidad	SubTotal
Total:					PEN 0.00
Total Impuesto:					PEN 0.00
Total Pagar:					PEN 0.00

Fig. 67: Vista implementada de crear un pedido

En la figura 68, se muestra el código del modelo Sales.

```

Sale.php  x
app > Models > Sale.php > ...
1  <?php
2
3  namespace App\Models;
4
5  use Carbon\Carbon;
6  use Illuminate\Database\Eloquent\Factories\HasFactory;
7  use Illuminate\Database\Eloquent\Model;
8
9  class Sale extends Model
10 {
11     protected $guarded = ['id'];
12
13     use HasFactory;
14
15     protected static function booted()
16     {
17         static::creating(function ($sale) {
18             // $sale->code = 'SALE-' . Carbon::parse($sale->sale_date)->format('YmdHis');
19             $sale->code = 'SALE-' . Carbon::parse($sale->sale_date)->format('Ymd').Carbon::now()->format('His');
20         });
21     }
22
23     public function user()
24     {
25         return $this->belongsTo(User::class);
26     }
27
28     public function client()
29     {
30         return $this->belongsTo(Client::class);
31     }
32
33     public function saleDetails()
34     {
35         return $this->hasMany(SaleDetail::class);
36     }
37
38     public function complaints()
39     {
40         return $this->hasMany(Complaint::class);
41     }
42 }
43

```

Fig. 68: Código del modelo venta

En la figura 69, se muestra el código del controlador de sales

```

app > Http > Controllers > SaleController.php > SaleController
1  <?php
2
3  namespace App\Http\Controllers;
4
5  use App\Models\Order;
6  use App\Models\Sale;
7  use Illuminate\Http\Request;
8
9  class SaleController extends Controller
10 {
11
12     public function index()
13     {
14         $sales = Sale::with('order')->latest()->paginate(20);
15         return view('admin.sale.index', compact('sales'));
16     }
17
18     public function store(Order $order)
19     {
20         $sale = Sale::create([
21             'order_id' => $order->id,
22             'type_payment' => $order->payment_type,
23             'status' => 'COMPLETED',
24         ]);
25
26         return $sale;
27     }
28
29     public function show(Sale $sale)
30     {
31         $sale->load('order.orderDetails.product', 'order.client');
32     }
33 }

```

Fig. 69: Código del controlador de venta

En la figura 70 se muestra la vista del index de ventas y en la figura 71 se muestra la vista del create de una venta.

Fig. 70: Vista implementada de ventas (listar ventas)

Fig. 71: Vista implementada de ventas (crear venta).

En la figura 72, se muestra la vista del cobro por mesas para los pedidos, cuando se presiona en el botón buscar, muestra la lista de los pedidos pendientes de pago por mesa o por delivery.

Fig. 72: Vista de cobrar por mesa

En la figura 73, se muestra cuando ya fue encontrado la venta y tiene dos opciones de ir a al pedido o pagar y muestra el modal para seleccionar el método de pago.

Fig. 73: Vista de pedido encontrado para cobrar por mesa

En la figura 74, se muestra el modal para seleccionar el método de pago.

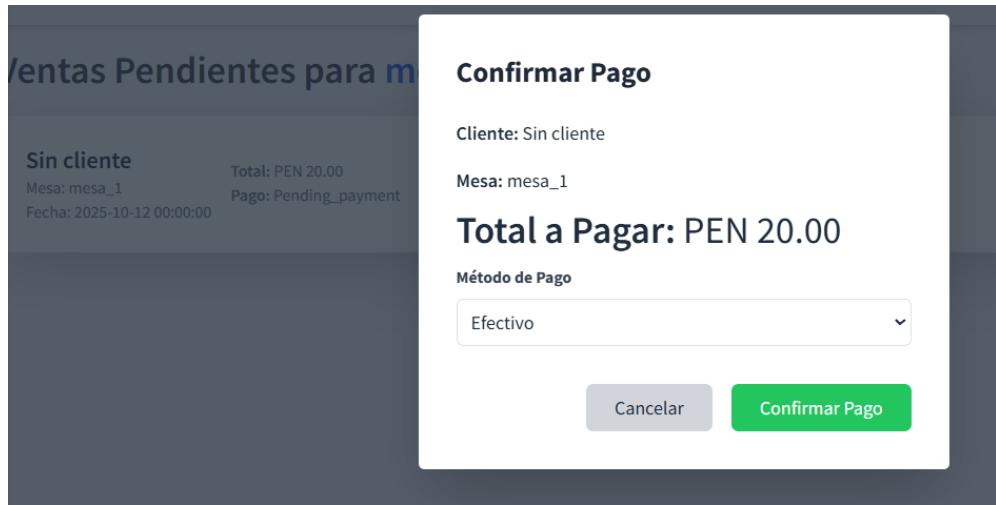


Fig. 74: Modal para seleccionar el método de pago

En la figura 75, se muestra parte del código de la vista de la información de la empresa.

```

<div class="card-body">
    <div class="items-start">
        <strong>Nombre de la empresa: </strong>
        <p>{{\$business→name}}</p>
        <hr>
    </div>
    <div class="items-start">
        <strong>Ruc: </strong>
        <p>{{\$business→ruc}}</p>
        <hr>
    </div>
    <div class="items-start">
        <strong>Email: </strong>
        <p>{{\$business→email}}</p>
        <hr>
    </div>
    <div class="items-start">
        <strong>Dirección: </strong>
        <p>{{\$business→address}}</p>
        <hr>
    </div>
    <div class="items-start">
        <strong>Celular: </strong>
        <p>{{\$business→phone}}</p>
        <hr>
    </div>
    <div class="items-start">
        <strong>Umbral de ventas: </strong>
        <p>S/. {{\$business→threshold}}</p>
        <hr>
    </div>
    <div class="items-start">
        <strong>Logo</strong>
        
    </div>
</div>

```

Fig. 75: Código de la vista para la información de la empresa.

En las figuras 76 y 77 se muestra la vista para registrar y actualizar la información de la empresa.

Nombre de la empresa:
Nikos Pizza

Ruc:
10123456789

Email:
nikostest@gmail.com

Dirección:
Jr. Reyna Farge N°

Celular:
987456321

Umbral de ventas:
S/. 0

Logo



Actualizar

Fig. 76: Vista implementada para la información de la empresa.

Nombre
Nikos Pizza

Descripción
Pizzería en General

Correo
nikostest@gmail.com

Dirección
Jr. Reyna Farge N°

Celular
987456321

Ruc
10123456789

Umbral de ventas (S/.)
0

Logo de la empresa Ningún archivo seleccionado

Actualizar **Cancel**

Fig. 77: Vista implementada para actualizar la información de la empresa.

En la figura 78, 79 se muestra parte del código del controlador y la vista de los reclamos respectivamente.

```

public function index()
{
    return view('admin.complaints.index');
}

public function create()
{
    return view('admin.complaints.create');
}

public function store(StoreComplaintRequest $request)
{
    Complaint::create($request->all() +[
        'complaint_date' =>Carbon::now('America/Lima')
    ]);

    return redirect()->route('complaints.index')->with('info', 'Reclamo creado correctamente');
}

public function show(Complaint $complaint)
{
    return view('admin.complaints.show', compact('complaint'));
}

```

Fig. 78: Código del controlador de reclamos

```

@extends('adminlte::page')

@section('title', 'Reclamos')

@section('content_header')
    <h1>Lista de Reclamos</h1>
@stop

@section('content')
@if (session('info'))
    <div class="alert alert-info">
        {{ session('info') }}
    </div>
@endif


<a href="{{ route('complaints.create') }}"
        class="bg-green-500 hover:bg-green-700 text-white font-bold py-2 px-4 rounded-full">Agregar nuevo reclamo</a>


@livewire('admin.complaint-index')
@stop

@section('css')
    <link rel="stylesheet" href="/css/admin_custom.css">
    <script src="https://cdn.tailwindcss.com"></script>

    <script>
        function confirmDelete(id) {
            if (confirm('¿Estás seguro de que quieres eliminar este elemento?')) {
                document.getElementById('delete-form-' + id).submit();
            }
        }
    </script>
@stop

@section('js')
@stop

```

Fig. 79: Código de la vista de reclamos

En la figura 80 se muestra la vista implementada de los reclamos, en la figura 81 se muestra la vista para registrar un reclamo, el pedido es opcional, en caso sea un reclamo general, pero si se trata de un reclamo hacia un pedido, en el input se ingresa el código del pedido y lo va buscando.

Lista de Reclamos

[Agregar nuevo reclamo](#)

Escriba el usuario o el título

ID	Usuario	Título	Estado	Prioridad	Acciones
----	---------	--------	--------	-----------	----------

Fig. 80: Vista implementada para listar reclamos

Crear un reclamo

Nombre de usuario *

Título *

Descripción *

Prioridad *

Bajo

Pedido (Opcional)

Buscar por código de pedido...

[Registrar](#) [Cancelar](#)

Fig. 81: Vista implementada para crear reclamos.

En la figura 82, se muestra el código del modelo de premios.

```

app > Models > Prize.php > ...
1  <?php
2
3  namespace App\Models;
4
5  use Illuminate\Database\Eloquent\Factories\HasFactory;
6  use Illuminate\Database\Eloquent\Model;
7
8  class Prize extends Model
9  {
10     protected $guarded = ['id'];
11
12     use HasFactory;
13 }
14

```

Fig. 82: Código del modelo de premios.

En la figura 83, se muestra el código del controlador de premios.

```

app > Http > Controllers > PrizeController.php > PrizeController
10  class PrizeController extends Controller
11
12  public function index()
13  {
14      $prizes = Prize::orderByDesc('created_at')
15          ->get();
16
17      return view('admin.prizes.index', compact('prizes'));
18  }
19
20  public function create()
21  {
22      return view('admin.prizes.create');
23  }
24
25
26  public function store(StorePrizeRequest $request)
27  {
28      Prize::create($request->all());
29
30      return redirect()->route('prizes.index')->with('info', 'Premio creado correctamente');
31  }
32
33  public function show(Prize $prize)
34  {
35      //
36  }
37
38  public function edit(Prize $prize)
39  {
40      return view('admin.prizes.edit', compact('prize'));
41  }
42
43
44  public function update(UpdatePrizeRequest $request, Prize $prize)
45  {
46      $prize->update($request->all());
47
48      return redirect()->route('prizes.index')->with('info', 'Premio actualizado correctamente');
49  }
50
51
52
53
54
55
56
57
58
59

```

Fig. 83: Código del controlador de premios

En las figuras 84 y 85 se muestra la vista implementada correspondiente a premios.

Lista de Premios

[Agregar nuevo premio](#)

ID	Nombre	Cantidad	Acciones
----	--------	----------	----------

Fig. 84: Vista implementada para listar premios.

Crear un premio

Nombre del premio *

Cantidad *

Registrar

Cancelar

Fig. 85: Vista implementada para crear premios

Finalmente, en la figura 86 se muestra la estructura de la base de datos en general, en donde incluye tanto las tablas que son de utilidad para la aplicación como también las tabla que utiliza Laravel para manejar las migraciones y las sesiones.

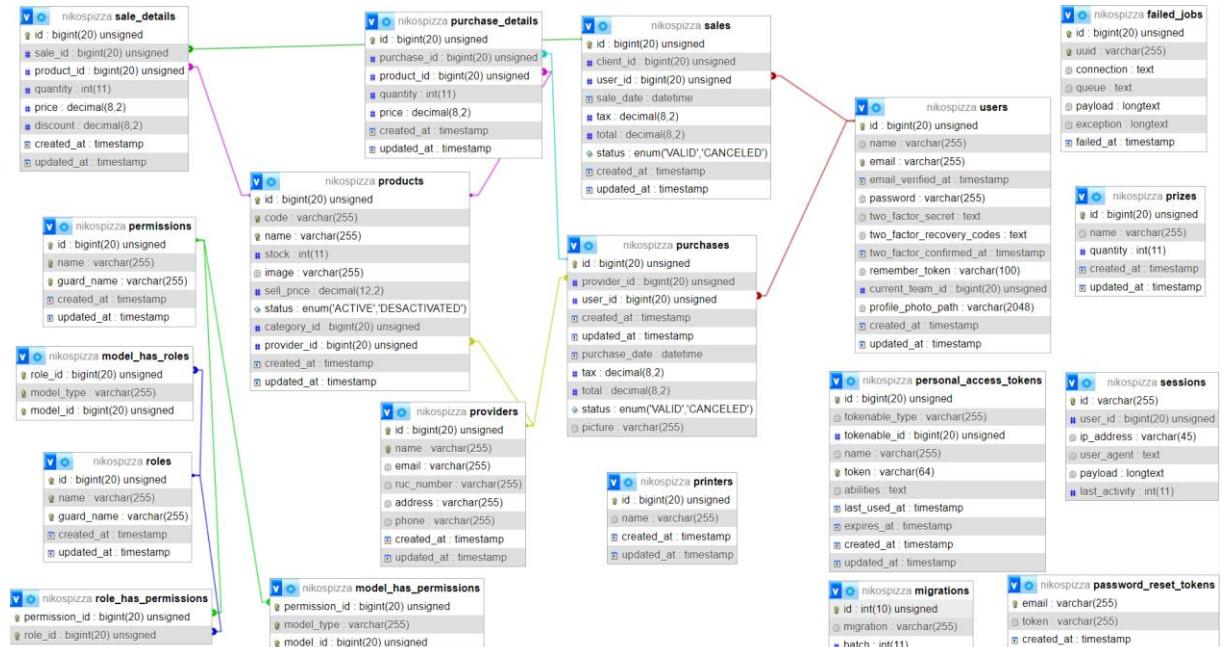


Fig. 86: Esquema de la base de datos

Luego de tener la estructura básica de la aplicación se procedió a programar los detalles de la aplicación. En la figura 87, se muestra el index principal, en donde se encuentra las opciones de inicio de sesión y de crear cuenta.

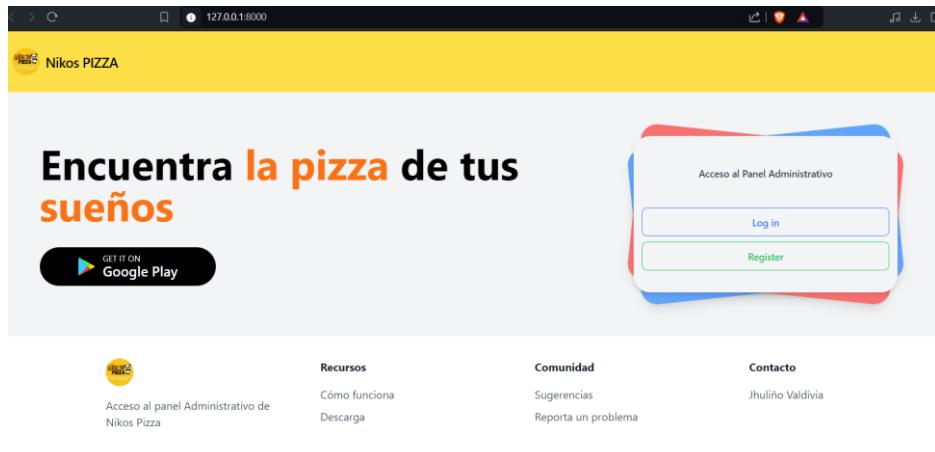


Fig. 87: Vista del index principal de la aplicación

En la figura 88, se muestra la vista de inicio de sesión, esto luego de presionar el botón de inicio de sesión redirige a un formulario donde se ingresa el correo y la contraseña, cabe mencionar que no se puede acceder a ninguna parte del sistema si no se está autenticado.

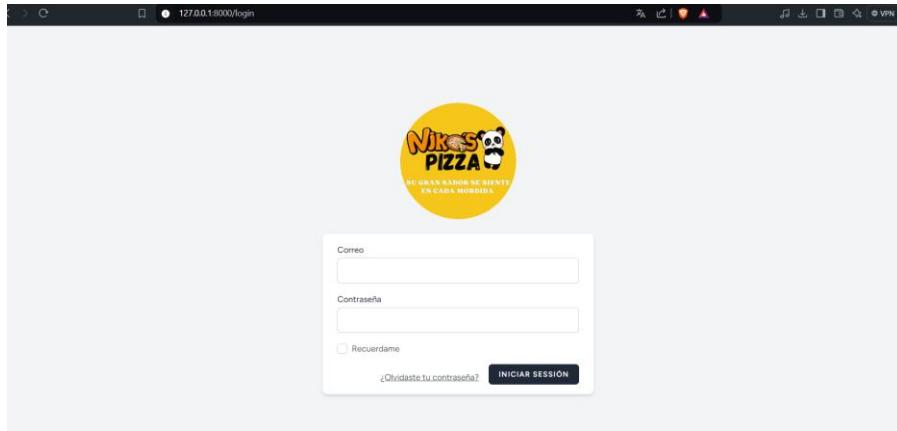


Fig. 88: Vista de inicio de sesión

Luego de que el sistema verifica que las credenciales son correctas redirige hacia el dashboard, se muestra el total de ventas en el mes actual, luego los gráficos de barras indican por cada mes las ventas durante el año. El dashboard se aprecia en la figura 89.

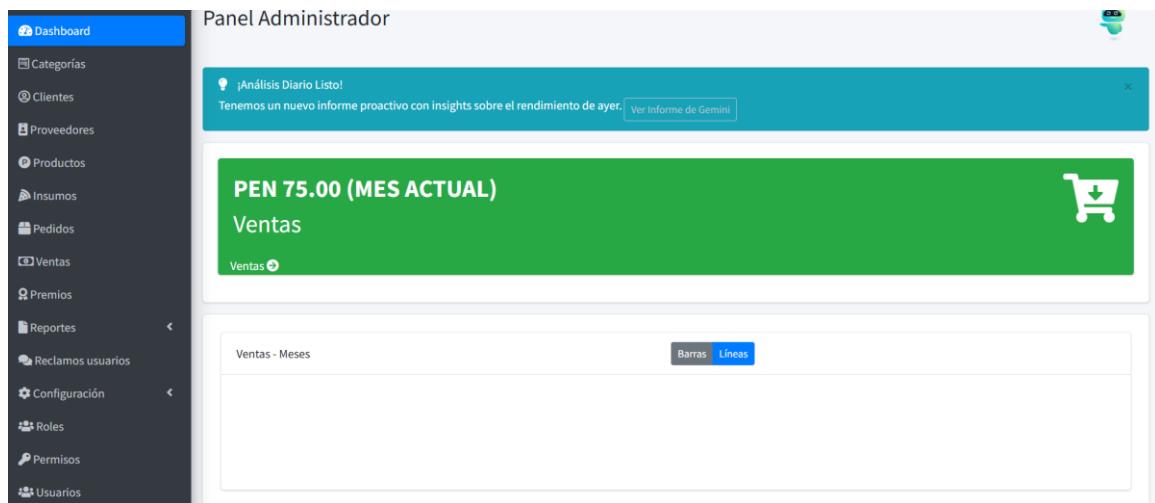


Fig. 89: Vista del dashboard, parte superior

La figura 90, corresponde al dashboard también, pero es la parte inferior en donde se muestra en un gráfico de líneas las ventas por día, durante el mes en que se encuentran, además si la venta a superado el punto de equilibrio saldrá de color verde sino de color rojo. Mas abajo se encuentra una tabla con los productos más vendidos.

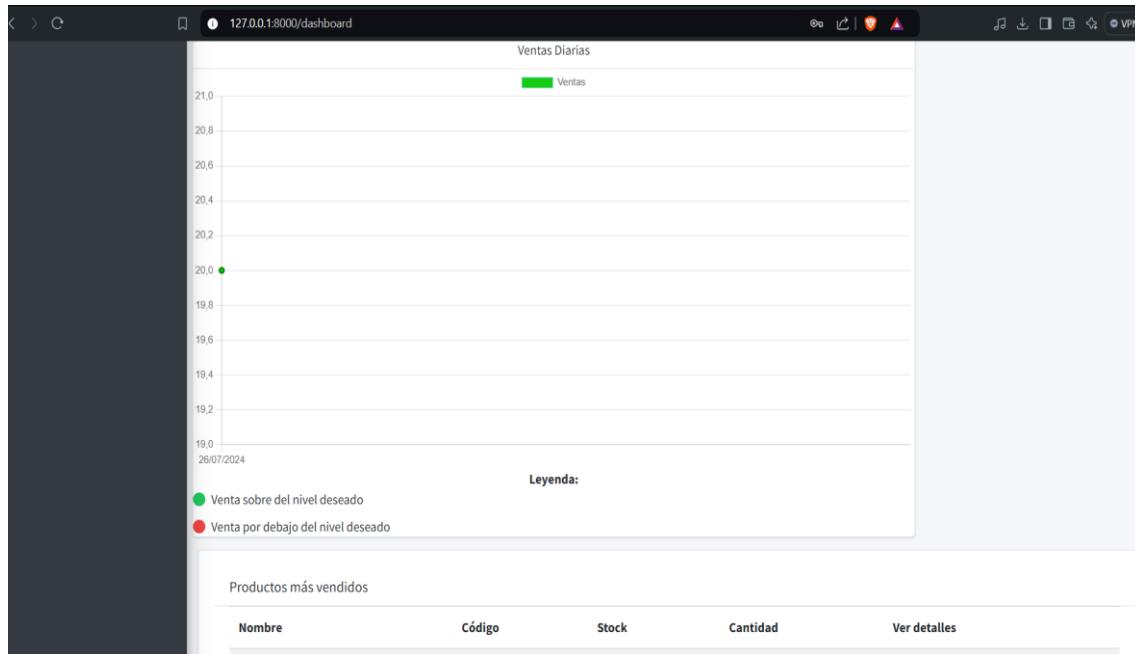


Fig. 90: Vista del dashboard, parte inferior

En la figura 91, se muestra la vista de los reportes por rango de fechas, cabe mencionar que para poder visualizar las ventas de ese rango de fecha es necesario presionar en el botón de consultar, y no es necesario llenar los demás campos, en cambio cuando ya se ha dado consultar y hay datos, el seleccionar los diferentes filtros se va mostrando de acuerdo a ello sin necesidad de presionar consultar.

