

**UNIVERSIDAD NACIONAL MICAELA BASTIDAS DE APURIMAC**

**-FACULTAD DE INGENIERÍA**

**ESCUELA ACADÉMICO PROFESIONAL DE INGENIERÍA INFORMÁTICA Y  
SISTEMAS**



## **DOCUMENTACIÓN DE INGENIERÍA DE SOFTWARE**

**“Digitalización del menú de la cafetería Polaris”**

**CURSO:** Ingeniería de software I

**GRUPO:** 3 – “Blaze Software”

**ESTUDIANTES:**

- Jennifer Castillo Castillo
- Jean Marcos Bellido Molina.
- Alipio Chuyma Vargas
- Jean Sharry Escalante Benites
- Alex Yujans Lopinta Soel

**DOCENTE:** Julio Cesar Lloclli Champi

## INDICE DE CONTENIDO

<b>I.</b>	<b>INTRODUCCION .....</b>	<b>7</b>
<b>II.</b>	<b>MISION Y VISION DE LA EMPRESA .....</b>	<b>9</b>
2.1.	Misión.....	9
2.2.	Visión .....	9
2.3.	Preguntas de Investigación: .....	9
<b>III.</b>	<b>PLANTEAMIENTO DEL PROBLEMA .....</b>	<b>10</b>
3.1.	Definición.....	10
3.2.	Descripción e impacto del problema .....	10
3.3.	Objetivos del proyecto .....	11
3.3.1.	Objetivo general.....	12
3.3.2.	Objetivos específicos .....	12
<b>IV.</b>	<b>JUSTIFICACION DEL PROYECTO .....</b>	<b>13</b>
4.1.	Beneficios esperados .....	13
4.2.	Beneficiarios del proyecto .....	14
<b>V.</b>	<b>ALCANCE DEL PROYECTO .....</b>	<b>16</b>
5.1.	Incluye.....	16
5.2.	No incluye .....	18
<b>VI.</b>	<b>MODELO.....</b>	<b>19</b>
6.1.	Análisis .....	19
6.1.1.	Introducción.....	20
6.1.1.1.	Contexto .....	20
6.1.1.2.	Propósito .....	21

6.1.1.3.	Alcance .....	22
6.1.1.4.	definiciones .....	23
<b>6.1.2.</b>	<b>Descripción general .....</b>	<b>24</b>
6.1.2.1.	Perspectiva del producto.....	24
6.1.2.2.	Funcionalidades .....	25
6.1.2.3.	Características del usuario .....	26
6.1.2.4.	