Reporte de ventas por día

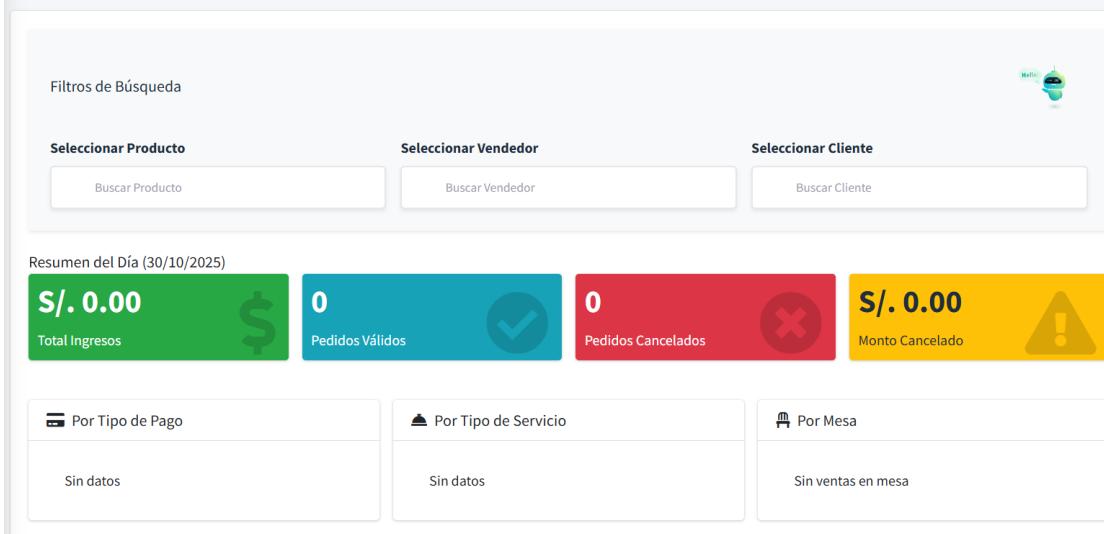


Fig. 91: Vista de reporte de ventas por rango de fecha.

En la figura 92, se muestra la vista de los roles que se va a contar en el sistema, es decir, cada usuario puede tener cierto rol que este asociado con ciertos permisos de lo que puede hacer en el sistema.

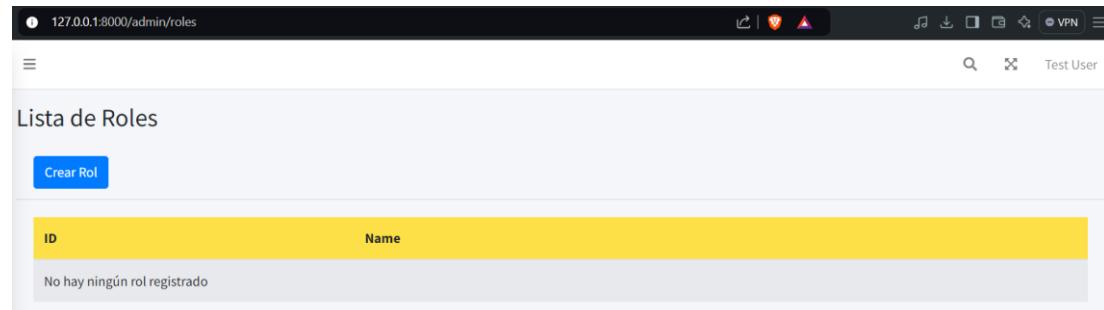


Fig. 92: Vista de roles

En la figura 93, se muestra la vista de la lista de usuario de la aplicación.

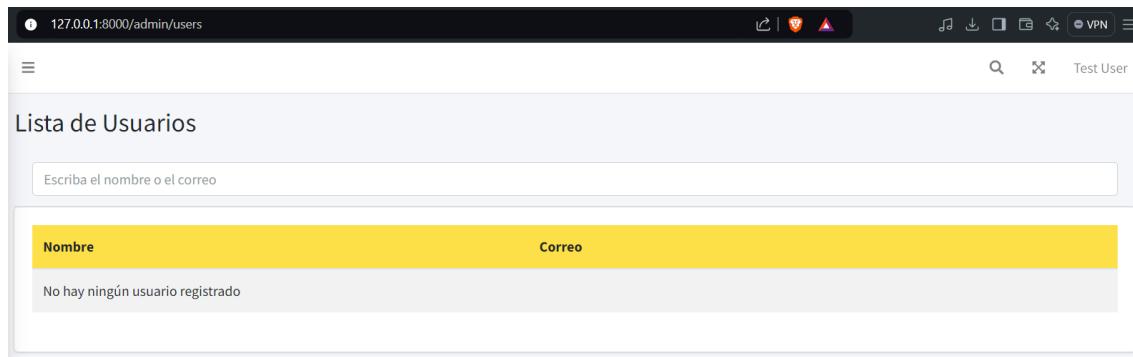


Fig. 93: Vista de usuarios de la aplicación.

En la figura 94, se muestra la vista principal de perfil del usuario y en donde el usuario puede cambiar sus datos como su nombre, correo o incluso su contraseña.

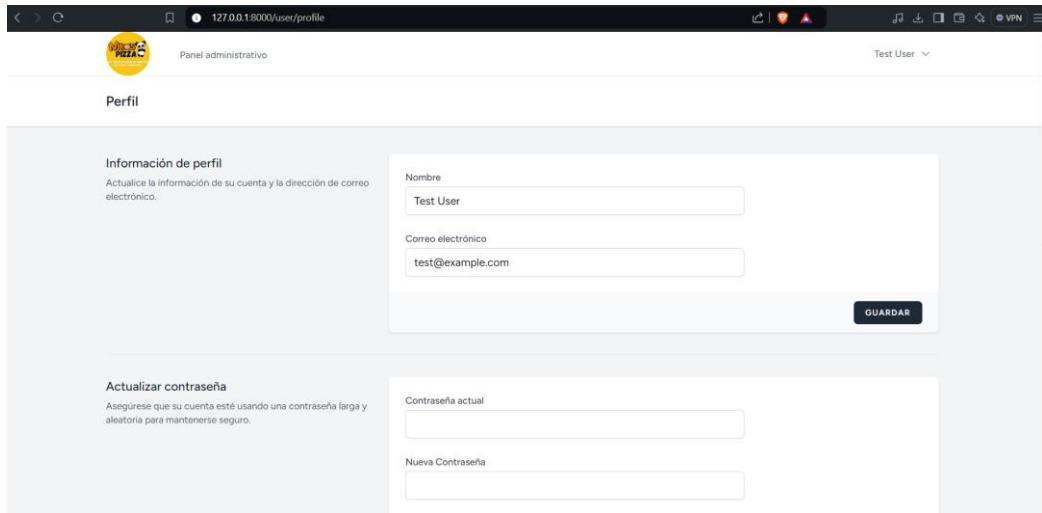


Fig. 94: Vista del perfil del usuario

En la figura 95, se muestra el icono que al presionar muestra el modal para poder preguntar a la inteligencia artificial. Este se encuentra en la parte superior derecha del dashboard.

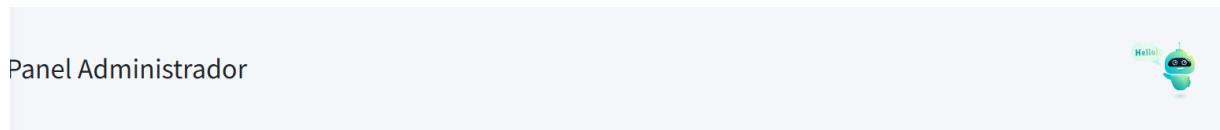


Fig. 95: Ubicación del icono para preguntar a la I.A.

En la figura 96 se muestra el modal en donde se ingresa la pregunta sobre las ventas a enviar a la I.A. respecto a la data mostrada en el dashboard. Al presionar en el botón de enviar, la pregunta sobre las ventas se envía al controlador que a su vez obtiene data de la base de datos y conjuntamente envía esa información a Gemini a través de su api, para luego mostrar el resultado.

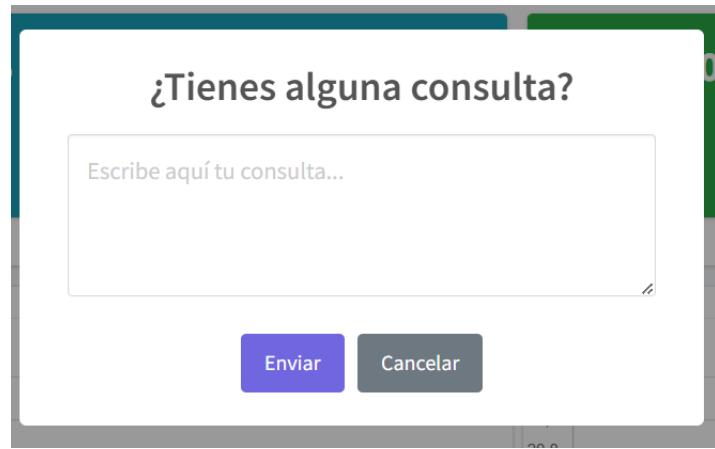


Fig. 96: Vista del modal para preguntar a la I.A. sobre las ventas

En la figura 97, se muestra la respuesta de la I.A. con respecto a la pregunta realizada en la figura anterior y la data del dashboard.

A screenshot of a modal window titled "Respuesta de Gemini IA". It displays the following data:

Registro de compras y ventas	
Compras mensuales	
Mes:	December
Total mensual:	160,00
Ventas mensuales	
Mes:	December
Total mensual:	20,00
Ventas diarias	
Día:	20/12/2024
Total diario:	20,00
Totales del mes actual	
Total compras:	0,00
Total ventas:	0,00
Productos más vendidos	

At the bottom are two buttons: "Volver a preguntar" (Ask again) in a purple box and "Salir" (Exit) in a grey box.

Fig. 97: Vista de la respuesta de la I.A. sobre las ventas

Implementación de la aplicación móvil

Se crea un proyecto desde cero en Android Studio y en la siguiente figura se muestra la organización del código siguiendo el patrón de diseño MVVM.

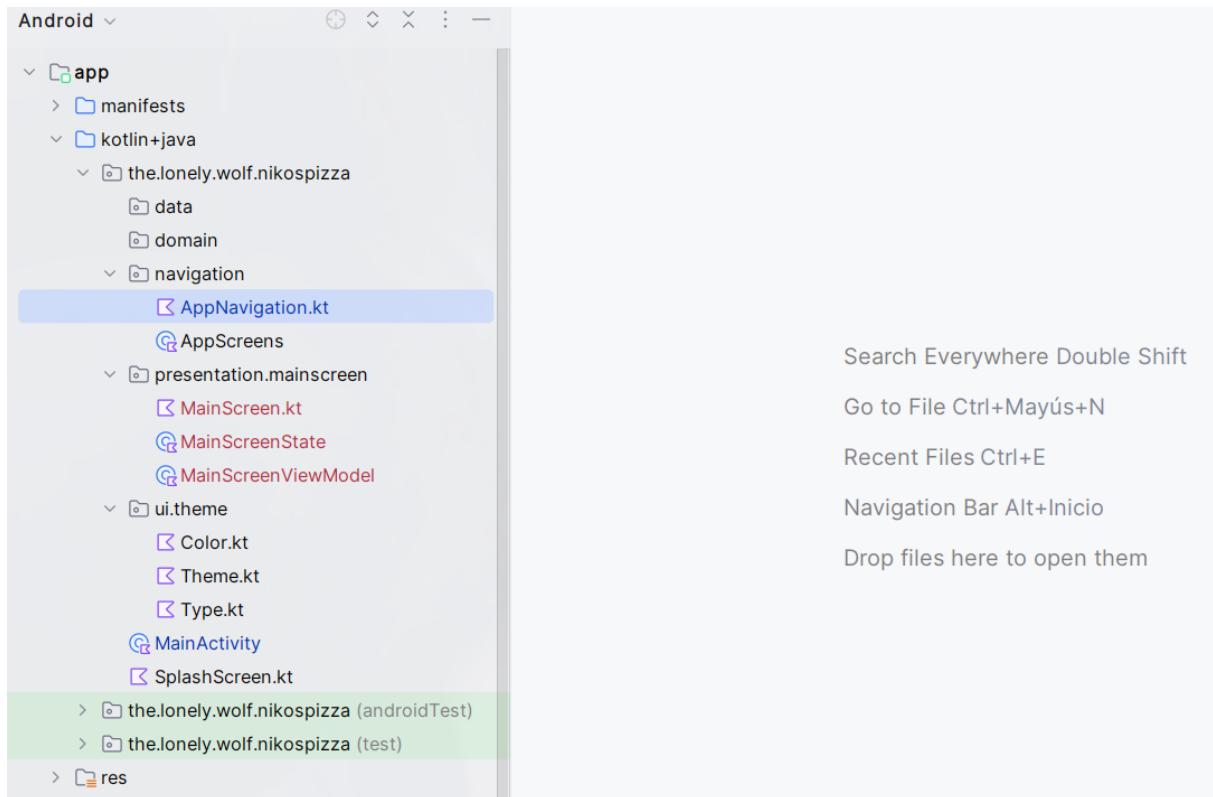
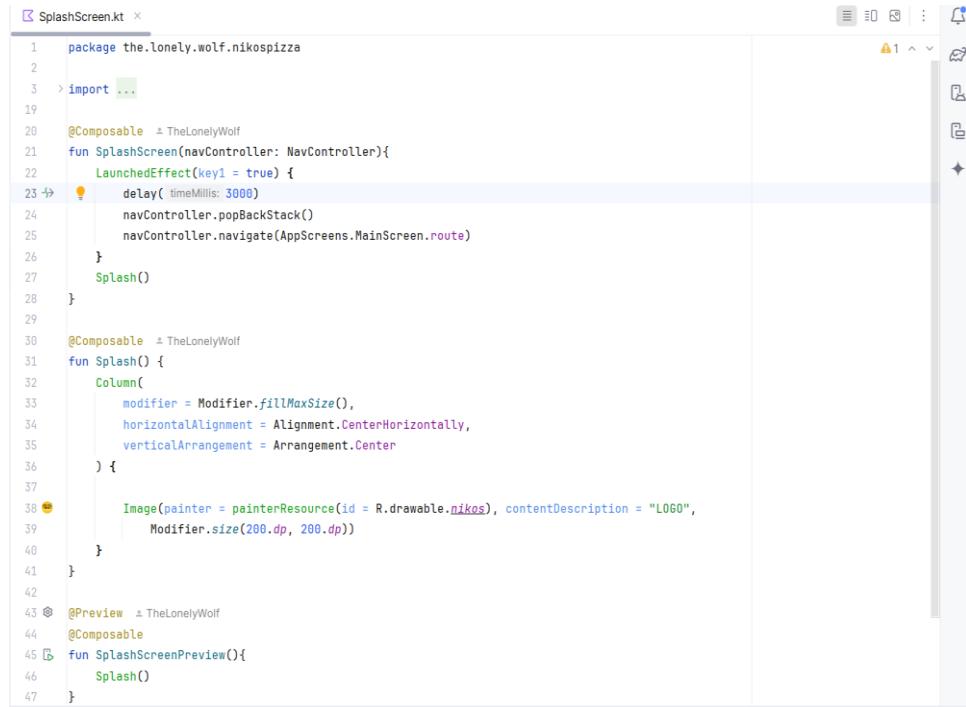


Fig. 98: Estructura inicial del proyecto en Android

Luego se crea el splashscreen para que cuando el usuario ingresa a la aplicación y mientras el webview carga, pueda ver una pantalla de loading.



```
1 package the.lonely.wolf.nikospizza
2
3 > import ...
19
20 @Composable  :- TheLonelyWolf
21 fun SplashScreen(navController: NavController){
22     LaunchedEffect(key1 = true) {
23         delay( timeMillis: 3000)
24         navController.popBackStack()
25         navController.navigate(AppScreens.MainScreen.route)
26     }
27     Splash()
28 }
29
30 @Composable  :- TheLonelyWolf
31 fun Splash() {
32     Column(
33         modifier = Modifier.fillMaxSize(),
34         horizontalAlignment = Alignment.CenterHorizontally,
35         verticalArrangement = Arrangement.Center
36     ) {
37
38         Image(painter = painterResource(id = R.drawable.nikos), contentDescription = "LOGO",
39               Modifier.size(200.dp, 200.dp))
40     }
41 }
42
43 @Preview  :- TheLonelyWolf
44 @Composable
45 fun SplashScreenPreview(){
46     Splash()
47 }
```

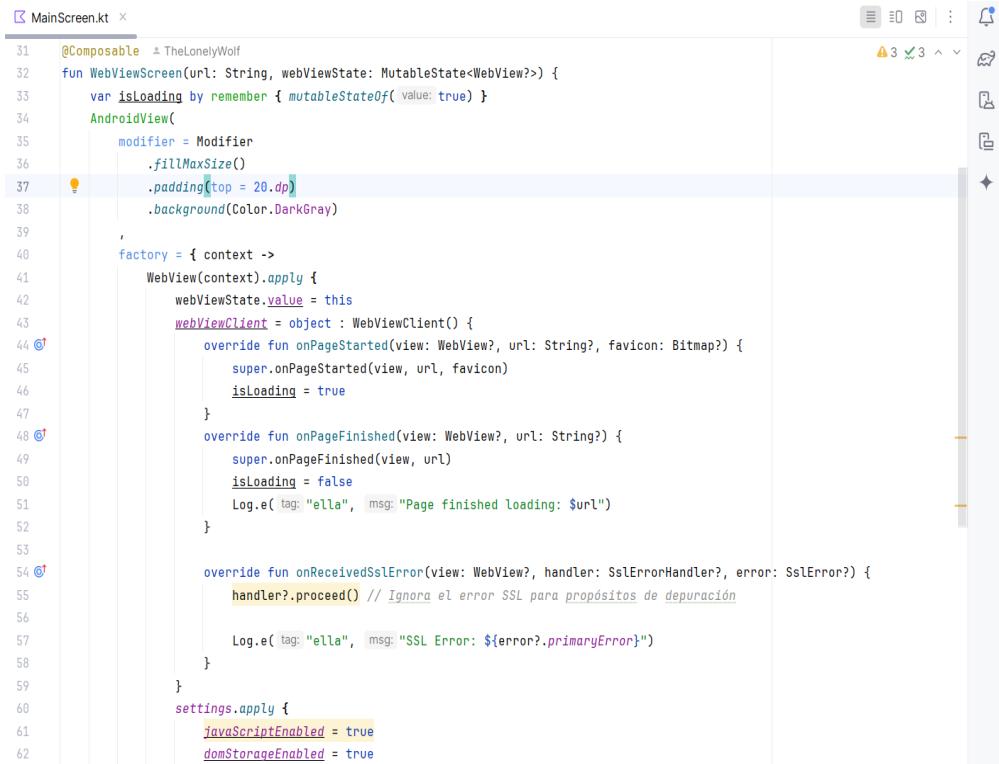
Fig. 99: Código de splashscreen

En la figura 100, se muestra la vista final de como quedó el splashscreen luego de su programación.



Fig. 100: Vista final de splashscreen

En la figura 101, se muestra el código para la implementación del webview.



```

31  @Composable fun WebViewScreen(url: String, webViewState: MutableState<WebView?>) {
32      var isLoading by remember { mutableStateOf( value: true) }
33      AndroidView(
34          modifier = Modifier
35              .fillMaxSize()
36              .padding(top = 20.dp)
37              .background(Color.DarkGray)
38      )
39
40      factory = { context ->
41          WebView(context).apply {
42              webViewState.value = this
43              webViewClient = object : WebViewClient() {
44                  override fun onPageStarted(view: WebView?, url: String?, favicon: Bitmap?) {
45                      super.onPageStarted(view, url, favicon)
46                      isLoading = true
47                  }
48                  override fun onPageFinished(view: WebView?, url: String?) {
49                      super.onPageFinished(view, url)
50                      isLoading = false
51                      Log.e( tag: "ella", msg: "Page finished loading: $url")
52                  }
53
54                  override fun onReceivedSslError(view: WebView?, handler: SslErrorHandler?, error: SslError?) {
55                      handler?.proceed() // Ignora el error SSL para propósitos de depuración
56
57                      Log.e( tag: "ella", msg: "SSL Error: ${error?.primaryError}")
58                  }
59              }
60              settings.apply {
61                  javaScriptEnabled = true
62                  domStorageEnabled = true
63              }
64          }
65      }
66  }

```

Fig. 101: Código de implementación del webview

En la figura 102, se muestra la pantalla de inicio de sesión de la aplicación, esto aparece luego de que el webview carga y el splashscreen ha desaparecido.



Fig. 102: Vista de inicio de sesión en la aplicación móvil

En la figura 103, se muestra la segunda parte de la vista del dashboard, cabe mencionar que el diseño esta para que en caso de que la pantalla sea más extensa se pueda hacer scroll vertical.



Fig. 103: Vista de los productos más vendidos en móvil

En la figura 104, se muestra todas las opciones que tiene la aplicación móvil que son todas las que tiene la aplicación web tal como se solicitó en los requerimientos.

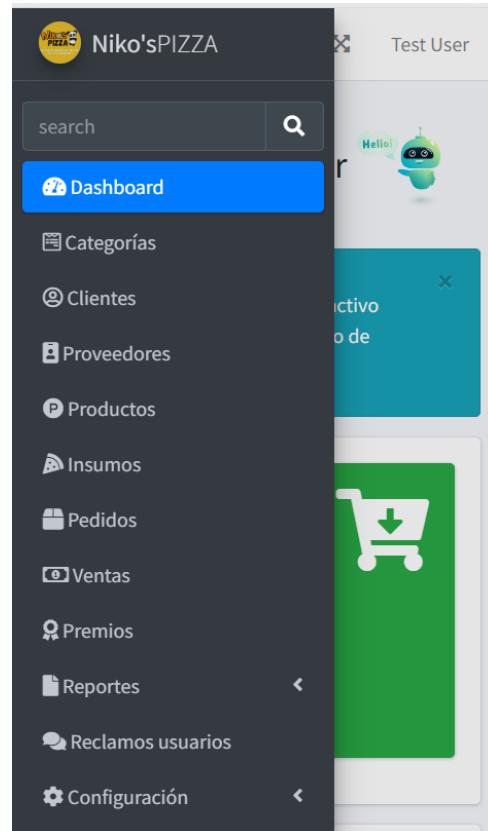


Fig. 104: Vista de todas las opciones en la aplicación móvil

3.1.5 PRUEBAS

En la etapa de pruebas se realiza las pruebas de aceptación del usuario, donde los usuarios reales prueban la aplicación web y móvil y se aseguran de que cumpla con los requisitos. A continuación, se muestra la aplicación con información registrada y las pruebas necesarias para determinar el cumplimiento de los requerimientos, asimismo en el **anexo 15** se muestra el informe por parte de los usuarios que probaron la aplicación.

En la figura 105, se muestra el dashboard con data registrada de una venta.

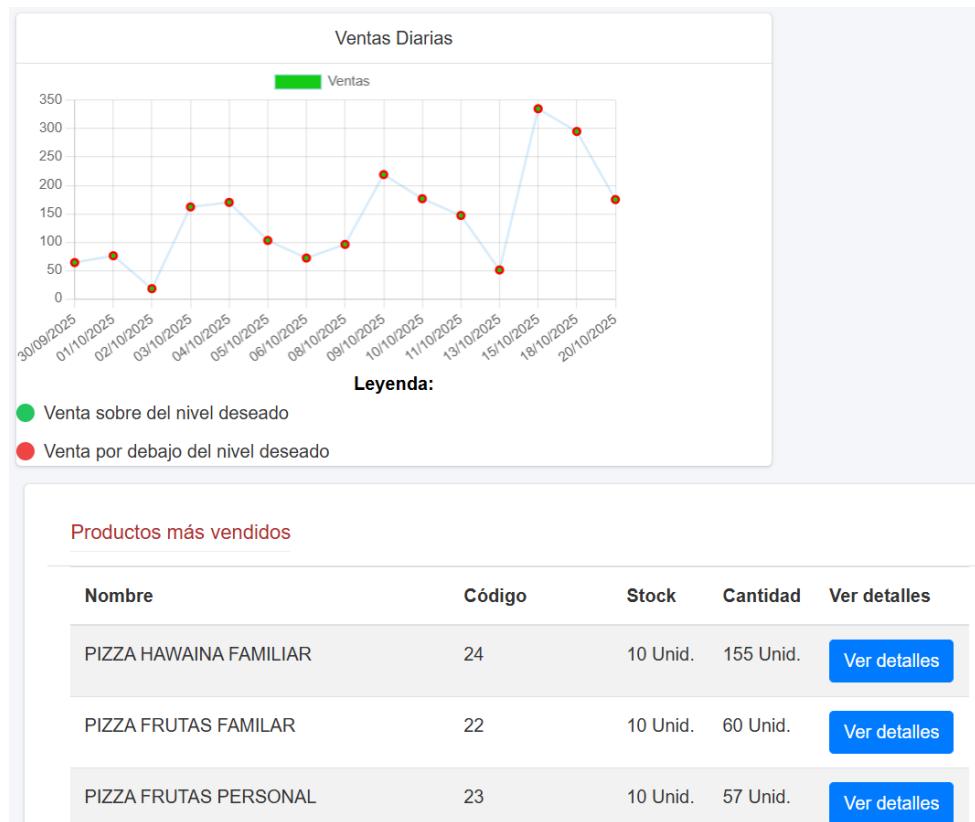


Fig. 105: Prueba de la vista del dashboard

En la figura 106, se muestra las categorías y sus respectivas descripciones, esta data fue registrada por el administrador de la empresa.

ID	Nombre	Descripción	Acciones
1	Pizzas Artesanales	Las pizzas artesanales se preparan a leña con insumos de alta calidad y su eslogan las distingue de nuestros competidores. "SU SABOR SE SIENTE EN CADA MORDIDA".	
2	Bebidas	Las bebidas se elaboran con frutas orgánicas y biológicas de alta calidad para brindarles un sabor distintivo y agradable al paladar al consumidor. Se utiliza maíz INIA 601 para preparar la rica chicha morada	
3	Hamburguesa	Hamburguesas artesanales, preparados por manos expertas y con el cariño y amor profundo que nos caracteriza para atenderlos a los clientes con la amabilidad que usted se merece	

Fig. 106: Prueba de la vista de categorías

En la figura 107, se muestra los productos de la empresa, se puede apreciar los nombres correspondientes, sus categorías y su estado de cada uno de ellos.

ID	Nombre	Stock	Estado	Categoría	Acciones
3	PIZZA AFRICANA FAMILIAR	10	Activo	Pizzas Artesanales	
4	PIZZA AFRICANA PERSONAL	10	Activo	Pizzas Artesanales	
5	PIZZA ALEMANA FAMILIAR	10	Activo	Pizzas Artesanales	
6	PIZZA ALEMANA PERSONAL	10	Activo	Pizzas Artesanales	
7	PIZZA AMBURPIZZA FAMILIAR	10	Activo	Pizzas Artesanales	
8	PIZZA AMBURPIZZA PERSONAL	10	Activo	Pizzas Artesanales	

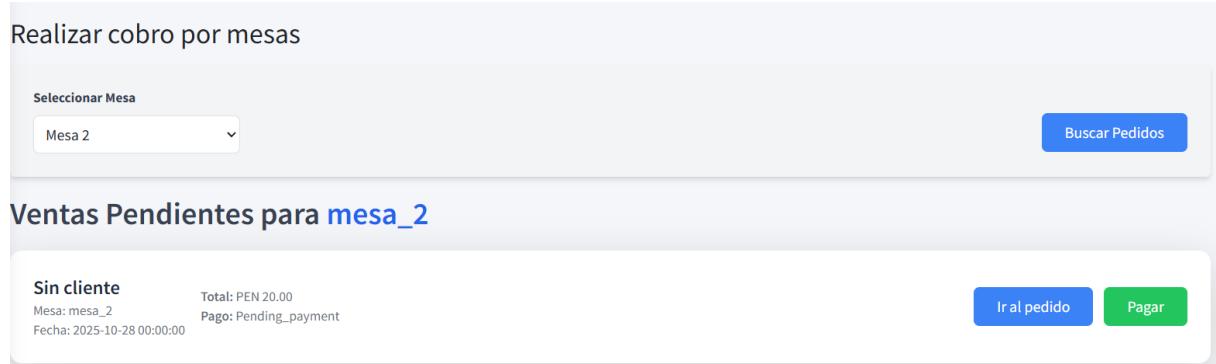
Fig. 107: Prueba de la vista de productos

Asimismo, en la figura 108 se puede apreciar la lista de ventas realizadas en donde se aprecia la fecha, el total y el estado de cada venta.

ID	Fecha	Total	Estado	Acciones
7	2024-09-16 00:41:46	17.00	Activo	
6	2024-09-16 00:40:31	29.00	Activo	
4	2024-09-16 00:32:36	29.00	Activo	
5	2024-09-16 00:32:36	29.00	Activo	
3	2024-09-16 00:31:47	34.00	Activo	
2	2024-09-16 00:29:20	34.00	Activo	

Fig. 108: Prueba de la vista de ventas

En la figura 109, se muestra la lista de ventas pendientes de pago cuando buscamos por mesa, con la opción de ver el detalle de esa venta y a la vez cambiar su estado.



The screenshot shows a user interface for managing table payments. At the top, a header reads "Realizar cobro por mesas". Below it, a search bar is labeled "Seleccionar Mesa" with a dropdown menu showing "Mesa 2" and a "Buscar Pedidos" button. The main content area is titled "Ventas Pendientes para mesa_2". It displays a single entry for a payment: "Sin cliente", "Mesa: mesa_2", "Total: PEN 20.00", "Pago: Pending_payment", and "Fecha: 2025-10-28 00:00:00". To the right of this entry are two buttons: "Ir al pedido" (in blue) and "Pagar" (in green).

Fig. 109: Prueba de búsqueda por mesa para cobrar en ventas

3.1.6 DESPLIGUE

Diagrama de despliegue

En la figura 110, se muestra el diagrama de despliegue de la aplicación web y móvil.

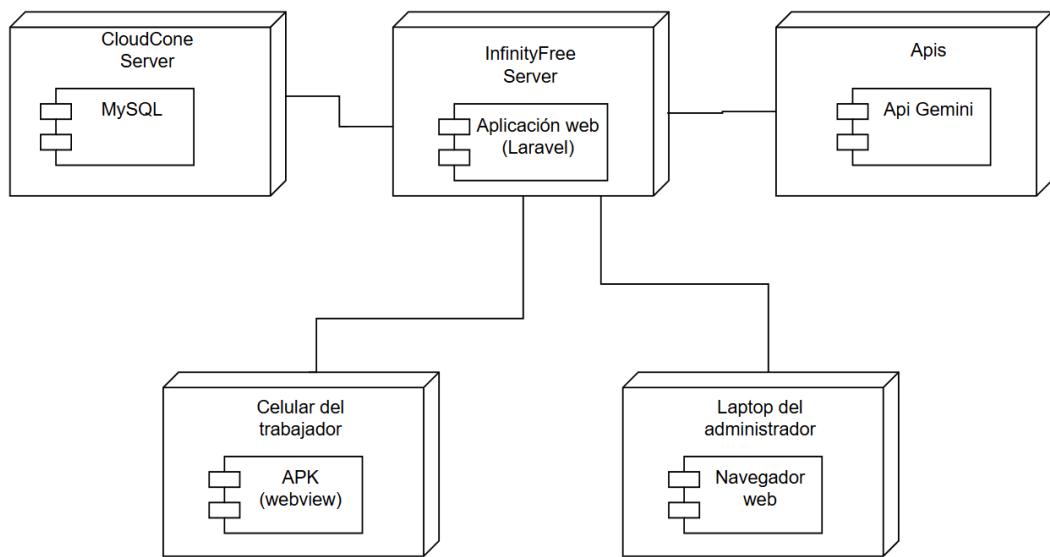


Fig. 110: Diagrama de despliegue

Luego de crear una cuenta en el hosting, se crea una cuenta en GitHub, en este caso no se realizó ello debido a que ya se contaba con una cuenta. Para poder desplegar con GitHub Actions y a través de FTP, primero vamos a “secrets and variables” e ingresamos las credenciales de acceso de FTP del hosting.

Fig. 111: Configuración de variables secretas.

En la figura 112, se muestra el resultado luego de realizar push hacia una rama y vemos como GitHub realizó el workflow correspondiente.

Fig. 112: Resultados de GitHub Actions

Despliegue de la aplicación móvil

Para desplegar una aplicación móvil en la PlayStore, primero se tiene que firmar la aplicación y generar el archivo a subir a la PlayConsole. En la figura 113, se muestra el camino para generar una aplicación firmada, en la parte de “Build” se seleccionar la opción “Generate Signed App Bundle / APK...”.

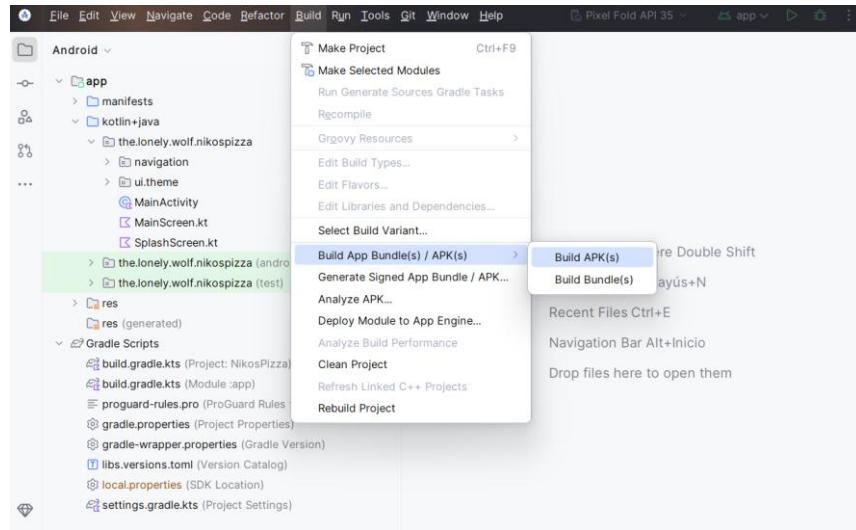


Fig. 113: Vista para generar apk firmado

Luego de presionar en esa opción se tiene que seleccionar el tipo de app a generar, hoy en día Google requiere que se genere una app bundle como se muestra en la figura 114.

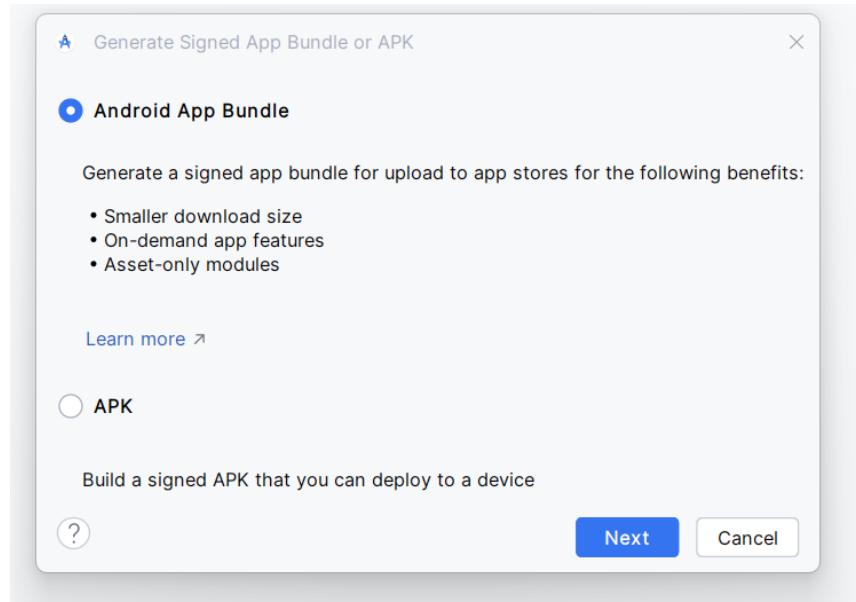


Fig. 114: Vista para seleccionar tipo App Bundle

Luego de ello se tiene que generar una llave para firma la app, a continuación, se muestra los campos necesarios para genera la llave para la aplicación NikosPizza.

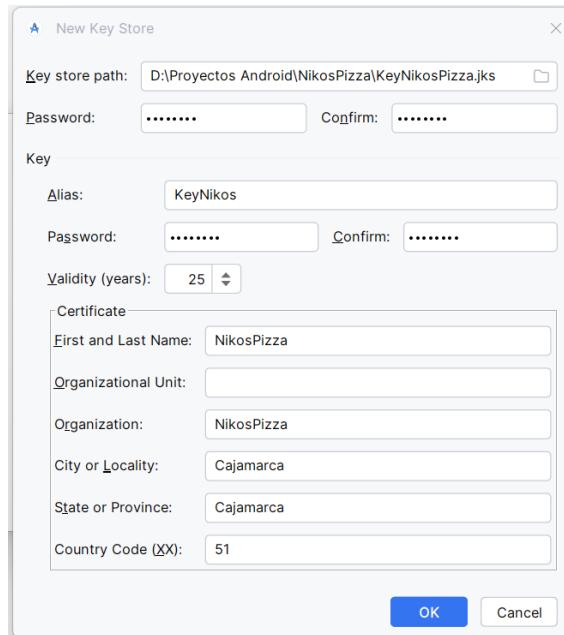


Fig. 115: Vista de generación de Key Signed

Luego de presionar en OK, en la siguiente ventana se debe seleccionar release y con ello se procede a crearse el .aab para subir a la PlayConsole.

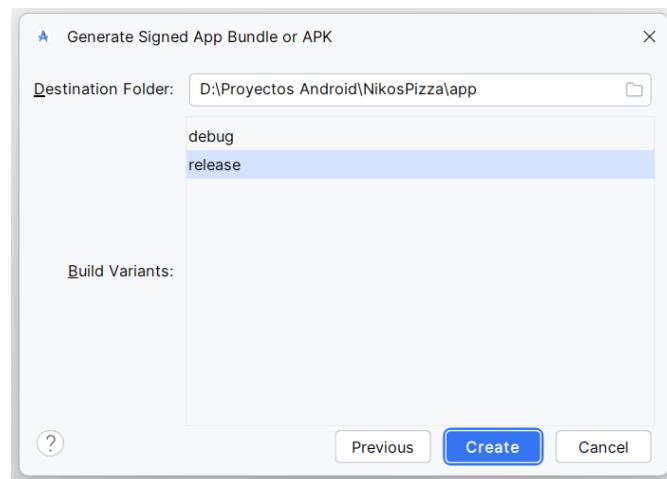


Fig. 116: Vista final para seleccionar tipo de app a generar

Luego de haber creado la app y estar firmada, en la cuenta de PlayConsole se crea la app, para ello se registra el nombre de la aplicación, en este caso se registró el nombre de NikosPizza, se seleccionó el idioma español como predeterminado y se seleccionó el tipo de aplicación como aplicación, a la vez se informó que será una aplicación gratuita como se muestra en la siguiente figura.

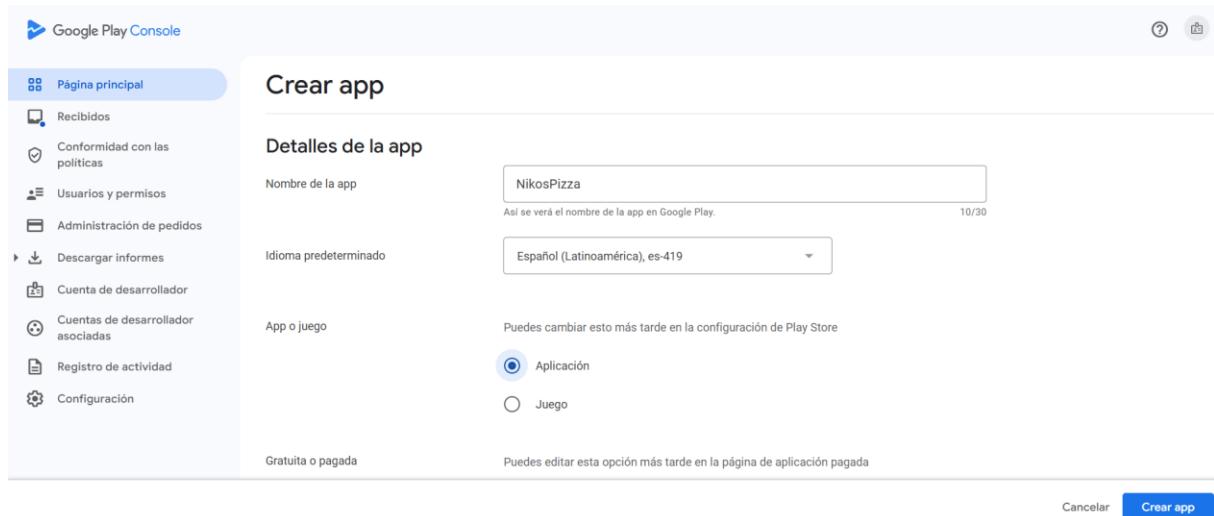


Fig. 117: Visión de como subir app a la PlayConsole

En la figura 118, se muestra el despliegue de la app móvil NikosPizza en Prueba Interna, luego de su aprobación se cambiará el lanzamiento para Producción y con ello concluye el despliegue de la aplicación móvil.

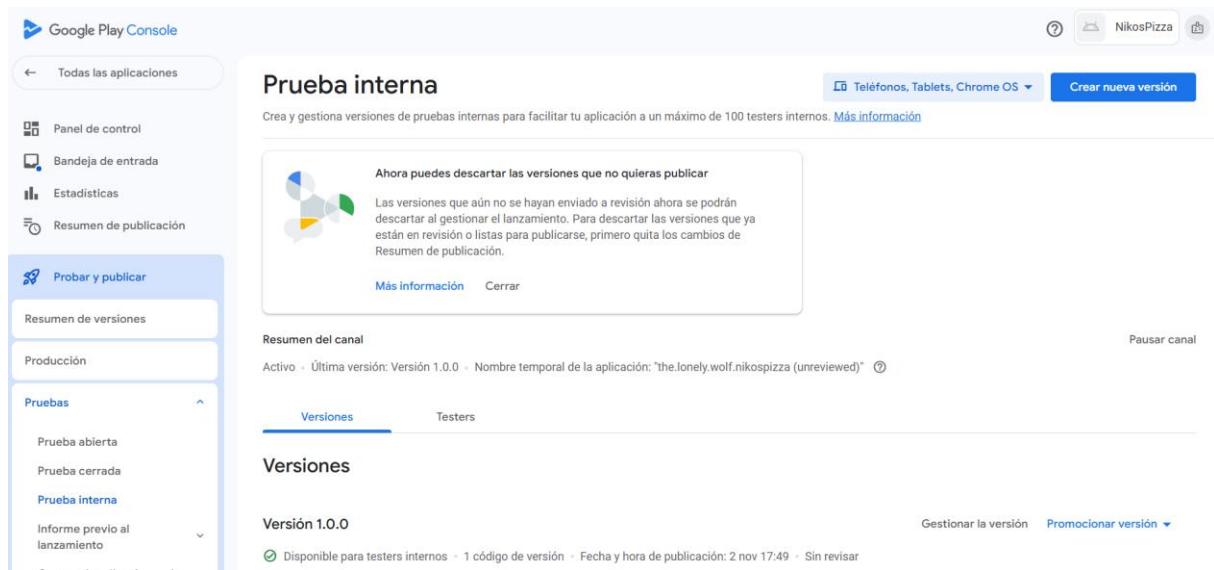


Fig. 118: Vista de despliegue de la app en prueba interna

3.1.7 MANTENIMIENTO

En la parte de mantenimiento es la etapa final del modelo en cascada, en esta se realiza la corrección a los errores presentados y/o actualizaciones necesarias que la aplicación web y móvil requiera frente a los cambios en los requisitos del negocio o en el entorno tecnológico. Cabe destacar que, por el tiempo de duración de la investigación y el alcance definido, los procedimientos, ajustes y modificaciones durante el mantenimiento no se detallan ni visualizan

en este apartado. Esto no altera la funcionalidad central de la aplicación web y móvil establecida.

3.2 TRATAMIENTO, ANÁLISIS DE DATOS Y PRESENTACIÓN DE RESULTADOS

3.2.1 TRATAMIENTO

Luego de aplicar las técnicas de recolección mediante los instrumentos diseñados (cuestionario y ficha de observación), se procedió a ingresar toda la información en una hoja de cálculo de Excel, codificando cada ítem de la escala Likert del 1(muy en desacuerdo / muy malo) al 5 (muy de acuerdo /muy bueno) y registrando los tiempos de cada proceso en minutos con dos decimales. El cuestionario de calidad del software fue respondido por un grupo expertos de hasta tres evaluadores, por lo que sus resultados se analizaron solo con estadísticos descriptivos, priorizando la mediana como medida de tendencia central debido al reducido tamaño muestral. Cabe mencionar que se aplicó el coeficiente alfa de Cronbach para comprobar su consistencia interna del instrumento.

Por otro lado, el cuestionario de satisfacción, aplicado a los cinco trabajadores de la empresa, fue evaluado tanto con estadísticos descriptivos (media, desviación estándar, frecuencias) como con el alfa de Cronbach, con el fin de verificar la fiabilidad del instrumento y analizar los niveles de satisfacción percibida.

En cuanto a las fichas de observación, los tiempos de cada proceso se midieron en 20 ciclos para el registro de pedidos, 20 ciclos para el registro de pagos y en 8 ciclos para la elaboración de reportes de ventas, esto siguiendo la cantidad de ciclos de acuerdo a la duración del ciclo (ver anexo 11), 5 trabajadores fueron evaluados y por cada uno se calculó el tiempo total en cada fase (pretest y posttest). Para determinar la normalidad de los datos se aplicó la prueba de Shapiro-Wilk, de acuerdo con los resultados obtenidos se aplicó la prueba de t de Student para las muestras relacionadas (datos normales) o la prueba de Wilcoxon para los datos no normales. Luego se contrastó las hipótesis, con un nivel de significancia de 0.05.

3.2.2 ANÁLISIS DE DATOS

Los resultados se presentan mediante tablas y gráficos que resumen los valores individuales, medianas, rangos y niveles de significancia estadística, con una interpretación práctica orientada a valorar la influencia del sistema sobre el proceso de ventas.

En este primer apartado se describe las dimensiones de la variable dependiente: Proceso de ventas.

3.2.2.1 DIMENSIÓN TIEMPO

La evaluación del tiempo se realizó mediante fichas de observación, analizando tres indicadores claves: el tiempo de registro de un pedido de un cliente, el tiempo de registro de un pago de un pedido y el tiempo de elaboración de un reporte de ventas. Para el primer indicador, se tomaron 20 mediciones, registrando un promedio de 4 minutos y 18 segundos en la fase antes (pretest). En el segundo caso igualmente se registraron 20 mediciones, registrando un promedio de 3 minutos y 4 segundos. En el caso de los reportes de ventas, se efectuaron 8 mediciones debido a su mayor duración promedio (17 minutos y 15 segundos en el pretest), siguiendo los ciclos establecidos por el método General Electric (Anexo 11).

Para medir el tiempo de registro de pedidos, durante la observación se utilizó un cronómetro digital. El proceso iniciaba cuando el mozo era llamado por el cliente en la mesa y finalizaba en el momento exacto en que el mozo finalizaba de registrar el pedido en el cuaderno de pedidos, en el caso de un delivery se registró el tiempo desde que llegaba la llamada y también finalizaba al terminar de registrar el pedido en el cuaderno. Asimismo, para el tiempo de registro de un pago, inicia cuando el cliente pide su cuenta y finaliza cuando en el cuaderno se registra el pago y el método de pago. Para el tiempo de elaboración de reporte de ventas, la medición iniciaba cuando el administrador comenzaba a recopilar manualmente los datos (revisar cada hoja del cuaderno donde se registraron los pedidos) y culminaba cuando después de generar en una hoja de cálculo el reporte este era guardado en una carpeta en la computadora del administrador. El instrumento utilizado para ambos indicadores fue una ficha de observación estandarizada (ver anexo 4), la cual fue aplicada únicamente por el investigador para así mantener la consistencia en la medición. En la figura 5, 7 y figura 8 se muestra el proceso para el registro de un pedido, la generación de reportes y el registro de un pago de un pedido respectivamente. Cabe destacar que para cada indicador se utilizó una ficha de observación independiente. En la tabla X se muestra el tiempo promedio y el número de observaciones de cada indicador en la fase pretest.

Tabla X: Tiempo promedio y número de observaciones en la fase de pretest para la dimensión tiempo

Indicador	Tiempo promedio pretest (min)	Número de observaciones
Tiempo de registro de un pedido del cliente	04:18	20
Tiempo de elaboración de un reporte de ventas	17:15	8
Tiempo de registro de un pago de un pedido	03:04	20

Luego de implementar la aplicación web y móvil el proceso de registro de pedido cambia ligeramente, en la figura 119 se muestra el proceso modelado en Bizagi luego de la implementación.

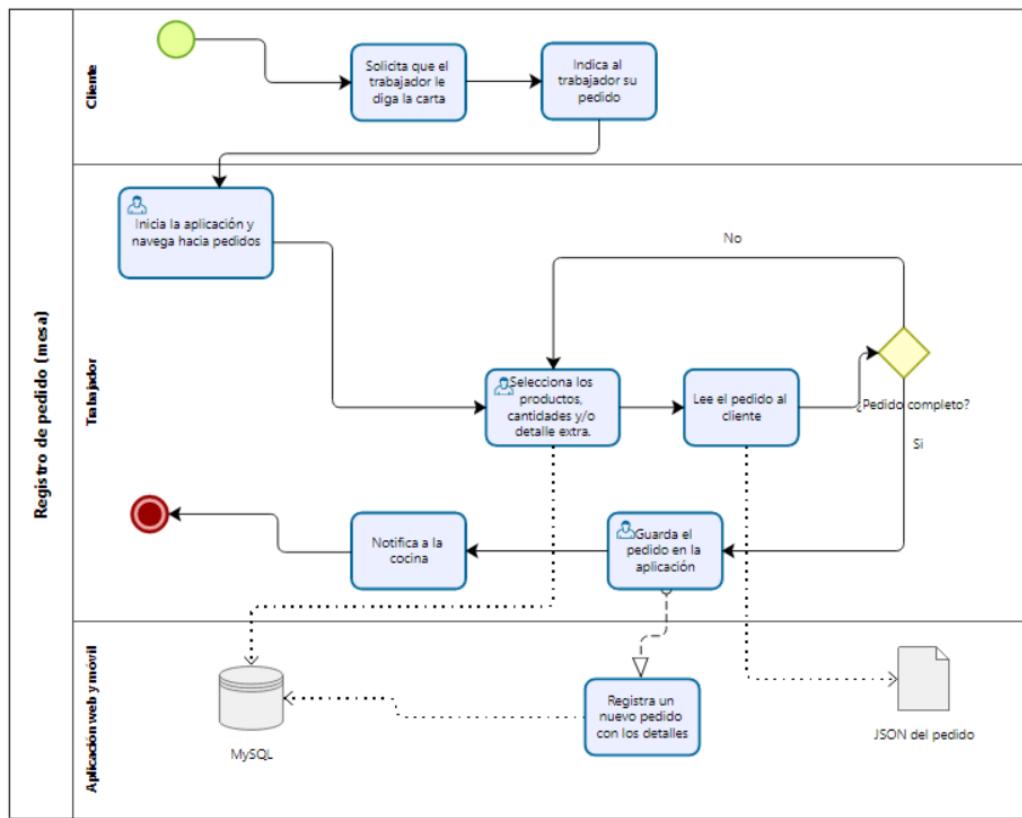


Fig. 119: Proceso modelado de registro de pedido (postest).

En la figura 120, se muestra el proceso de registro de un pedido a través de delivery modelado luego de la implementación.

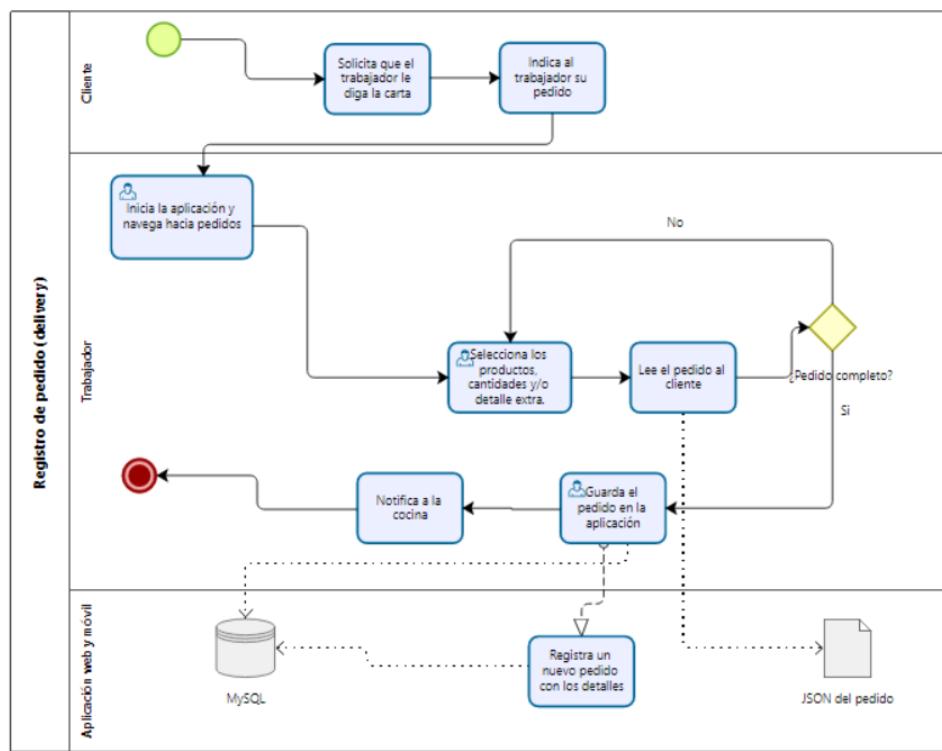


Fig. 120: Diagrama del proceso de registro de pedido por delivery (posttest)

Asimismo, luego de la implementación de la aplicación web y móvil el proceso de elaboración de reporte de ventas cambia ligeramente, en la figura 121 se muestra el proceso modelado en Bizagi luego de la implementación.

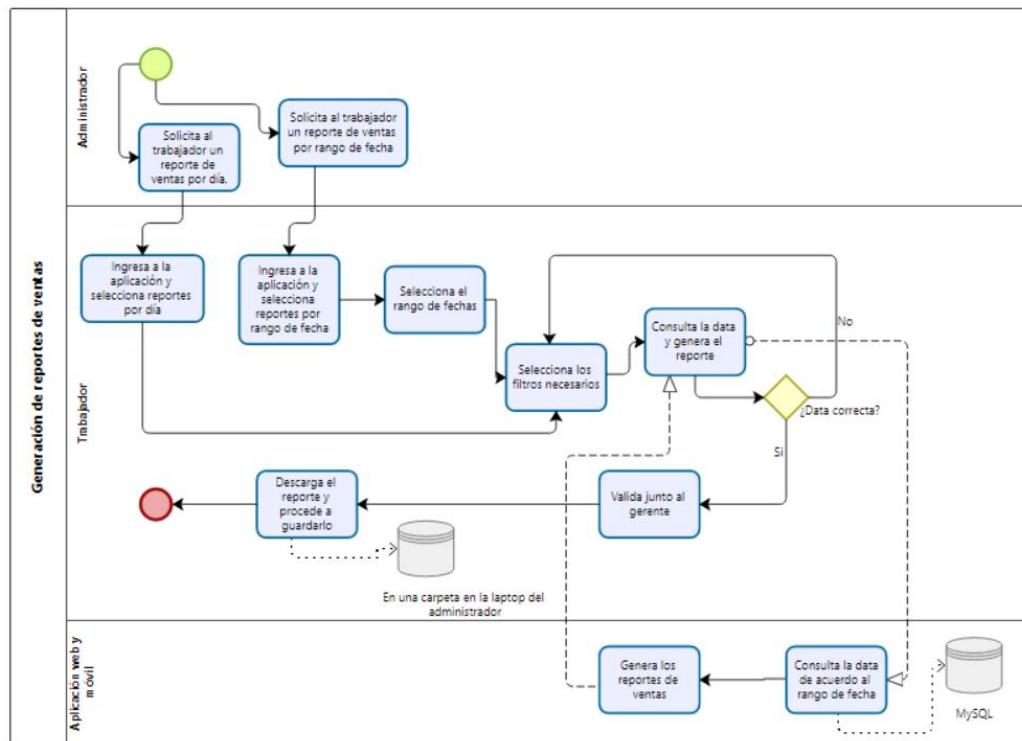


Fig. 121: Proceso modelado de elaboración de reportes de ventas (posttest).

En la figura 122, se muestra el proceso de registro de pagos luego de implementar la aplicación web y móvil.

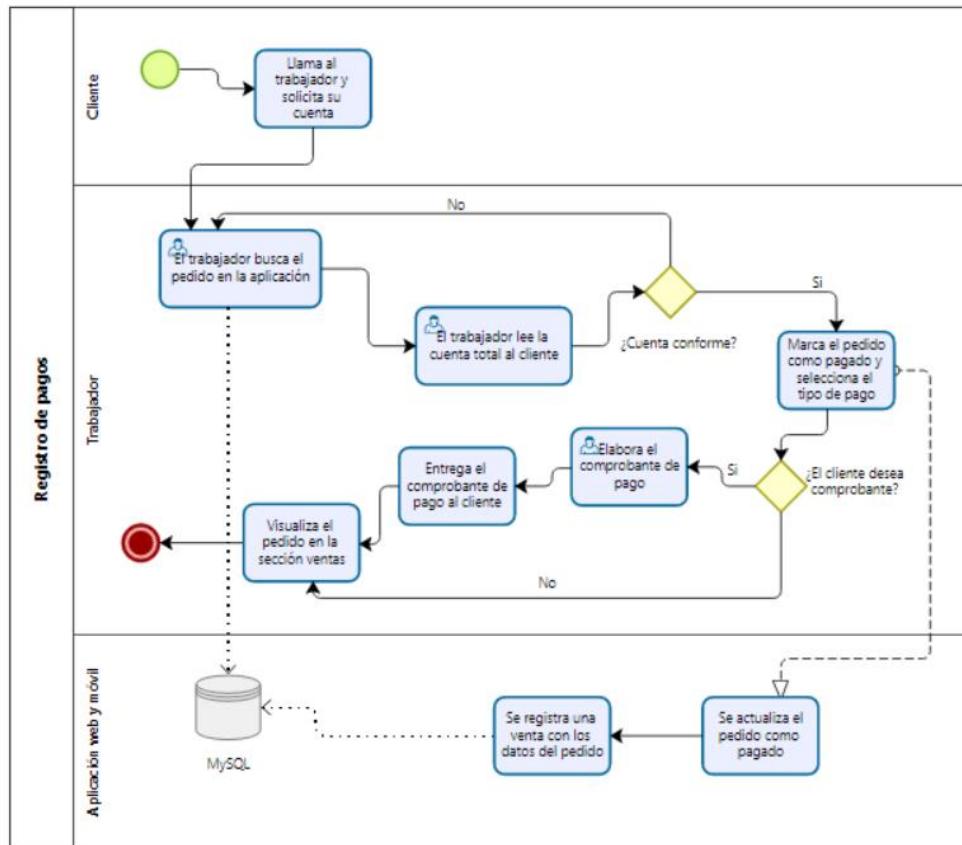


Fig. 122: Modelado del proceso de registro de pagos (postest)

Para medir el tiempo luego de la implementación de la aplicación web y móvil también se utilizó un cronómetro digital y fue aplicado únicamente por el investigador. En la siguiente tabla se muestra un resumen de los tiempos por cada indicador en la fase postest.

Tabla XI: Tiempo promedio y número de observaciones en la fase de postest para la dimensión tiempo

Indicador	Tiempo promedio postest (min)	Número de observaciones
Tiempo de registro de un pedido del cliente	01:19	20
Tiempo de elaboración de un reporte de ventas	02:31	8
Tiempo de registro de un pago de un pedido	02:20	20

Para contrastar la hipótesis específica (la implementación de una aplicación web y móvil reduce el tiempo de registros de los pedidos y la elaboración de reporte de ventas en Nikos Pizza) se procedió a determinar la normalidad de los datos, y para ello se aplicaron las pruebas de

Shapiro-Wilk y Kolmogorov-Smirnov (Anexo 12). Los resultados de estas pruebas permitieron verificar el tipo de distribución de los tiempos registrados, un paso fundamental para seleccionar los métodos estadísticos adecuados en el análisis posterior.

A continuación, en la tabla XII, se detallan los tiempos registrados en cada ciclo de observación para el proceso de registro de pedidos, comparando las fases pretest y postest:

Tabla XII: Detalle del tiempo que tarda registrar un pedido en cada ciclo de observación.

Tiempo de registro de un pedido		
Observaciones y/o ciclos (C)	Tiempos(m)	
	Pretest	Postest
C1	04:05	01:13
C2	04:48	01:47
C3	03:53	01:03
C4	04:21	01:21
C5	04:03	01:22
C6	05:35	01:23
C7	04:43	01:16
C8	04:03	01:06
C9	05:12	01:05
C10	03:48	01:34
C11	04:36	01:30
C12	03:48	01:38
C13	03:23	01:15
C14	03:13	01:31
C15	04:28	01:17
C16	04:33	01:20
C17	04:44	00:53
C18	03:55	01:01
C19	04:41	01:18
C20	04:15	01:31

En la tabla XIII se muestra los tiempos registrados en cada ciclo de observación para la elaboración de un reporte de ventas, haciendo una comparación en cada fase.

Tabla XIII: Detalle del tiempo que tarda para elaborar un reporte de ventas en cada ciclo.

Tiempo de elaboración de un reporte de ventas		
Observaciones y/o ciclos (C)	Tiempos(m)	
	Pretest	Postest
C1	15:50	02:27
C2	17:40	02:28
C3	16:30	02:27
C4	17:20	02:26
C5	18:20	02:42
C6	18:30	02:30
C7	17:30	02:33
C8	16:20	02:38

En la tabla XIV se muestra los tiempos registrados para el antes y el después del registro de cada pago.

Tabla XIV: Detalle del tiempo que tarda en registrar un pago de un pedido en cada ciclo

Tiempo de registro de un pago de un pedido		
Observaciones y/o ciclos (C)	Tiempos(m)	
	Pretest	Postest
C1	03:06	02:24
C2	03:02	02:13
C3	03:05	02:25
C4	03:03	02:37
C5	02:52	02:08
C6	03:23	02:26
C7	03:07	02:25
C8	02:23	02:23
C9	02:52	02:08
C10	03:13	02:04
C11	02:58	02:23
C12	02:43	02:31
C13	03:06	02:12
C14	03:13	02:14
C15	03:06	02:13
C16	02:40	02:12
C17	03:12	02:16
C18	03:11	02:16
C19	03:18	02:37
C20	03:09	02:35

Prueba de hipótesis

- Para evaluar la influencia de la implementación de la aplicación web y móvil en el tiempo de registro de los pedidos, tiempo de registro de los pagos y la elaboración de reporte de ventas en Nikos Pizza, se aplicó la prueba T-Student de muestras emparejadas, comparando los tiempos registrados en las fases pretest (antes de la implementación, AI) y postest (después de la implementación, DI). Esta prueba permitió determinar la aceptación o rechazo de las hipótesis planteadas, utilizando un nivel de significancia del 5% ($\alpha = 0.05$) y un nivel de confianza del 95%.

Hipótesis nula (H_0):

El tiempo de registro de un pedido, de un pago y de la elaboración de un reporte de ventas es menor al tiempo después de implementar la aplicación web y móvil.

$$H_0: AI \leq DI$$

Hipótesis alternativa (H_1):

El tiempo de registro de un pedido, de un pago y de la elaboración de un reporte de ventas es mayor al tiempo después de implementar la aplicación web y móvil.

$$H_1: AI > DI$$

Nota:

H_0 : Hipótesis nula

H_1 : Hipótesis alternativa

AI: Antes de la implementación

DI: Despues de la implementación

Nivel de significancia

Para el análisis de hipótesis, se estableció un nivel de significancia del 5%, lo que implica un 95% de confianza en los resultados. Este umbral determina la probabilidad máxima aceptable para cometer un error Tipo I (rechazar incorrectamente la hipótesis nula cuando es verdadera).

Valor estadístico

Como se evidencia en las siguientes tablas, la aplicación de la prueba T-Student para muestras emparejadas permitió cuantificar el grado de significancia de las diferencias observadas entre las fases pretest y postest. Estos valores nos ayudan a respaldar la validez de las conclusiones.

Tabla XV: Estadísticas de muestras emparejadas para la dimensión tiempo

Estadística de muestras emparejadas					
		Media	N	Desviación estándar	Media de error estándar
P1	Tiempo de registro de un pedido (pretest)	4.3055	20	0.58024	0.12975
	Tiempo de registro de un pedido (postest)	1.32	20	0.22613	0.0557
P2	Tiempo de elaboración de un reporte de ventas (pretest)	17.2488	8	0.95614	0.33805
	Tiempo de elaboración de un reporte de ventas (postest)	2.52	8	0.09721	0.03437
P3	Tiempo de registro de un pago de un pedido (pretest)	3.0600	20	0.18507	0.04138
	Tiempo de registro de un pago de un pedido (postest)	0.3352	20	0.16617	0.03716

Tabla XVI: Pruebas de medias emparejadas para la dimensión tiempo

Prueba de medias emparejadas								
		Diferencias emparejadas						
		Media	Desv. est	95% de intervalo de confianza de la diferencia		t	gl	P de un factor
				Inf.	Sup.			
P1	Tiempo de registro de un pedido (Pre) & Tiempo de registro de un pedido (Post)	2.98550	0.64152	2.68526	3.28574	20.812	19	<0.001
P2	Tiempo de elaboración de un reporte de ventas (Pre) & Tiempo de elaboración de un reporte de ventas (Post)	14.72625	0.93096	13.94795	15.50455	44.741	7	<0.001
P3	Tiempo de registro de pago de un pedido (pre) & Tiempo de registro de pago de un pedido (post)	0.72483	0.22621	0.61896	0.83070	14.330	19	<0.000

Los resultados de la prueba T para muestras emparejadas demostraron una reducción significativa en los tiempos analizados después de la intervención. En el proceso de registro de pedidos (P1), se observó una disminución promedio de 2.99 unidades ($p < 0.001$), con un intervalo de confianza del 95% entre 2.69 y 3.29. De manera similar, el tiempo de elaboración de reportes (P2) presentó una reducción aún mayor, con una media de 14.73 unidades ($p < 0.001$) y un intervalo de confianza entre 13.95 y 15.50. Asimismo, para el tiempo de registro de un pago de un pedido se observa una disminución promedio de 0.72 unidades ($p < 0.000$), con un intervalo de confianza entre 0.61 y 0.83. Dado que en todos los casos el valor de p fue inferior a 0.05, se rechaza la hipótesis nula y se acepta la hipótesis alternativa, concluyendo que los tiempos después de la implementación fueron menores que antes de la implementación de

la aplicación web y móvil. Estos hallazgos respaldan la influencia y la reducción en los tiempos luego de implementar la aplicación web y móvil.

3.2.2.2 DIMENSIÓN SATISFACCIÓN

En la evaluación de la satisfacción se aplicó un cuestionario, en ello se analizó los indicadores claves: porcentaje de trabajadores satisfechos con el registro de los pedidos del cliente, porcentaje de trabajadores satisfechos con la reducción de errores en las especificaciones de los pedidos, porcentaje de trabajadores satisfechos con la gestión de reclamos y porcentaje de trabajadores satisfechos con la elaboración de reportes de ventas. Para el primer indicador se elaboraron 7 preguntas, para el segundo indicador se elaboraron 3 preguntas, para el 3 indicador se elaboraron 2 preguntas y para el 4 indicador se elaboraron 5 preguntas. La cantidad de observaciones fue de 5 debido a que se entrevistó a 5 trabajadores de la empresa.

En la tabla XVII se presenta los indicadores y el puntaje promedio obtenido en escala Likert como también el número de personas encuestadas en la fase pretest.

Tabla XVII: Puntaje promedio de cada indicador de la dimensión satisfacción (pretest).

Indicador	Número de personas encuestadas	Puntaje promedio (pretest)
Porcentaje de trabajadores satisfechos con el registro de los pedidos del cliente	5	1.9714
Porcentaje de trabajadores satisfechos con la reducción de errores en las especificaciones de los pedidos	5	3.1333
Porcentaje de trabajadores satisfechos con la gestión de reclamos	5	2.4000
Porcentaje de trabajadores satisfechos con la elaboración de reportes	5	1.6400

En la tabla XVIII se presenta los indicadores y el puntaje promedio obtenido en escala Likert como también el número de personas encuestadas en la fase postest.

Tabla XVIII: Puntaje promedio de cada indicador de la dimensión satisfacción (postest).

Indicador	Número de personas encuestadas	Puntaje promedio (Postest)
Porcentaje de trabajadores satisfechos con el registro de los pedidos del cliente	5	4.1429
Porcentaje de trabajadores satisfechos con la reducción de errores en las especificaciones de los pedidos	5	4.0000
Porcentaje de trabajadores satisfechos con la gestión de reclamos	5	4.1000
Porcentaje de trabajadores satisfechos con la elaboración de reportes	5	4.0400

Nota: Se utilizó la escala Likert (1 = Muy en desacuerdo, 2 = En desacuerdo, 3 = Neutral, 4 = De acuerdo, 5 = Muy de acuerdo)

Para contrastar la hipótesis específica (la implementación de una aplicación web y móvil aumenta la satisfacción de los trabajadores inmersos en el proceso de ventas de Nikos Pizza) se procedió a determinar la normalidad de los datos aplicando las pruebas de Shapiro-Wilk y Kolmogorov-Smirnov (ver anexo 13). Los resultados de estas pruebas permitieron verificar el tipo de distribución de las respuestas del cuestionario en escala de Likert, un paso fundamental para seleccionar los métodos estadísticos adecuados en el análisis posterior. Debido a la cantidad de la muestra muy pequeña ($n = 5$) se optó por pruebas no paramétricas como se detalla más adelante.

Prueba de hipótesis

A su vez, para evaluar la influencia de la implementación de la aplicación web y móvil en la satisfacción de los trabajadores inmersos en el proceso de ventas de Nikos Pizza, se aplicó la prueba no paramétrica de muestras emparejadas, comparando las respuestas registradas en las fases pretest (antes de la implementación, AI) y postest (después de la implementación, DI). Esta prueba permitió determinar la aceptación o rechazo de las hipótesis planteadas, utilizando un nivel de significancia del 5% ($\alpha = 0.05$) y un nivel de confianza del 95%.

Hipótesis nula (H_0):

La implementación de la aplicación web y móvil no mejora la satisfacción de los trabajadores inmersos en el proceso de ventas en la empresa Nikos Pizza.

Hipótesis alternativa (H_1):

La implementación de la aplicación web y móvil mejora la satisfacción de los trabajadores inmersos en el proceso de ventas en la empresa Nikos Pizza.

Nota:

H_0 : Hipótesis nula

H_1 : Hipótesis alternativa

Nivel de significancia

Para el análisis de hipótesis, se estableció un nivel de significancia del 5%, lo que implica un 95% de confianza en los resultados. Este umbral determina la probabilidad máxima aceptable para cometer un error Tipo I (rechazar incorrectamente la hipótesis nula cuando es verdadera).

Valor estadístico

Como se evidencia en las siguientes tablas, la aplicación de la prueba no paramétrica Wilcoxon para muestras emparejadas permitió cuantificar el grado de significancia de las diferencias observadas entre las fases pretest y postest. Estos valores nos ayudan a respaldar la validez de las conclusiones.

Tabla XIX: Prueba no paramétrica para el indicador 1 de la dimensión satisfacción.

Resumen de prueba de rangos con signo de Wilcoxon para muestras relacionadas	
N total	5
Estadístico de prueba	15.000
Error estándar	3.708
Estadístico de prueba estandarizado	2.023
Sig. Asintótica (prueba bilateral)	0.43

Tabla XX: Prueba no paramétrica para el indicador 2 de la dimensión satisfacción.

Resumen de prueba de rangos con signo de Wilcoxon para muestras relacionadas	
N total	5
Estadístico de prueba	15.000
Error estándar	3.708
Estadístico de prueba estandarizado	2.023
Sig. Asintótica (prueba bilateral)	0.43

Tabla XXI: Prueba no paramétrica para el indicador 3 de la dimensión satisfacción.

Resumen de prueba de rangos con signo de Wilcoxon para muestras relacionadas	
N total	5
Estadístico de prueba	15.000
Error estándar	3.640
Estadístico de prueba estandarizado	2.060
Sig. Asintótica (prueba bilateral)	0.39

Tabla XXII: Prueba no paramétrica para el indicador 4 de la dimensión satisfacción.

Resumen de prueba de rangos con signo de Wilcoxon para muestras relacionadas	
N total	5
Estadístico de prueba	15.000
Error estándar	3.640
Estadístico de prueba estandarizado	2.060
Sig. Asintótica (prueba bilateral)	0.39

Los resultados demuestran que la implementación de la aplicación web y móvil en el proceso de ventas de la empresa Nikos Pizza generó mejoras en la satisfacción de los trabajadores, esto se evidencia por aumentos en todos los indicadores evaluados. Si bien estas mejoras no alcanzaron un umbral de significancia estadísticamente ($p > 0.05$), debido al tamaño muestral reducido ($n = 5$), los puntajes promedio postest superaron el pretest, estos hallazgos respaldan la hipótesis de una mejora en la satisfacción de los trabajadores. Cabe mencionar, el tamaño muestral reducido ($n = 5$) sugiere precaución al generalizar los resultados.

3.2.2.3 DIMENSIÓN CALIDAD

Para la evaluación de la calidad se aplicó un cuestionario, en ello se analizó los indicadores claves: nivel de exactitud de los reportes de ventas y nivel de consistencia de los reportes de

ventas. Para el primer indicador se elaboraron 3 preguntas y para el segundo indicador se elaboraron 2 preguntas. La cantidad de observaciones fue de 5 debido a que se entrevistó a 5 trabajadores de la empresa.

En la tabla XXIII se presenta los indicadores y sus valores promedio en escala de Likert y también el número de observaciones en la fase pretest.

Tabla XXIII: Puntaje promedio de cada indicador de la dimensión satisfacción (pretest).

Indicador	Número de personas encuestadas	Puntaje promedio (pretest)
Porcentaje de exactitud de los reportes de ventas	5	1.6667
Porcentaje de consistencia de los reportes de ventas	5	1.9000

En la tabla XXIV se presenta los indicadores y sus valores promedio en escala de Likert junto al número de observaciones en la fase postest.

Tabla XXIV: Puntaje promedio de cada indicador de la dimensión satisfacción (postest).

Indicador	Número de personas encuestadas	Puntaje promedio (Postest)
Porcentaje de exactitud de los reportes de ventas	5	4.0073
Porcentaje de consistencia de los reportes de ventas	5	4.1000

Nota: Se utilizó la escala Likert (1 = Muy en desacuerdo, 2 = En desacuerdo, 3 = Neutral, 4 = De acuerdo, 5 = Muy de acuerdo)

Para contrastar la hipótesis específica (la implementación de una aplicación web y móvil aumenta la calidad de los reportes de ventas en Nikos Pizza) se procedió a determinar la normalidad de los datos, para ello se aplicaron las pruebas de Shapiro-Wilk y Kolmogorov-Smirnov (ver anexo 14). Los resultados de estas pruebas permitieron verificar el tipo de distribución de las respuestas del cuestionario en escala de Likert, para luego poder seleccionar los métodos estadísticos adecuados en el análisis posterior. Cabe resaltar que la muestra ($n = 5$) es muy pequeña y por ello se optó por usar pruebas no paramétricas como se detalla más adelante.

Prueba de hipótesis

Para evaluar la influencia de la implementación de la aplicación web y móvil en la calidad de los reportes de ventas en Nikos Pizza, se aplicó la prueba no paramétrica Wilcoxon de muestras emparejadas, comparando las respuestas en escala Likert en las fases pretest (antes de la implementación, AI) y postest (después de la implementación, DI). Esta prueba permitió determinar la aceptación o rechazo de las hipótesis planteadas, utilizando un nivel de significancia del 5% ($\alpha = 0.05$) y un nivel de confianza del 95%.

Hipótesis nula (H_0):

La implementación de la aplicación web y móvil no mejora la calidad de los reportes de ventas en el proceso de ventas en la empresa Nikos Pizza.

Hipótesis alternativa (H_1):

La implementación de la aplicación web y móvil mejora la calidad de los reportes de ventas en el proceso de ventas en la empresa Nikos Pizza.

Nota:

H_0 : Hipótesis nula

H_1 : Hipótesis alternativa

Nivel de significancia

Para el análisis de la hipótesis, se estableció un nivel de significancia del 5%, lo que implica un 95% de confianza en los resultados. Este umbral determina la probabilidad máxima aceptable para cometer un error Tipo I (rechazar incorrectamente la hipótesis nula cuando es verdadera).

Valor estadístico

Como se evidencia en las siguientes tablas, la aplicación de la prueba no paramétrica para muestras emparejadas permitió cuantificar el grado de significancia de las diferencias observadas entre las fases pretest y postest. Estos valores nos ayudan a respaldar la validez de las conclusiones.

Tabla XXV: Prueba no paramétrica para el indicador 1 de la dimensión calidad.

Resumen de prueba de rangos con signo de Wilcoxon para muestras relacionadas	
N total	5
Estadístico de prueba	15.000
Error estándar	3.708
Estadístico de prueba estandarizado	2.023
Sig. Asintótica (prueba bilateral)	0.43

Tabla XXVI: Prueba no paramétrica para el indicador 2 de la dimensión calidad.

Resumen de prueba de rangos con signo de Wilcoxon para muestras relacionadas	
N total	5
Estadístico de prueba	15.000
Error estándar	3.691
Estadístico de prueba estandarizado	2.032
Sig. Asintótica (prueba bilateral)	0.42

La evaluación de la calidad se realizó mediante un cuestionario Likert aplicado a 5 trabajadores, analizando los indicadores de exactitud y consistencia de los reportes de ventas. Los puntajes promedio en el pretest (1,67 y 1.90) evidenciaron un descontento inicial, mientras que en el postest (4.01 y 4.10) se reflejaron una mejora notable. Para validar estadísticamente estos resultados, se aplicó la prueba de Wilcoxon esto debido a que el tamaño muestral es pequeño ($n = 5$). Los valores de los estadísticos estandarizados (2.023 para exactitud; 2.032 para consistencia) y los p-valores bilaterales (0.43 y 0.42) superaron el nivel de significancia ($\alpha = 0.05$), a pesar de ello los puntajes promedio después de la implementación son mayores por lo tanto sugieren un efecto práctico positivo. Esto confirma que la aplicación web y móvil mejoró la calidad de los reportes en ambos indicadores.

3.2.3 PRESENTACIÓN DE RESULTADOS

A continuación, se muestran los resultados obtenidos tanto para la variable independiente y dependiente.

En la variable independiente, se aplicó un cuestionario a 3 expertos en desarrollo de software de manera anónima, cuyos resultados se muestran más adelante.

3.2.3.1 DIMENSIÓN CALIDAD DEL SOFTWARE

En lo que respecta a calidad del software se contaron con 5 indicadores y para cada indicador se elaboraron preguntas, en la figura 123 se muestra las respuestas de acuerdo a cada indicador.

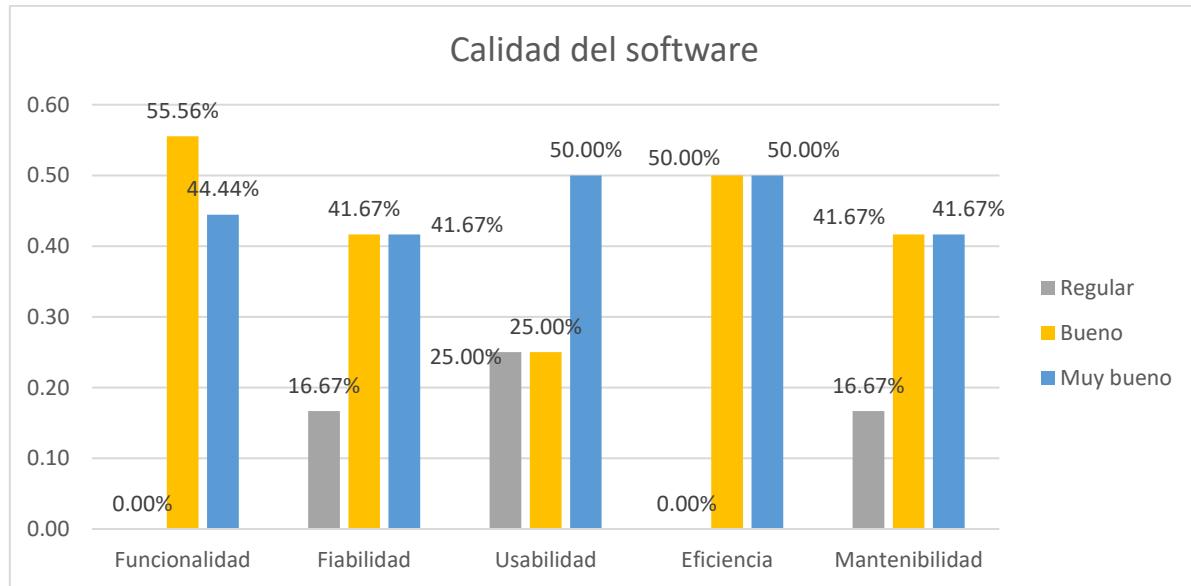


Fig. 123: Porcentaje de respuestas para los indicadores de la dimensión calidad del software

Respecto a la funcionalidad como se muestra en la figura 123, se tiene que, según las respuestas de los expertos, el 44% de sus respuestas es que la aplicación web y móvil es “muy bueno” y el 56% de sus respuestas dice que es “bueno”. En lo que es fiabilidad se tiene un 42% de las respuestas con respecto a la aplicación web y móvil se considera “muy bueno”, un 41% se considera “bueno” y solo un 17% se considera regular. Para el indicador usabilidad se aprecia que un 50% de las respuestas sobre la aplicación web y móvil considera “muy bueno”, un 25% considera “bueno” y un 25% considera “regular”, esto debido a que los expertos recomiendan mejorar la experiencia de usuario. Para el indicador eficiencia, el 50% de las respuestas afirma que la aplicación web y móvil es “muy bueno” y el otro 50% afirma que es “bueno”. En el indicador mantenibilidad el 42% de las respuestas afirma que la aplicación web y móvil es “muy bueno”, el 41% afirma que es “bueno” y solo el 17% afirma que es “regular”.

La variable dependiente tiene 3 dimensiones: tiempo, satisfacción y calidad. A continuación, se presenta los resultados obtenidos por cada dimensión.

3.2.3.2 DIMENSIÓN TIEMPO

En la tabla XXVII acerca de la dimensión tiempo se observa una reducción en lo que es tiempo de registro de un pedido, tiempo de registro de un pago y tiempo de elaboración de un reporte de ventas.

Tabla XXVII: Porcentaje de reducción en la dimensión tiempo.

Tiempo	Pretest(min)	Posttest(min)	Reducción (min)	% Reducción
Tiempo de registro de un pedido del cliente	04:18	01:19	02:59	69.37%
Tiempo de elaboración de un reporte de ventas	17:05	02:31	14:57	85.39%
Tiempo de registro de un pago de un pedido	03:04	02:20	00:44	23.69%

*Nota: los tiempos para calcular el porcentaje de reducción se han convertido a formato decimal (04:18 min = 4.30 min, etc.).

En la dimensión tiempo se observó una reducción significativa luego de la implementación de la aplicación web y móvil. El tiempo de registro de un pedido de un cliente se redujo de 4 minutos con 18 segundos a 1 minuto con 19 segundos en promedio, esto representa una disminución de 2 minutos con 59 segundos equivalente al 69.37% de reducción (para calcular el porcentaje se aplicó la fórmula, $\% Reducción = \left[\frac{(Valor\ Inicial - Valor\ Final)}{Valor\ Inicial} \right] * 100$). De igual manera, el tiempo de elaboración de un reporte de ventas tuvo una reducción notable al pasar de 17 minutos con 5 segundos a solamente 2 minutos con 31 segundos lo que equivale al 85.39% de reducción. A su vez, en el registro de un pago de un pedido se redujo en un 23.69%, pasando de 3 minutos con 4 segundos inicialmente a 2 minutos con 20 segundos. La significativa reducción en todos los indicadores evidencia que la implementación de la aplicación web y móvil influye en el tiempo de registro de un pedido y la elaboración de un reporte de ventas.

3.2.3.3 DIMENSIÓN SATISFACCIÓN

En la tabla XXVIII se muestra los porcentajes de mejora por cada indicador de la dimensión satisfacción, evaluados mediante una escala Likert.

Tabla XXVIII: Porcentaje de mejora para cada indicador de la dimensión satisfacción.

Satisfacción	Pretest (escala Likert)	Posttest (escala Likert)	Aumento (escala Likert)	%Mejora
Porcentaje de trabajadores satisfechos con el registro de los pedidos del cliente	1.9714	4.1429	2.1715	110.14%
Porcentaje de trabajadores satisfechos con la reducción de errores en las especificaciones de los pedidos	3.1333	4.0000	0.08667	27.66%
Porcentaje de trabajadores satisfechos con la gestión de reclamos	2.4000	4.1000	1.7000	70.83%
Porcentaje de trabajadores satisfechos con la elaboración de reportes	1.6400	4.0400	2.4000	146.34%

Para esta dimensión se observa que todos los indicadores presentaron mejoras luego de la implementación de la aplicación web y móvil. La satisfacción con la elaboración de reportes experimentó el mayor incremento (146.34%), seguido de la satisfacción con el registro de pedidos que aumentó 110.14%, la gestión de reclamos mostró una mejora del 70.83%, mientras que la satisfacción con la reducción de errores en las especificaciones de los pedidos registró una mejora del 27.66%. Estos resultados demuestran que la aplicación web y móvil ha generado una influencia positiva en la satisfacción de los trabajadores inmersos en el proceso de ventas en Nikos Pizza.

3.2.3.4 DIMENSIÓN CALIDAD

En la tabla XXIX se muestra el porcentaje de mejora para cada indicador de la dimensión calidad, evaluados mediante una escala Likert.

Tabla XXIX: Porcentaje de mejora para cada indicador de la dimensión calidad.

Calidad	Pretest (escala Likert)	Posttest (escala Likert)	Aumento (escala Likert)	%Mejora
Nivel de exactitud de los reportes de ventas	1.9333	4.0400	2.1067	108.96%
Nivel de consistencia de los reportes de ventas	1.5000	4.0000	2.5000	166.67%

Los resultados para la dimensión calidad muestran mejoras en ambos indicadores evaluados luego de la implementación de la aplicación web y móvil. El nivel de consistencia de los reportes de ventas fue el que experimentó un mayor incremento (166.96%), asimismo, el nivel de exactitud de los reportes mostró una mejora del 108.96%. Estos resultados evidencian que

la implementación de la aplicación web y móvil influye positivamente en la calidad de los reportes de ventas.

CAPÍTULO IV. ANÁLISIS Y DISCUSIÓN DE RESULTADOS

Para determinar la influencia de la aplicación web y móvil en el proceso de ventas de la empresa Nikos Pizza, se presentan los resultados en base a los objetivos específicos trazados para este estudio. Primero se identificó la situación actual de la empresa sin la implementación de la aplicación web y móvil para cada uno de las dimensiones de la variable dependiente, luego se procedió al diseño e implementación de la aplicación web y móvil teniendo como hipótesis la reducción del tiempo del registro de un pedido, registro de un pago y la elaboración de un reporte de ventas, a su vez el aumento de la satisfacción de los trabajadores y la calidad de los reportes; posteriormente, se evaluó la calidad del software determinando su desempeño en el proceso a estudiar. Finalmente, se identificó la situación de la empresa luego de implementar la aplicación web y móvil y se procedió a hacer una comparación entre el antes y el después, revelando así la influencia de la aplicación web y móvil en el proceso de ventas.

Para el primer objetivo específico (determinar la influencia de la implementación de una aplicación web y móvil en el tiempo de registro de pedidos, registro del pago y en la elaboración de reportes de ventas en Nikos Pizza), se ha obtenido resultados claves como una reducción del 69.30% en el tiempo de registro de un pedido, es decir lo que anteriormente se tardaba en promedio 4 minutos con 18 segundos, con el sistema se realiza en 1 minuto y 19 segundos. A su vez, en la elaboración de reportes se ha reducido el tiempo en 85.27% pasando de 17 minutos con 15 segundos en promedio antes de la implementación de la aplicación web y móvil, luego de la implementación el tiempo promedio es de 2 minutos y 31 segundos como también la reducción en un 23.69% en el tiempo de registro de un pago de un pedido. Estos hallazgos se alinean con estudios previos, como las de Huamán y Alvarado [11], quienes reportaron una reducción de 3.38 minutos en el registro de pedidos pasando a tomar un tiempo promedio de 1 minuto con 29 segundos luego de la implementación, pero la diferencia con la presente investigación que luego de la implementación toma en promedio 1 minuto con 19 segundos se debe a la cantidad de productos, que generalmente es una pizza. Asimismo Sakihara [12] documentó una reducción en el tiempo de un pedido en 58.99% lo que evidencia que la aplicación en Nikos Pizza tiene una mayor reducción del tiempo. A su vez, Herrera [13], también obtuvo una reducción del 75% en el tiempo de atención de pedidos luego de implementar un sistema web con algoritmo AHP, aunque en la investigación de Herrera el porcentaje de reducción es mayor eso se explica porque allí considera el tiempo de envío de un lugar hacia otro que está muy distante. Por otro lado, Gaona [14] registró una disminución del 66.6% en el tiempo de operación de ventas gracias a la implementación de una aplicación móvil,

este porcentaje tiene cierta diferencia con la presente investigación debido a que en la investigación de Gaona el cliente toma más tiempo en tomar una decisión. Estos hallazgos confirman que las aplicaciones web y móviles influyen de manera positiva en los procesos, independientemente del sector o contexto de aplicación. Asimismo, la integración con bases de datos en tiempo real ayuda a reducir el uso de papel y acelera procesos críticos lo que explica la drástica reducción de tiempo en la generación de reportes, coincidiendo con Quintana y Cunuhay [10], quienes documentaron disminuciones del 70%, claro que en la presente investigación la reducción es de 85.27% y esto se explica por el enfoque específico en el sector gastronómico, en donde sus reportes fueron diseñados para un fácil y rápido entendimiento.

En el segundo objetivo específico (determinar la influencia de la implementación de una aplicación web y móvil en la satisfacción de los trabajadores inmersos en el proceso de ventas de Nikos Pizza), se obtuvieron resultados que muestran una mejora del 110.14% en satisfacción de los trabajadores con el registro de pedidos y un aumento del 146.34% en la satisfacción con la elaboración de reportes. La reducción de carga operativa y correcta gestión de errores explica la mayor satisfacción, alineándose con Guerrero [15] quien reportó aumentos del 48% en trabajadores de restaurantes. La interfaz intuitiva y funciones como filtros automáticos en reportes redujeron estrés, respaldando la teoría de Gómez [34] sobre usabilidad como factor crítico en satisfacción laboral. A su vez, los resultados obtenidos revelan el potencial de una mejora mucho mayor cuando las soluciones tecnológicas se diseñan enfocados en la usabilidad y analizando bien la frustración de los trabajadores al utilizar las herramientas.

Para el tercer objetivo específico (determinar la influencia de la implementación de una aplicación web y móvil en la calidad de los reportes de ventas en Nikos Pizza), se encontraron resultados claves que muestran un aumento de 108.96% en la exactitud de reportes y un aumento de 166.67% en la consistencia de reportes. La eliminación de errores humanos en transcripción y cálculos justifica la mejora en calidad. Esto corrobora a Cayo y Seracapa [9], donde el 78% de usuarios percibió mayor calidad en reportes automatizados. Asimismo se alinea con los resultados de Rivera [8], donde su modelo de datos con dashboards interactivos y sus reportes dinámicos permiten a los trabajadores explorar los datos de manera intuitiva y detallada, claro que en la presente investigación los reportes permiten filtrar por producto, cliente y/o trabajador, además el dashboard permite visualizar por gráfico de líneas o gráfico de barras; todo esto se ve reflejado en el porcentaje de calidad de reportes percibida por los trabajadores.

Asimismo, la calidad del software presentó resultados claves en donde se pudo apreciar que en funcionalidad un 44% de las respuestas considera “muy bueno” a la aplicación web y móvil y un 56% considera “bueno”, asimismo, en usabilidad el 50% consideró “muy bueno” y aunque el 25% considera “regular” sugiere oportunidades de mejora en UX. La alta evaluación en funcionalidad valida el cumplimiento de requisitos. La usabilidad recibió críticas menores, coincidiendo con la norma ISO 25010 que destaca la curva de aprendizaje como desafío. La arquitectura MVC y el uso de Laravel/Kotlin fueron clave para la mantenibilidad de la aplicación.

CAPÍTULO V. CONCLUSIONES Y RECOMENDACIONES

5.1 CONCLUSIONES

La implementación de la aplicación web y móvil influyó de manera positiva en el proceso de ventas de la empresa Nikos Pizza, esto se puede apreciar en la reducción de tiempos en el registro de pedidos, en el registro de los pagos y en la elaboración de reportes, a su vez, aumentando la satisfacción laboral y mejorando la calidad de los reportes, lo que confirma la hipótesis general de la investigación.

La implementación de la aplicación web y móvil influyó de manera positiva en el tiempo, reduciendo en un 69.30% el tiempo de registro de pedidos, un 23.69% en el registro de un pago de un pedido y en un 85.27% el tiempo de elaboración de reportes de ventas, esto gracias a la eliminación de procesos manuales y la automatización de flujos críticos. Es así que se confirma la hipótesis específica de que la implementación de una aplicación web y móvil reduce el tiempo de registro de pedidos y la elaboración de reportes de ventas.

En cuanto a la satisfacción de los trabajadores se obtuvo un aumento de 146.34% de mejora en la elaboración de reportes, un 110.14% en el registro de pedidos, un 70.83% de aumento en la gestión de reclamos y un 27.66% de aumento en la satisfacción con respecto a la reducción de errores en las especificaciones. Estos resultados confirman la hipótesis específica acerca de que la implementación de la aplicación web y móvil aumenta la satisfacción de los trabajadores inmersos en el proceso de ventas.

Asimismo, para la calidad se obtuvo una mejora sustancial de 108.96% en exactitud de los reportes de ventas y un aumento de 166.67% en la consistencia de los reportes de ventas, todo esto gracias a la integración directa con bases de datos y herramientas de visualización interactiva. Estos resultados confirman la hipótesis específica de que la implementación de la aplicación web y móvil aumenta la calidad de los reportes de ventas en Nikos Pizza.

Tecnológicamente, la arquitectura basada en Laravel y la utilización de un Webview para la aplicación móvil probó ser robusta y escalable, con evaluaciones de calidad de software donde el 83% de expertos calificaron la mantenibilidad como “buena” o “muy buena”, sentando precedentes replicables para pymes del sector gastronómico.

5.2 RECOMENDACIONES

Se recomienda ampliar las capacidades de la aplicación web y móvil mediante funcionalidades hacia los clientes, integrando un sistema de compras en línea que permita a los clientes realizar sus pedidos de forma autónoma. Asimismo, se recomienda incorporar un módulo de facturación electrónica que cumpla con los requisitos de SUNAT. A su vez, se recomienda ampliar la aplicación móvil para iOS para ampliar la cobertura de mercado y garantizar una experiencia multiplataforma.

Para potenciar la capacidad de distribución, se recomienda integrar con plataformas de delivery externas (Rappi, PedidosYa, Uber Eats) mediante el desarrollo de APIs especializadas. Adicionalmente, se recomienda implementar múltiples opciones de pago electrónico para llegar a más clientes y reducir las barreras de compra.

Debido a su influencia positiva en Nikos Pizza se recomienda replicar el estudio en otras pymes del sector gastronómico para validar su escalabilidad y compartir infraestructura tecnológica promoviendo así una transformación digital accesible y de alto impacto. Por otra parte, se recomienda implementar un programa de capacitación trimestral hacia los trabajadores y que profundice en funciones avanzadas de la aplicación, especialmente en el uso de análisis predictivo mediante inteligencia artificial.

REFERENCIAS BIBLIOGRÁFICAS

- [1] I. Beerepoot *et al.*, «The biggest business process management problems to solve before we die», *Comput. Ind.*, vol. 146, p. 103837, abr. 2023, doi: 10.1016/j.compind.2022.103837.
- [2] R. Rodríguez, G. Svensson, y E. J. Mehl, «Digitalization process of complex B2B sales processes – Enablers and obstacles», *Technol. Soc.*, vol. 62, p. 101324, ago. 2020, doi: 10.1016/j.techsoc.2020.101324.
- [3] A. R. Kusuma, R. Syarie, A. Sukmawati, y A. Ekananta, «Factors influencing the digital transformation of sales organizations in Indonesia», *Helijon*, vol. 10, n.º 5, p. e27017, mar. 2024, doi: 10.1016/j.helijon.2024.e27017.
- [4] R. Icasuriaga, «La Transformación Digital de las PyMEs en Latinoamérica», News Center Latinoamérica. Accedido: 23 de mayo de 2024. [En línea]. Disponible en: <https://news.microsoft.com/es-xl/la-transformacion-digital-de-las-pymes-en-latinoamerica-4-hallazgos-clave-sobre-su-recorrido/>
- [5] «Casi la mitad de los empleados están dispuestos a dejar su empleador», WTW. Accedido: 31 de mayo de 2024. [En línea]. Disponible en: <https://www.wtwco.com/es-ar/insights/2022/04/infographic-more-than-half-of-employees-open-to-leaving-employer>
- [6] C. Torres Tutiven, «Gremio de restaurantes peruanos enfrentan dificultades debido a los altos costos y la inflación», América Noticias. Accedido: 7 de mayo de 2024. [En línea]. Disponible en: <https://www.americatv.com.pe/noticias/actualidad/restaurantes-enfrentan-dificultades-debido-altos-costos-y-inflacion-n482993>
- [7] J. A. Badajoz Ramos, M. J. Jaime Flores, D. B. Martínez Quispe, y R. Conde Vera, «Calidad de servicio en la decisión de compra en centros comerciales de Perú», *Quipukamayoc*, vol. 31, n.º 65, pp. 61-71, jul. 2023, doi: 10.15381/quipu.v31i65.24665.
- [8] Rivera Vaca, Francisco Javier, «Modelo de datos para la gestión de ventas de la empresa MegaHome», Trabajo de Titulación, Universidad Nacional de Chimborazo, Riobamba, Ecuador, 2024.
- [9] W. C. Cayo Gallardo Wendy y L. A. Seracapa Achote, «Desarrollo de un sistema web de gestión de ventas para la empresa EcoMejia mediante la aplicación de inteligencia de negocios», Tesis, Universidad Técnica de Cotopaxi, Latacunga-Ecuador, 2024. [En línea]. Disponible en: <https://repositorio.utc.edu.ec/bitstream/27000/12057/1/PI-002761.pdf>
- [10] W. C. Cunuhay Cuchipe y P. E. Quintana Herrera, «Análisis comparativo de aplicaciones web de control de ventas en restaurantes para la toma de decisiones», *Bastcorp Int. J.*, vol. 4, n.º 1, ene. 2025, doi: 10.62943/bij.v4n1.2025.169.
- [11] S. M. Huamán Sanchez y M. A. Alvarado Trillo, «Elaboración de un sistema web para la optimización de la gestión de pedidos en la empresa Chifa Kam Lun de Ica», Tesis, Universidad Autónoma de Ica, Ica, 2021. [En línea]. Disponible en: <https://hdl.handle.net/20.500.14441/916>
- [12] F. Sakihara Kunikami, «Influencia de la aplicación de un sistema informático en el proceso de venta en una empresa del sector gastronómico en el Perú», Tesis, Universidad Nacional Mayor de San Marcos, Lima, 2022. [En línea]. Disponible en: <https://hdl.handle.net/20.500.12672/18287>

[13] D. L. Herrera Cagusol, «Sistema web para optimizar la gestión de pedidos de la empresa agrícola Viña Vieja Viña Santa Isabel S. A. C.», Tesis, Universidad Católica Santo Toribio de Mogrovejo, Chiclayo, 2018. [En línea]. Disponible en: https://tesis.usat.edu.pe/xmlui/bitstream/handle/20.500.12423/1600/TL_HerreraCagusolDora.pdf?sequence=1&isAllowed=y

[14] I. Gaona Arredondo, «Diseño e implementación de una aplicación móvil para mejorar el proceso de venta de líneas prepago en una empresa de telecomunicaciones», Tesis, Universidad Tecnológica del Perú, Lima, 2020. [En línea]. Disponible en: https://repositorio.utp.edu.pe/bitstream/handle/20.500.12867/3503/Ivan%20Gaona_Trabajo%20de%20Suficiencia%20Profesional_Titulo%20Profesional_2020.pdf?sequence=3&isAllowed=y

[15] B. F. Guerrero Díaz, «Efecto de la implementación de un sistema de información en los procesos de atención al cliente en el recreo campestre “La Querencia”, Cajamarca 2022», Tesis, Universidad Nacional de Cajamarca, Cajamarca, 2023. [En línea]. Disponible en: <http://hdl.handle.net/20.500.14074/6025>

[16] «¿Qué es una aplicación web? - Explicación de las aplicaciones web - AWS», ¿Qué es una aplicación web? Accedido: 9 de mayo de 2024. [En línea]. Disponible en: <https://aws.amazon.com/es/what-is/web-application/>

[17] J. Chavira-García, A. A. Arredondo-López, J. Chavira-García, y A. A. Arredondo-López, «Aplicaciones móviles como herramientas en los servicios de salud», *Horiz. Sanit.*, vol. 16, n.º 2, pp. 85-91, ago. 2017, doi: 10.19136/hs.v16n2.1498.

[18] «Definición y cómo funcionan las aplicaciones móviles», Softcorp. Accedido: 26 de marzo de 2024. [En línea]. Disponible en: <https://servisoftcorp.com/definicion-y-como-funcionan-las-aplicaciones-moviles/>

[19] D. A. Biancha, «Diseño de framework web para el desarrollo dinámico de aplicaciones», vol. XVI, n.º 44, pp. 178-183, 2010.

[20] S. Necula, «Exploring The Model-View-Controller (MVC) Architecture: A Broad Analysis of Market and Technological Applications», abr. 2024, doi: 10.20944/preprints202404.1860.v1.

[21] J. Kouraklis, «MVVM as Design Pattern», en *MVVM in Delphi*, Berkeley, CA: Apress, 2016, pp. 1-12. doi: 10.1007/978-1-4842-2214-0_1.

[22] M. Tulin, «Native vs WebView: What to choose for your news app?» Accedido: 14 de enero de 2025. [En línea]. Disponible en: <https://www.vitec-visiolink.com/insights/blog/native-vs-webview-what-to-choose-for-your-news-app/>

[23] J. Sánchez Gallego, *Dirección de proyectos entre David y Goliat: marco común para colaborar*. Aula Magna Proyecto clave McGraw Hill, 2024.

[24] B. Lutkevich, «What is the Waterfall Model? - Definition and Guide», Software Quality. Accedido: 4 de abril de 2024. [En línea]. Disponible en: <https://www.techtarget.com/searchsoftwarequality/definition/waterfall-model>

[25] «ISO 25010», ISO 25000. Accedido: 9 de mayo de 2024. [En línea]. Disponible en: <https://iso25000.com/index.php/normas-iso-25000/iso-25010>

[26] N. J. Acosta, L. A. Espinel, y J. L. García, «Estándares para la calidad de software», vol. 5, n.º 1, 2017.

[27] J. D. Blischak, E. R. Davenport, y G. Wilson, «A Quick Introduction to Version Control with Git and GitHub», *PLOS Comput. Biol.*, vol. 12, n.º 1, p. e1004668, ene. 2016, doi: 10.1371/journal.pcbi.1004668.

[28] M. Á. Mallar, «La Gestión por Procesos: un enfoque de gestión eficiente», *Visión Futuro*, vol. 13, n.º 1, pp. 0-0, jun. 2010.

[29] J. D. G. De Leon Valdes, J. D. J. Romero Alvarez, y O. E. Fuentes Martínez, «Effective Sales Process», *J. Adm. Sci.*, vol. 5, n.º 9, pp. 12-16, jul. 2023, doi: 10.29057/jas.v5i9.9880.

[30] D. Jobber, «Administracion de Ventas».

[31] «¿Cómo Revolucionan las Comandas Electrónicas los Restaurantes?» Accedido: 16 de enero de 2025. [En línea]. Disponible en: <https://merchants.rappi.com/es-co/comandas-electronicas>

[32] J. Konrath, *Snap Selling: Speed Up Sales and Win More Business with Today's Frazzled Customers*. England, 2010.

[33] «¿Cómo se puede utilizar el tiempo de procesamiento de pedidos como métrica de rendimiento en las operaciones de almacén?» Accedido: 16 de enero de 2025. [En línea]. Disponible en: <https://www.linkedin.com/advice/3/how-can-you-use-order-processing-time-performance-8nmfe?lang=es&originalSubdomain=es>

[34] D. Gómez, «Qué es la satisfacción laboral (y cómo medirla y mejorarla)». Accedido: 28 de marzo de 2024. [En línea]. Disponible en: <https://blog.hubspot.es/service/satisfaccion-laboral>

[35] T. C. Redman, «The impact of poor data quality on the typical enterprise», *Commun. ACM*, vol. 41, n.º 2, pp. 79-82, feb. 1998, doi: 10.1145/269012.269025.

[36] R. Piontekewicz, M. D. C. D. Freitas, A. Kemczinski, y C. Duran San Martin, «Management of intellectual capital in a system of management accounting information», en *2016 6th International Conference on Computers Communications and Control (ICCCC)*, Oradea: IEEE, may 2016, pp. 180-187. doi: 10.1109/ICCCC.2016.7496758.

[37] H. Almohri, R. B. Chinnam, y M. Colosimo, *Data-Driven Analytics for Benchmarking and Optimizing Retail Store Performance*. arXiv, 2018. Accedido: 13 de mayo de 2024. [En línea]. Disponible en: <https://doi.org/10.48550/arXiv.1806.05563>

[38] G. Boronat Ombuena, R. Leotescu, y J. Navarro Enguídanos, «CAPEX Y OPEX. La GESTIÓN de las inversiones empresariales», p. 13, 2019.

[39] F. Marcillo, Á. E. Vinueza Verdezoto, A. M. Romero López, y L. Begnini, «La inteligencia artificial como aliada en la gestión de ventas: Caso de estudio empresa MakroHospital | Revista Científica Kosmos», 24-07-2024, vol. Vol. 3, n.º Núm. 2, Accedido: 15 de enero de 2025. [En línea]. Disponible en: <https://editorialinnova.com/index.php/rck/article/view/90>

[40] R. G. Castagnola, «La inteligencia artificial en la gestión de ventas, profundizándose y sin su permiso».

[41] P. Kotler y G. Armstrong, *Principles of marketing*, 17th [edition]. en Always learning. New York, NY Hoboken: Pearson Higher Education, 2018.

[42] R. Hernández Sampieri, *Metodología de la Investigación*. México, 2014.

[43] «Journal of Applied Psychology». Accedido: 10 de septiembre de 2025. [En línea]. Disponible en: <https://www.apa.org/pubs/journals/apl>

[44] «Libro-1.-E.Produccion.-Tema10-CalidadFinal2Edicion.pdf». Accedido: 10 de septiembre de 2025. [En línea]. Disponible en: <https://www.esfernand.es/wp-content/uploads/2020/11/Libro-1.-E.Produccion.-Tema10-CalidadFinal2Edicion.pdf>

[45] «International Vocabulary of Metrology – Basic and general concepts and associated terms (VIM), 3rd edition». Accedido: 10 de septiembre de 2025. [En línea]. Disponible en: <https://www.bipm.org/doi/10.59161/JCGM200-2012>

[46] J. M. Juran, Ed., *Juran's quality handbook*, 5. ed. New York, NY: McGraw-Hill, 1999.

[47] «¿Qué es un marco de trabajo? - Explicación sobre los marcos de trabajo en programación e ingeniería - AWS», Amazon Web Services, Inc. Accedido: 8 de octubre de 2025. [En línea]. Disponible en: <https://aws.amazon.com/es/what-is/framework/>

[48] R. A. Rojas y S. D. Contreras, «Revisión de Framework Laravel y su aplicación en sistema web de mesa de ayuda», 2023.

[49] «Introducción | Laravel Jetstream». Accedido: 8 de octubre de 2024. [En línea]. Disponible en: <https://jetstream.laravel.com/introduction.html>

[50] «Programación en Kotlin - Infografía de Developer Ecosystem en 2021», JetBrains: Developer Tools for Professionals and Teams. Accedido: 8 de octubre de 2024. [En línea]. Disponible en: <https://www.jetbrains.com/lp/devecosystem-2021>

[51] «Aspectos básicos de Jetpack Compose», Android Developers. Accedido: 8 de octubre de 2024. [En línea]. Disponible en: <https://developer.android.com/codelabs/jetpack-compose-basics?hl=es-419>

[52] P. Londoño, «Qué es MySQL, para qué sirve y características principales». Accedido: 8 de octubre de 2024. [En línea]. Disponible en: <https://blog.hubspot.es/website/que-es-mysql>

[53] «¿Qué es un ORM y para qué sirve Eloquent?», /clases/1920-eloquent-laravel/28515-que-es-un-orm-y-para-que-sirve-eloquent/. Accedido: 8 de octubre de 2024. [En línea]. Disponible en: <https://platzi.com/clases/1920-eloquent-laravel/28515-que-es-un-orm-y-para-que-sirve-eloquent/>

[54] «Cómo crear apps web en WebView», Android Developers. Accedido: 8 de octubre de 2024. [En línea]. Disponible en: <https://developer.android.com/develop/ui/views/layout/webapps/webview?hl=es-419>

[55] «Acerca de GitHub y Git - Documentación de GitHub», GitHub Docs. Accedido: 8 de octubre de 2024. [En línea]. Disponible en: <https://docs.github.com/es/get-started/start-your-journey/about-github-and-git>

[56] «Entender las GitHub Actions - Documentación de GitHub», GitHub Docs. Accedido: 8 de octubre de 2024. [En línea]. Disponible en: <https://docs.github.com/es/actions/about-github-actions/understanding-github-actions>

[57] «Hosting web: una introducción | IBM». Accedido: 8 de octubre de 2024. [En línea]. Disponible en: <https://www.ibm.com/mx-es/topics/web-hosting>

[58] «¿Qué es un servidor privado virtual (VPS)?», Google Cloud. Accedido: 8 de octubre de 2024. [En línea]. Disponible en: <https://cloud.google.com/learn/what-is-a-virtual-private-server>

[59] C. Alvarado, «Composer es un manejador de dependencias para PHP», Centro Universitario Regional de Cabañas, Universidad Luterna Salvadoreña. Accedido: 8 de octubre de 2024. [En línea]. Disponible en: <http://curc.ulsa.edu.sv/sopor/contenido.php?id=2>

[60] «¿Qué es un usuario? Explicación de los permisos de usuario y el DCU | Lenovo España». Accedido: 9 de octubre de 2025. [En línea]. Disponible en: <https://www.lenovo.com/es/es/glossary/what-is-a-user/>

[61] «¿Qué es el diseño responsive? | Lenovo Perú». Accedido: 9 de octubre de 2025. [En línea]. Disponible en: <https://www.lenovo.com/pe/es/glosario/que-es-el-diseno-responsive/>

[62] «Definición de roles». Accedido: 9 de octubre de 2025. [En línea]. Disponible en: <https://www.ibm.com/docs/es/product-master/12.0.0?topic=model-defining-roles>

[63] G. B. Baena Paz, *Metodología de la investigación*. México, 2014.

[64] E. A. Rodríguez Moguel, *Metodología de la investigación*. México, 2005.

[65] J. Casas Anguita, J. R. Repullo Labrador, y J. Donado Campos, «La encuesta como técnica de investigación. Elaboración de cuestionarios y tratamiento estadístico de los datos (I)», *Aten. Primaria*, vol. 31, n.º 8, pp. 527-538, may 2003.

[66] R. S. Pressman, *Ingeniería del Software. Un Enfoque Práctico*, Séptima edición. Mexico.

[67] «Enfoque de prioridad de Kotlin en Android | Android Developers». Accedido: 9 de octubre de 2025. [En línea]. Disponible en: <https://developer.android.com/kotlin/first?hl=es-419>

[68] «Blog para desarrolladores de Android: novedades de Jetpack Compose». Accedido: 9 de octubre de 2025. [En línea]. Disponible en: <https://android-developers.googleblog.com/2025/05/whats-new-in-jetpack-compose.html>

ANEXOS

ANEXO 1: OPERACIONALIZACIÓN DE VARIABLES

Tabla XXX: Operacionalización de variables

VARIABLES	DEFINICIÓN CONCEPTUAL	DIMENSIÓN	INDICADOR	TÉCNICA/INSTRUMENTO
Aplicación web y móvil	Es el software diseñado para ejecutarse en navegadores web y dispositivos móviles, permitiendo a las empresas mostrar información en tiempo real [16].	Calidad del software	Funcionalidad: Número de funcionalidades implementadas según requisitos del usuario. Fiabilidad: Porcentaje de fallos de la aplicación web y móvil. Usabilidad: Tiempo requerido para aprender a usar la aplicación Eficiencia: Tiempo de arranque de la aplicación en frio Mantenibilidad: Número de errores corregidos	Encuesta / cuestionario
Proceso de ventas	Un proceso, del latín <i>processus</i> que significa "avance" o "progreso", es un conjunto de pasos organizados para alcanzar un objetivo específico. En el ámbito empresarial, el proceso productivo transforma materia prima, insumos y conocimientos en una solución final [22]. El proceso de ventas incluye identificar el mercado objetivo, realizar una buena presentación personal y de la empresa, entender las necesidades del cliente, ofrecer y presentar el producto adecuado, cerrar la venta y brindar servicio post-venta [23]	Tiempo Satisfacción Calidad	Tiempo de registro del pedido del cliente. Tiempo de la elaboración de reportes de ventas. Tiempo de registro del pago Porcentaje de trabajadores satisfechos con el registro de los pedidos del cliente. Porcentaje de trabajadores satisfechos con la reducción de errores en las especificaciones de los pedidos. Porcentaje de trabajadores satisfechos con la gestión de reclamos Porcentaje de trabajadores satisfechos con la elaboración de reportes de ventas. Nivel de exactitud de los reportes de ventas Nivel de consistencia de los reportes de ventas	Observación/ficha de observación Encuesta / cuestionario

ANEXO 2: MATRIZ DE CONSISTENCIA

Tabla XXXI: Matriz de consistencia

PROBLEMA GENERAL	OBJETIVO GENERAL	HIPÓTESIS GENERAL	VARIABLES
¿De qué manera influye la implementación de una aplicación web y móvil en el proceso de ventas en la empresa Nikos Pizza?	Determinar la influencia de la implementación de una aplicación web y móvil en el proceso de ventas de la empresa Nikos Pizza.	La implementación de una aplicación web y móvil influye de manera positiva en el proceso de ventas de la empresa Nikos Pizza.	Variable independiente: Aplicación web y móvil Variable dependiente: Proceso de ventas
PROBLEMAS ESPECÍFICOS	OBJETIVOS ESPECÍFICOS	HIPÓTESIS ESPECÍFICAS	
<ul style="list-style-type: none"> - ¿De qué manera influye la implementación de una aplicación web y móvil en el tiempo de registro de pedidos, en el registro de pagos y en la elaboración de reporte de ventas en Nikos Pizza? - ¿Cómo influye la implementación de una aplicación web y móvil en la satisfacción de los trabajadores inmersos en el proceso de ventas de Nikos Pizza? - ¿De qué manera influye la implementación de una aplicación web y móvil en la calidad de los reportes de ventas en Nikos Pizza? 	<ul style="list-style-type: none"> - Determinar la influencia de la implementación de una aplicación web y móvil en el tiempo de registro de pedidos, registro del pago y en la elaboración de reportes de ventas en Nikos Pizza. - Determinar la influencia de la implementación de una aplicación web y móvil en la satisfacción de los trabajadores inmersos en el proceso de ventas de Nikos Pizza. - Determinar la influencia de la implementación de una aplicación web y móvil en la calidad de los reportes de ventas en Nikos Pizza. 	<ul style="list-style-type: none"> - La implementación de una aplicación web y móvil reduce el tiempo de registros de los pedidos, registro de los pagos y la elaboración de reportes de ventas en Nikos Pizza. - La implementación de una aplicación web y móvil aumenta la satisfacción de los trabajadores inmersos en el proceso de ventas de Nikos Pizza. - La implementación de una aplicación web y móvil aumenta la calidad de los reportes de ventas en Nikos Pizza. 	

ANEXO 3: CUESTIONARIO PARA LA VARIABLE INDEPENDIENTE

Tabla XXXII: Cuestionario para la variable independiente

Cuestionario para medir la variable independiente (aplicación web y móvil)								
Encuestador: Roberto Jhuliño Tafur Valdivia			Fecha:					
Objetivo: Medir la calidad del software a través de los indicadores de funcionalidad, fiabilidad, usabilidad, eficiencia y mantenibilidad.								
Instrucciones: Responda cada pregunta de acuerdo a su percepción marcando con una equis (X) en el recuadro correspondiente.								
Este cuestionario forma parte de una investigación académica cuyo objetivo es determinar la influencia de una aplicación web y móvil en el proceso de ventas. La información recopilada será utilizada exclusivamente con fines científicos y académicos.								
Muy malo 1	Malo 2	Regular 3	Bueno 4	Muy bueno 5				
Datos generales del experto								
Entidad en la que labora:								
Cargo que desempeña:								
Cuestionario				Valor cuantitativo				
				1	2	3	4	5
Funcionalidad								
1	La aplicación implementa todas las funcionalidades solicitadas en los requisitos.							
2	La aplicación permite registrar pedidos y compras sin problemas.							
3	Los gráficos y reportes de ventas cubren toda la información necesaria.							
Fiabilidad								
4	La aplicación funciona la mayor parte del tiempo sin fallar.							
5	Si ocurre una falla, el trabajo puede continuar rápidamente.							
6	Los datos no se pierden cuando sucede un error.							
7	Al repetir una acción, la aplicación siempre ofrece el mismo resultado.							
Usabilidad								
8	Un usuario promedio puede aprender a utilizar la aplicación en poco tiempo.							
9	La organización de menús, botones e iconos hace que la navegación sea intuitiva sin manuales extensos.							
10	Los mensajes de la aplicación son claros cuando algo sale mal.							

11	Los colores y el tamaño de letra facilitan la lectura de la información.					
Eficiencia						
12	Las pantallas se abren rápidamente.					
13	Las búsquedas responden sin demoras.					
14	La aplicación no vuelve lento el dispositivo mientras se usa.					
15	Los reportes se generan en poco tiempo.					
Mantenibilidad						
16	La aplicación se puede actualizar sin afectar otras partes.					
17	Los errores reportados pueden corregirse con rapidez y sin generar nuevos problemas.					
18	Es sencillo agregar nuevas funciones a futuro.					
19	Los registros(logs) y mensajes de error proveen información suficiente para diagnosticar fallas.					

ANEXO 4: FICHA DE OBSERVACIÓN PARA LA VARIABLE DEPENDIENTE, DIMENSIÓN TIEMPO

Tabla XXXIII: Ficha de observación para la dimensión tiempo (pretest)

Ficha de observación N°1			Medición: Pre-test () Post-test ()	
Título de la investigación: Influencia de una aplicación web y móvil en el proceso de ventas de la empresa Nikos Pizza – Cajamarca				
Objetivo: Medir el tiempo en el registro de un pedido antes y después de implementar la aplicación web y móvil.				
Observado por: Roberto Jhuliño Tafur Valdivia			Herramienta: Cronómetro	
Trabajador	Proceso: Registrar pedido			
	Actividades (min)			Tiempo total
	Actividad 1	Actividad 2	Actividad 3	.. Actividad n
C1				
C2				
C3				
...				
...				
...				
C20				

Tabla XXXIV: Ficha de observación para la dimensión tiempo (postest)

Ficha de observación N°2			Medición: Pre-test () Post-test ()	
Título de la investigación: Influencia de una aplicación web y móvil en el proceso de ventas de la empresa Nikos Pizza – Cajamarca				
Objetivo: Medir el tiempo en la elaboración de un reporte de ventas antes y después de implementar la aplicación web y móvil.				
Observado por: Roberto Jhuliño Tafur Valdivia			Herramienta: Cronómetro	
Trabajador	Proceso: Elaboración de reportes de ventas			
	Actividades (min)			Tiempo total
	Actividad 1	Actividad 2	Actividad 3	.. Actividad n
C1				
C2				
C3				
C4				
C5				
C6				
C7				
C8				

ANEXO 5: CUESTIONARIO PARA LA VARIABLE DEPENDIENTE, DIMENSIÓN SATISFACCIÓN Y CALIDAD

Objetivo: Evaluar la satisfacción y la calidad de las herramientas utilizadas en el proceso de ventas.

Instrucciones: Responda cada pregunta según su experiencia actual. Marca con una “x” la letra de la respuesta. Los datos recolectados son anónimos y solo se utilizarán para la presente investigación.

Tipo de medición: () Pre test () Post test

1. ¿Está usted satisfecho con la forma actual del registro de pedidos?

() Muy en desacuerdo () En desacuerdo () Neutral () De acuerdo () Muy de acuerdo

2. ¿Las herramientas actuales permite registrar pedidos correctamente sin omitir información importante?

() Muy en desacuerdo () En desacuerdo () Neutral () De acuerdo () Muy de acuerdo

3. ¿Es posible la gestión y seguimiento de los pedidos realizados de manera eficiente?

() Muy en desacuerdo () En desacuerdo () Neutral () De acuerdo () Muy de acuerdo

4. ¿El tiempo necesario para registrar o gestionar un pedido es adecuado con las herramientas actuales?

() Muy en desacuerdo () En desacuerdo () Neutral () De acuerdo () Muy de acuerdo

5. ¿El uso de la herramienta reduce significativamente la demora en el registro de pedidos?

() Muy en desacuerdo () En desacuerdo () Neutral () De acuerdo () Muy de acuerdo

6. ¿La herramienta reduce significativamente el tiempo de generación del reporte de ventas?

() Muy en desacuerdo () En desacuerdo () Neutral () De acuerdo () Muy de acuerdo

7. ¿Con la herramienta se reduce significativamente el tiempo de registro de los reclamos?

() Muy en desacuerdo () En desacuerdo () Neutral () De acuerdo () Muy de acuerdo

8. ¿La herramienta utilizada evita pérdidas de datos durante el registro o consulta de pedidos?

() Muy en desacuerdo () En desacuerdo () Neutral () De acuerdo () Muy de acuerdo

9. ¿La herramienta actual es fácil de aprender y usar por los trabajadores?

() Muy en desacuerdo () En desacuerdo () Neutral () De acuerdo () Muy de acuerdo

10. ¿La verificación de las especificaciones antes de realizar un pedido son fáciles de realizar?

() Muy en desacuerdo () En desacuerdo () Neutral () De acuerdo () Muy de acuerdo

11. ¿El proceso de registro y gestión de pedidos es adecuado para los empleados involucrados en el proceso de ventas?

() Muy en desacuerdo () En desacuerdo () Neutral () De acuerdo () Muy de acuerdo

12. ¿La gestión de reclamos es adecuado con las herramientas actuales?

() Muy en desacuerdo () En desacuerdo () Neutral () De acuerdo () Muy de acuerdo

13. ¿El proceso de elaboración de reportes de ventas es fácil y rápido?

() Muy en desacuerdo () En desacuerdo () Neutral () De acuerdo () Muy de acuerdo

14. ¿Se puede acceder a los reportes de ventas cuando se los necesita?

() Muy en desacuerdo () En desacuerdo () Neutral () De acuerdo () Muy de acuerdo

15. ¿El proceso actual facilita la toma de decisiones al proporcionar información precisa y actualizada sobre las ventas?

() Muy en desacuerdo () En desacuerdo () Neutral () De acuerdo () Muy de acuerdo

16. ¿Es fácil realizar ajustes o modificaciones de la información registrada con las herramientas actuales?

Muy en desacuerdo En desacuerdo Neutral De acuerdo Muy de acuerdo

17. ¿Las actualizaciones o modificaciones necesarias en la herramienta se realizan sin complicaciones?

Muy en desacuerdo En desacuerdo Neutral De acuerdo Muy de acuerdo

18. ¿ Se puede registrar operaciones (ventas o compras) con un bajo nivel de errores o interrupciones?

Muy en desacuerdo En desacuerdo Neutral De acuerdo Muy de acuerdo

19. ¿Es fácil realizar reportes de ventas por fecha sin errores?

Muy en desacuerdo En desacuerdo Neutral De acuerdo Muy de acuerdo

20. ¿Los productos más vendidos están correctamente listados?

Muy en desacuerdo En desacuerdo Neutral De acuerdo Muy de acuerdo

21. ¿Los datos registrados sobre las ventas son consistentes y precisos?

Muy en desacuerdo En desacuerdo Neutral De acuerdo Muy de acuerdo

22. ¿Los reportes generados con la herramienta son coherentes entre sí?

Muy en desacuerdo En desacuerdo Neutral De acuerdo Muy de acuerdo

ANEXO 6: FICHAS DE VALIDACIÓN PARA EL CUESTIONARIO DE LA VARIABLE INDEPENDIENTE

FICHA PARA VALIDACIÓN DEL INSTRUMENTO

I. REFERENCIA

- 1.1. Experto: Néstor Elías Muñoz Abanto
- 1.2. Especialidad:
- 1.3. Cargo actual: Docente
- 1.4. Grado académico: Ingeniero de Sistemas
- 1.5. Institución: UNC
- 1.6. Tipo de instrumento: Cuestionario
- 1.7. Lugar y fecha: 02-06-2025

II. TABLA DE VALORACIÓN POR EVIDENCIAS

Nº	EVIDENCIAS	VALORACION					
		5	4	3	2	1	0
1	Pertinencia de indicadores	x					
2	Formulado con lenguaje apropiado	x					
3	Adecuado para los sujetos en estudio	x					
4	Facilita la prueba de hipótesis	x					
5	Suficiencia para medir la variable		x				
6	Facilita la interpretación del instrumento	x					
7	Acorde al avance de la ciencia y tecnología	x					
8	Expresado en hechos perceptibles	x					
9	Tiene secuencia lógica	x					
10	Basado en aspectos teóricos	x					
	Total	5 2	3	3			

Coeficiente de valoración porcentual: $c = 80\%$

III. OBSERVACIONES Y/O RECOMENDACIONES

Sugiero considerar la dimensión de compatibilidad multiplataforma, así como la dimensión de seguridad, por ejemplo, si hay respaldos de los datos. Coordine con su asesor para determinar si en el título de su tesis, el proceso de ventas o sería de registro ya que en los instrumentos no hay medición directa de ventas reales (volumen, ingresos, ticket promedio, conversión, etc.)



Firma y sello del Experto

Fig. 124: Ficha 1 de validación del instrumento (variable independiente)

FICHA PARA VALIDACIÓN DEL INSTRUMENTO

I. REFERENCIA

- 1.1. Experto: Ing. Lisi Janet Vásquez Fernández
- 1.2. Especialidad: Ingeniería de Sistemas
- 1.3. Cargo actual: Docente Nombrada.
- 1.4. Grado académico: Doctor en Ciencias
- 1.5. Institución: Universidad Nacional de Cajamarca
- 1.6. Tipo de instrumento: Cuestionario
- 1.7. Lugar y fecha: 13-06-2025

II. TABLA DE VALORACIÓN POR EVIDENCIAS

Nº	EVIDENCIAS	VALORACION					
		5	4	3	2	1	0
1	Pertinencia de indicadores	X					
2	Formulado con lenguaje apropiado	X					
3	Adequado para los sujetos en estudio	X					
4	Facilita la prueba de hipótesis	X					
5	Suficiencia para medir la variable	X					
6	Facilita la interpretación del instrumento	X					
7	Acorde al avance de la ciencia y tecnología		X				
8	Expresado en hechos perceptibles	X					
9	Tiene secuencia lógica	X					
10	Basado en aspectos teóricos		X				
	Total	4	8				
		0					

Coeficiente de valoración porcentual: $c = 96\%$

III. OBSERVACIONES Y/O RECOMENDACIONES



Firma y sello del Experto

Fig. 125: Ficha 2 de validación del instrumento (variable independiente)

ANEXO 7: FICHAS DE VALIDACIÓN PARA EL CUESTIONARIO DE LA VARIABLE DEPENDIENTE

FICHA PARA VALIDACIÓN DEL INSTRUMENTO

I. REFERENCIA

- 3.8. Experto: Néstor Elías Muñoz Abanto
- 3.9. Especialidad:
- 3.10. Cargo actual: Docente
- 3.11. Grado académico: Ingeniero de Sistemas
- 3.12. Institución: UNC
- 3.13. Tipo de instrumento: Cuestionario
- 3.14. Lugar y fecha: 02-06-2025

II. TABLA DE VALORACIÓN POR EVIDENCIAS

Nº	EVIDENCIAS	VALORACION					
		5	4	3	2	1	0
1	Pertinencia de indicadores	x					
2	Formulado con lenguaje apropiado	x					
3	Adecuado para los sujetos en estudio	x					
4	Facilita la prueba de hipótesis	x					
5	Suficiencia para medir la variable			x			
6	Facilita la interpretación del instrumento	x					
7	Acorde al avance de la ciencia y tecnología	x					
8	Expresado en hechos perceptibles	x					
9	Tiene secuencia lógica	x					
10	Basado en aspectos teóricos	x					
	Total	5	3	3	2		

Coefficiente de valoración porcentual: $c = 80\%$

III. OBSERVACIONES Y/O RECOMENDACIONES

Uniformizar la escala de respuesta:

Actualmente varía entre letras (a-e) y texto.

Evita preguntas con doble contenido: Divida de manera independiente una para cada

proceso : registro de pedidos, reporte de ventas y/o registro de reclamos?"

Ítem "¿El uso de la herramienta reduce significativamente las demoras en el registro de pedidos, reporte de ventas y/o registro de reclamos?"

Observaciones levantadas

Néstor Elías Muñoz Abanto

Firma y sello del Experto

Fig. 126: Ficha 1 de validación para el cuestionario (variable dependiente)

FICHA PARA VALIDACIÓN DEL INSTRUMENTO

I. REFERENCIA

- 1.1. Experto: Ing. Lisi Janet Vásquez Fernández
- 1.2. Especialidad: Ingeniería de Sistemas
- 1.3. Cargo actual: Docente Nombrada.
- 1.4. Grado académico: Doctor en Ciencias
- 1.5. Institución: Universidad Nacional de Cajamarca
- 1.6. Tipo de instrumento: Ficha de observación
- 1.7. Lugar y fecha: 13-06-2025

II. TABLA DE VALORACIÓN POR EVIDENCIAS

Nº	EVIDENCIAS	VALORACION					
		5	4	3	2	1	0
1	Pertinencia de indicadores	X					
2	Formulado con lenguaje apropiado	X					
3	Adecuado para los sujetos en estudio	X					
4	Facilita la prueba de hipótesis	X					
5	Suficiencia para medir la variable	X					
6	Facilita la interpretación del instrumento	X					
7	Acorde al avance de la ciencia y tecnología		X				
8	Expresado en hechos perceptibles	X					
9	Tiene secuencia lógica	X					
10	Basado en aspectos teóricos		X				
	Total	4	8				
		0					

Coeficiente de valoración porcentual: $c = 96\%$

III. OBSERVACIONES Y/O RECOMENDACIONES



.....
Firma y sello del Experto

Fig. 127: Ficha 2 de validación para el cuestionario (variable dependiente)

ANEXO 8: ALFA DE CRONBACH PARA EL CUESTIONARIO 1

En las siguientes tablas se muestra el análisis de fiabilidad para el cuestionario de la variable independiente, obteniendo un Alfa de Cronbach de 0.801.

Tabla XXXV: Resumen de procesamiento de casos para cuestionario 1

Resumen de procesamiento de casos		N	%
Casos	Válido	5	100.0
	Excluido	0	.0
	Total	5	100.0

Tabla XXXVI: Estadística de fiabilidad para cuestionario 1

Estadística de fiabilidad	
Alfa de Cronbach	Nº de elementos
0.801	19

Tabla XXXVII: Estadísticas de elementos para cuestionario 1

Estadísticas de elementos			
	Media	Desviación estándar	N
Pregunta 1	4.60	.548	5
Pregunta 2	4.20	.447	5
Pregunta 3	4.60	.548	5
Pregunta 4	4.40	.548	5
Pregunta 5	4.60	.548	5
Pregunta 6	4.40	.548	5
Pregunta 7	4.60	.548	5
Pregunta 8	4.00	.707	5
Pregunta 9	4.40	.548	5
Pregunta 10	4.00	.707	5
Pregunta 11	4.60	.548	5
Pregunta 12	4.80	.447	5
Pregunta 13	4.60	.548	5
Pregunta 14	4.00	.707	5
Pregunta 15	4.80	.447	5
Pregunta 16	4.40	.548	5
Pregunta 17	4.00	.707	5
Pregunta 18	4.40	.548	5
Pregunta 19	4.40	.548	5

ANEXO 9: ALFA DE CRONBACH PARA EL CUESTIONARIO 2

En las siguientes tablas se muestra el análisis de fiabilidad para el cuestionario de la variable dependiente, obteniendo un Alfa de Cronbach de 0.814.

Tabla XXXVIII: Resumen de procesamiento de casos para cuestionario 2

Resumen de procesamiento de casos		N	%
Casos	Válido	5	100.0
	Excluido	0	.0
	Total	5	100.0

Tabla XXXIX: Estadística de fiabilidad para cuestionario 2

Estadística de fiabilidad	
Alfa de Cronbach	Nº de elementos
.814	22

Tabla XL: Estadística de elementos para cuestionario 2

Estadísticas de elementos			
	Media	Desviación estándar	N
Pregunta 1	1.80	.447	5
Pregunta 2	3.40	.894	5
Pregunta 3	2.60	1.140	5
Pregunta 4	3.80	1.304	5
Pregunta 5	3.80	1.304	5
Pregunta 6	2.80	1.483	5
Pregunta 7	2.80	1.095	5
Pregunta 8	2.40	1.140	5
Pregunta 9	2.40	1.949	5
Pregunta 10	2.00	.707	5
Pregunta 11	2.80	1.304	5
Pregunta 12	2.20	1.789	5
Pregunta 13	3.20	1.789	5
Pregunta 14	2.40	1.342	5
Pregunta 15	3.00	1.000	5
Pregunta 16	3.80	1.304	5
Pregunta 17	3.00	1.871	5
Pregunta 18	3.60	1.140	5
Pregunta 19	3.00	1.000	5
Pregunta 20	2.20	1.643	5
Pregunta 21	2.20	.837	5
Pregunta 22	4.00	1.414	5

ANEXO 10: FICHAS DE VALIDACIÓN DE FICHA DE OBSERVACIÓN

FICHA PARA VALIDACIÓN DEL INSTRUMENTO

I. REFERENCIA

- 1.1. Experto: Néstor Elías Muñoz Abanto
- 1.2. Especialidad:
- 1.3. Cargo actual: Docente
- 1.4. Grado académico: Ingeniero de Sistemas
- 1.5. Institución: UNC
- 1.6. Tipo de instrumento: Ficha de observación
- 1.7. Lugar y fecha: 02-06-2025

II. TABLA DE VALORACIÓN POR EVIDENCIAS

Nº	EVIDENCIAS	VALORACION					
		5	4	3	2	1	0
1	Pertinencia de indicadores		x				
2	Formulado con lenguaje apropiado	x					
3	Adecuado para los sujetos en estudio		x				
4	Facilita la prueba de hipótesis			x			
5	Suficiencia para medir la variable		x				
6	Facilita la interpretación del instrumento		x				
7	Acorde al avance de la ciencia y tecnología	x					
8	Expresado en hechos perceptibles		x				
9	Tiene secuencia lógica		x				
10	Basado en aspectos teóricos		x				
	Total	1	2	3			
		0	8				

Coeficiente de valoración porcentual: $c = 82\%$

III. OBSERVACIONES Y/O RECOMENDACIONES

Incluir definición de actividades a observar: Define qué se entiende por "Actividad 1", "Actividad 2", etc. Esto evita interpretaciones subjetivas.

Estandarizar la tabla con columnas como: nombre del trabajador, tiempo por actividad, tiempo total, observaciones.

Respecto a la herramienta: cronómetro, indicar que utiliza herramienta de apoyo el cronómetro (para registrar el tiempo con precisión)

Observaciones levantadas.



Néstor Elías Muñoz Abanto

Firma y sello del Experto

Fig. 128: Ficha 1 de validación para ficha de observación

FICHA PARA VALIDACIÓN DEL INSTRUMENTO

I. REFERENCIA

- 1.1. Experto: Ing. Lisi Janet Vásquez Fernández
- 1.2. Especialidad: Ingeniería de Sistemas
- 1.3. Cargo actual: Docente Nombrada.
- 1.4. Grado académico: Doctor en Ciencias
- 1.5. Institución: Universidad Nacional de Cajamarca
- 1.6. Tipo de instrumento: Ficha de observación
- 1.7. Lugar y fecha: 13-06-2025

II. TABLA DE VALORACIÓN POR EVIDENCIAS

Nº	EVIDENCIAS	VALORACION					
		5	4	3	2	1	0
1	Pertinencia de indicadores	X					
2	Formulado con lenguaje apropiado	X					
3	Adecuado para los sujetos en estudio	X					
4	Facilita la prueba de hipótesis		X				
5	Suficiencia para medir la variable		X				
6	Facilita la interpretación del instrumento	X					
7	Acorde al avance de la ciencia y tecnología		X				
8	Expresado en hechos perceptibles	X					
9	Tiene secuencia lógica	X					
10	Basado en aspectos teóricos		X				
	Total	3	1				
		0	6				

Coeficiente de valoración porcentual: c = 92%

III. OBSERVACIONES Y/O RECOMENDACIONES


.....
Firma y sello del Experto

Fig. 129: Ficha 2 de validación para ficha de observación

ANEXO 11: NÚMERO DE LECTURAS SEGÚN LA DURACIÓN DEL CICLO

Tiempo del ciclo u operación (minutos)	Número de ciclos a cronometrar
Hasta 0,10	200
0,25 – 0,50	100
0,50 – 0,75	60
0,75 – 1,00	40
1,00 – 2,00	30
2,00 – 4,00	20
4,00 – 5,00	15
5,00 – 10,00	10
10,00 – 20,00	8
20,00 – 40,00	5
40,00 o más	3

Fig. 130: Número de lecturas según duración del ciclo

ANEXO 12: PRUEBAS DE NORMALIDAD PARA LA DIMENSIÓN TIEMPO

Pruebas de normalidad

	Kolmogorov-Smirnov ^a			Shapiro-Wilk		
	Estadístico	gl	Sig.	Estadístico	gl	Sig.
TT_Registro_pretest	,101	20	,200*	,981	20	,943
TT_Registro_postest	,095	20	,200*	,986	20	,988

*. Esto es un límite inferior de la significación verdadera.

a. Corrección de significación de Lilliefors

Fig. 131: Prueba de normalidad para registro de pedido

Pruebas de normalidad

	Kolmogorov-Smirnov ^a			Shapiro-Wilk		
	Estadístico	gl	Sig.	Estadístico	gl	Sig.
TT_Reporte_pretest	,159	8	,200*	,948	8	,687
TT_Reporte_postest	,217	8	,200*	,867	8	,141

*. Esto es un límite inferior de la significación verdadera.

a. Corrección de significación de Lilliefors

Fig. 132: Prueba de normalidad para reporte de ventas

Pruebas de normalidad

	Kolmogorov-Smirnov ^a			Shapiro-Wilk		
	Estadístico	gl	Sig.	Estadístico	gl	Sig.
RegistroPagoPretest	,150	20	,200*	,956	20	,476
RegistroPagoPostest	,160	20	,194	,942	20	,261

*. Esto es un límite inferior de la significación verdadera.

a. Corrección de significación de Lilliefors

Fig. 133: Prueba de normalidad para registro de pago

ANEXO 13: PRUEBAS DE NORMALIDAD PARA LA DIMENSIÓN SATISFACCIÓN

Pruebas de normalidad

	Kolmogorov-Smirnov ^a			Shapiro-Wilk		
	Estadístico	gl	Sig.	Estadístico	gl	Sig.
satisfaccion_i1_pre	,261	5	,200*	,859	5	,223
satisfaccion_i1_post	,197	5	,200*	,934	5	,627
satisfaccion_i2_pre	,198	5	,200*	,957	5	,787
satisfaccion_i2_post	,241	5	,200*	,821	5	,119
satisfaccion_i3_pre	,221	5	,200*	,902	5	,421
satisfaccion_i3_post	,372	5	,022	,828	5	,135
satisfaccion_i4_pre	,372	5	,022	,828	5	,135
satisfaccion_i4_post	,261	5	,200*	,859	5	,223

*. Esto es un límite inferior de la significación verdadera.

a. Corrección de significación de Lilliefors

Fig. 134: Prueba de normalidad para satisfacción

ANEXO 14: PRUEBAS DE NORMALIDAD PARA LA DIMENSIÓN CALIDAD

Pruebas de normalidad

	Kolmogorov-Smirnov ^a			Shapiro-Wilk		
	Estadístico	gl	Sig.	Estadístico	gl	Sig.
Calidad_i1_pre	,241	5	,200*	,821	5	,119
Calidad_i1_post	,354	5	,040	,777	5	,052
Calidad_i2_pre	,221	5	,200*	,902	5	,421
Calidad_i2_post	,330	5	,079	,735	5	,021

*. Esto es un límite inferior de la significación verdadera.

a. Corrección de significación de Lilliefors

Fig. 135: Prueba de normalidad para calidad

ANEXO 15: PRUEBAS DE ACEPTACIÓN DE USUARIO

Tabla XLI: Pruebas de aceptación de usuario

Ficha de evaluación para pruebas de aceptación de requerimientos funcionales				
Finalidad: Evaluar y determinar el porcentaje de cumplimiento de la aplicación web y móvil teniendo en cuenta los requerimientos funcionales solicitados por la empresa NikosPizza.				
Nombres: Nicanor Quiroz Minchan		DNI: 41707546		
Nº	Requerimiento funcional			% de cumplimiento
			50%	75%
1	La aplicación debe permitir visualizar gráficos sobre las ventas mensuales, y un gráfico sobre las ventas diarias y que sea de rápida visualización			X
2	La aplicación debe permitir analizar la información del panel principal utilizando herramientas de inteligencia artificial cuando el usuario lo requiera.			X
3	La aplicación debe permitir mostrar la lista de productos más vendidos y visualizar el detalle de cada producto seleccionado			X
4	La aplicación debe permitir gestionar categorías de productos indicando un nombre obligatorio y una descripción opcional.			X
5	La aplicación debe permitir gestionar clientes indicando, como mínimo, su nombre, número de DNI y fecha de nacimiento, para identificar a los clientes que cumplen años y ofrecerles promociones especiales.			X
6	La aplicación debe permitir gestionar proveedores de productos complementarios indicando obligatoriamente su nombre y dirección			X
7	La aplicación debe permitir gestionar productos y poder cambiar el estado del producto entre activo o desactivado de forma rápida			X
8	La aplicación debe permitir gestionar ventas con la opción de asociar o no un cliente, aplicar descuentos, identificar si corresponde a una mesa o delivery, y generar un comprobante en formato PDF			X

9	La aplicación debe permitir gestionar premios indicando su nombre, descripción y cantidad de compras necesarias, generando alertas cuando un cliente alcance o esté próximo a alcanzar un premio			X
10	La aplicación debe permitir generar reportes de ventas diarias mostrando el total de ingresos, número de ventas, ventas realizadas en mesa y/o delivery, ventas pagadas en yape y/o efectivo y pedidos cancelados, con opciones de filtrado por producto, vendedor y cliente			X
11	La aplicación debe permitir generar reportes de ventas por rango de fechas con opciones de filtrado por producto, vendedor o cliente			X
12	La aplicación debe permitir gestionar reclamos clasificándolos por estado (pendiente, en atención o atendido) y prioridad (baja, media o alta).		X	
13	La aplicación debe permitir registrar y actualizar los datos generales de la empresa			X
14	La aplicación debe permitir modificar el umbral mínimo de ventas diarias mostrado en el panel principal			X
15	La aplicación debe permitir controlar el acceso a las diferentes secciones de la aplicación			X
16	La aplicación debe permitir visualizar y modificar la información personal de la cuenta de usuario		X	
17	La aplicación debe permitir utilizar las mismas funcionalidades de la versión web en la aplicación móvil y acceder a esta desde la Google Play Store			X

ANEXO 16: CAPEX Y OPEX

CAPEX: Gastos de capital (inversión inicial)

Los gastos de capital son las inversiones a largo plazo en activos que generarán beneficios futuros. Para tu sistema, la inversión inicial de desarrollo se considera un activo, ya que crea un valor a largo plazo.

Tabla XLII: Capex y opex

Concepto	Descripción	Monto	Comentarios
Desarrollo del software	Costo del tiempo y esfuerzo para crear el software a medida.	Valor a determinar	Si lo desarrollaste tú mismo, el costo es tu tiempo invertido. Si contrataste a un tercero, este sería el monto pagado. Se considera una inversión que tiene un periodo de amortización.
Adquisición de hardware	Compra de la laptop para la caja.	Valor a determinar	Se considera un activo que se deprecia con el tiempo. Si ya la tenías, su valor actual debe considerarse en el cálculo.
Licencias de software permanentes	En tu caso, al usar herramientas gratuitas (María DB, Laravel), este costo es cero.	\$0	N/A
Total, de CAPEX	Suma de los gastos de capital.	(Desarrollo + Laptop)	Se debe contabilizar el gasto inicial del desarrollo del software y la compra de la laptop para la caja.

OPEX: Gastos operativos (costos recurrentes)

Los gastos operativos son los costos continuos y recurrentes necesarios para mantener el negocio en marcha. Para tu sistema, estos son los gastos mensuales o anuales asociados al uso y mantenimiento.

Concepto	Descripción	Periodicidad	Monto	Comentarios
Servidor VPS	Alquiler anual del servidor para alojar el sistema.	Anual	\$50	Se distribuye mensualmente en los gastos (\$4.17/mes).
Dominio web	Al usar un dominio gratuito, este costo es cero.	Anual	\$0	Si en el futuro adquieres un dominio propio, se convertiría en un OPEX anual.
Energía y mantenimiento	Consumo eléctrico y posibles mantenimientos de la laptop.	Mensual	Estimado	Dependerá del uso, pero es un gasto operativo recurrente.
Conexión a internet	Costo del servicio de internet del negocio, necesario para el sistema.	Mensual	Estimado	Es un gasto recurrente vital para la operación.
Soporte y mantenimiento	Costo de posibles correcciones de errores, actualizaciones y soporte técnico.	Puntual	Valor a determinar	Si lo haces tú, es tiempo no cobrado. Si necesitas ayuda externa, podría ser un gasto puntual o un contrato de mantenimiento mensual.
Recursos humanos	Salarios de los mozos y el cajero.	Mensual	Estimado	Aunque no es un gasto del software, el sistema habilita el trabajo de estas personas, por lo que es un gasto operativo del negocio.
Total, de OPEX	Suma de los gastos recurrentes.	Mensual o Anual	(VPS + Mantenimiento +	Tu OPEX mensual es bastante bajo, lo que indica

Concepto	Descripción	Periodicidad	Monto	Comentarios
			Energía + Internet + etc.)	un sistema eficiente en costos.

Análisis y conclusiones

Eficiencia de costos: El modelo de negocio es muy eficiente en costos. Al desarrollar el software internamente y utilizar herramientas gratuitas y un dominio sin costo, se ha minimizado el CAPEX y el OPEX.

Modelo híbrido: El modelo de inversión híbrido. El desarrollo del software y la laptop son CAPEX (inversión inicial), mientras que el VPS es un OPEX (gasto recurrente).

Viabilidad financiera: El bajo costo recurrente del sistema lo hace muy viable para un negocio pequeño. El gasto más significativo está en el desarrollo inicial (CAPEX), pero una vez amortizado, los costos de mantenimiento (OPEX) son mínimos.

Escalabilidad a futuro: Aunque actualmente el VPS es económico, si el negocio crece y el tráfico aumenta, los costos de alojamiento (OPEX) podrían incrementarse.

Riesgos minimizados: Al usar software de código abierto y herramientas gratuitas, se ha evitado los altos costos de licencias comerciales. El principal riesgo está en el soporte, que depende del conocimiento o de la disponibilidad de un desarrollador externo.

ANEXO 17: CASOS DE USO DE LA APLICACIÓN WEB Y MÓVIL

Caso de uso gestionar usuarios

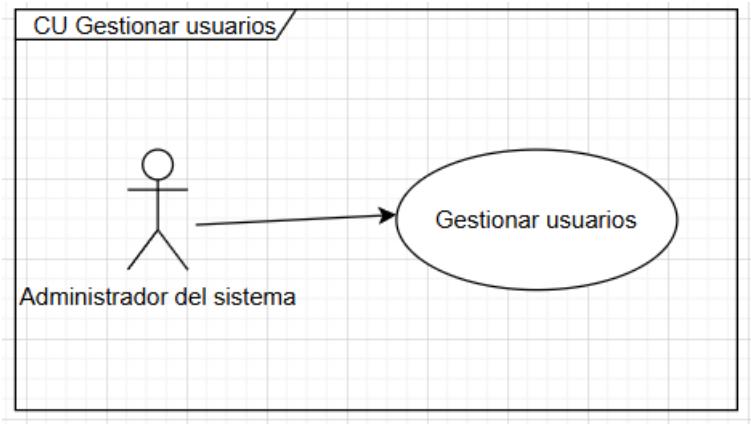


Fig. 136: Caso de uso gestionar usuarios

Especificación de caso de uso gestionar usuarios.

Tabla XLIII: Especificación de caso de uso gestionar usuarios

Nombre	Gestionar usuarios
Autor	Tafur Valdivia, Roberto Jhuliño
Fecha	27/04/2025
Descripción	Este caso de uso permite gestionar los usuarios dentro de la aplicación web y móvil, cada usuario por independiente debe crear su cuenta, luego el administrador se encarga de asignar los roles y permisos.
Actores	Actor principal: El administrador del sistema. Actor secundario: El trabajador.
Precondiciones	El actor debe haber iniciado sesión en la aplicación web y móvil. El actor secundario debe haber creado su cuenta en la aplicación mediante el proceso de registro.
Flujo normal	El actor principal se ubica en el panel de control y selecciona usuarios. La aplicación muestra una lista con todos los usuarios y por cada usuario dos opciones de editar y eliminar. El actor selecciona una opción: <ul style="list-style-type: none">- Editar el usuario: ver subflujo1- Eliminar el usuario: ver subflujo2
Subflujos	Subflujo1:

	<ul style="list-style-type: none"> - La aplicación muestra una interfaz con el nombre del usuario y la lista de roles disponibles. - El actor principal selecciona el rol apropiado para el usuario. - Al presionar “asignar rol”, la aplicación guarda los cambios y redirige a la lista de usuarios. <p>Subflujo2:</p> <ul style="list-style-type: none"> - El actor principal selecciona la opción “eliminar” para un usuario específico. - La aplicación muestra una alerta de confirmación. - Al confirmar, la aplicación procede con la eliminación del usuario.
Flujo alternativo	<ul style="list-style-type: none"> - Si el administrador cancela la alerta de confirmación, no se realiza ninguna acción y se mantiene la lista de usuarios. - Si no se selecciona ningún rol antes de presionar "Asignar rol", el sistema muestra un mensaje solicitando seleccionar una opción válida
Post condiciones	<ul style="list-style-type: none"> - Los roles asignados se reflejan inmediatamente en los permisos del usuario. - Los usuarios eliminados pierden acceso al sistema de forma permanente

Caso de uso registrar reclamos

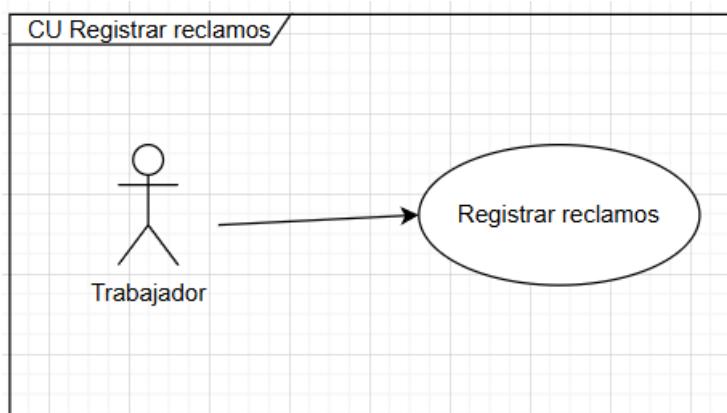


Fig. 137: Caso de uso registrar reclamos

Especificación de caso de uso registrar reclamos.

Tabla XLIV: Especificación de caso de uso registrar reclamos

Nombre	Registrar reclamos
Autor	Tafur Valdivia, Roberto Jhuliño
Fecha	01/05/2025
Descripción	Este caso de uso permite gestionar los reclamos de los clientes, ya sea relacionados con un pedido específico o de carácter general, incluyendo su registro, edición, eliminación y cambio de estado.
Actores	Actor principal: Vendedor (trabajador).
Precondiciones	El actor debe haber iniciado sesión en la aplicación web y móvil. El actor debe tener permisos para gestionar reclamos.
Flujo normal	<p>El actor principal se ubica en el panel de control y selecciona “reclamos usuarios”.</p> <p>La aplicación muestra una lista con todos los reclamos anteriores, un buscador para buscar un reclamo en específico y un botón de agregar un nuevo reclamo:</p> <p>El vendedor seleccionar una opción:</p> <ul style="list-style-type: none"> - Si elije registrar un nuevo reclamo: ver subflujo1 - Si elije editar un reclamo: ver subflujo2 - Si elige eliminar el usuario: ver subflujo3
Subflujos	<p>Subflujo1:</p> <ul style="list-style-type: none"> - La aplicación muestra formulario con campos: nombre del reclamante, título, descripción, prioridad y código de pedido (opcional). - El vendedor completa la información requerida. - Al presionar "Registrar", el sistema guarda el reclamo y redirige a la lista con mensaje de confirmación. <p>Subflujo2:</p> <ul style="list-style-type: none"> - El vendedor localiza el reclamo mediante el buscador (por usuario o título). - Selecciona "Editar" y el sistema muestra el formulario con datos actuales. - Realiza las modificaciones necesarias y presiona "Actualizar".

	<ul style="list-style-type: none"> - La aplicación guarda los cambios y redirige con mensaje de confirmación <p>Subflujo3:</p> <ul style="list-style-type: none"> - El vendedor localiza el reclamo y selecciona "Eliminar". - La aplicación muestra alerta de confirmación. - Al confirmar, elimina el reclamo y muestra mensaje de éxito.
Flujo alternativo	<ul style="list-style-type: none"> - En la lista de reclamos, el vendedor puede hacer clic directamente en el botón de estado. - La aplicación cicla automáticamente entre los estados: "Pendiente" → "En Proceso" → "Solucionado" → "Pendiente". - El cambio se aplica inmediatamente sin requerir edición completa del reclamo
Post condiciones	<ul style="list-style-type: none"> - Los reclamos registrados quedan disponibles para consulta y seguimiento. - Los cambios de estado se reflejan en tiempo real en el sistema. - Los reclamos eliminados se remueven permanentemente de la base de datos. - Se mantiene la integridad de la información relacionada con pedidos existentes.

Caso de uso registrar productos

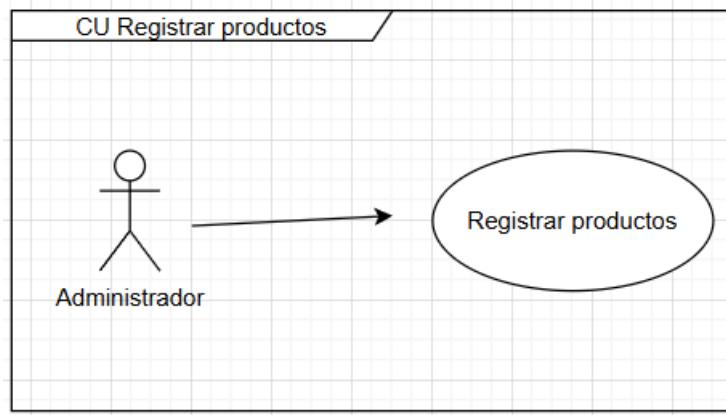


Fig. 138: Caso de uso registrar productos

Especificaciones del caso de uso registrar producto

Tabla XLV: Especificación de caso de uso registrar producto

Nombre	Registrar producto
Autor	Tafur Valdivia, Roberto Jhuliño
Fecha	03/05/2025
Descripción	Este caso de uso permite gestionar el catálogo de productos, incluyendo su registro, modificación, eliminación y control de estado dentro de la aplicación.
Actores	Actor principal: El administrador.
Precondiciones	El actor debe haber iniciado sesión en la aplicación web y móvil. Deben existir categorías registradas para asociar los productos.
Flujo normal	<p>El actor principal se ubica en el panel de control y selecciona productos. La aplicación muestra una lista con todos los productos ya registrados, un buscador para buscar un producto y un botón de agregar un nuevo producto.</p> <p>El administrador selecciona una opción:</p> <ul style="list-style-type: none"> - Si elige registrar un nuevo producto: ver subflujo1 - Si elige editar un producto: ver subflujo2 - Si elige eliminar un producto: ver subflujo3
Subflujos	<p>Subflujo1:</p> <ul style="list-style-type: none"> - La aplicación muestra el formulario con campos: nombre, código, stock, precio de venta, categoría, proveedor, imagen, etc. - El administrador completa los datos requeridos. - Al presionar "Registrar", la aplicación guarda el producto y redirige a la lista con mensaje "Producto creado correctamente" <p>Subflujo2:</p> <ul style="list-style-type: none"> - El administrador localiza el producto mediante el buscador (por nombre). - Selecciona "Editar" y la aplicación muestra el formulario con datos actuales. - Realiza las modificaciones necesarias y presiona "Actualizar". - La aplicación guarda los cambios y redirige con mensaje "Producto actualizado correctamente". <p>Subflujo3:</p>

	<ul style="list-style-type: none"> - El administrador localiza el producto y selecciona "Eliminar". - La aplicación muestra alerta de confirmación. - Al confirmar, elimina el producto y muestra mensaje "Producto eliminado correctamente"
Flujo alternativo	<ul style="list-style-type: none"> - En la lista de productos, el administrador puede hacer clic directamente en el indicador de estado. - La aplicación alterna entre "Activo" y "Desactivado" con cada clic. - El cambio se aplica inmediatamente sin requerir edición completa del producto
Post condiciones	<ul style="list-style-type: none"> - Los productos registrados quedan disponibles para ser utilizados en ventas y compras. - Los productos desactivados no aparecen disponibles para nuevas ventas.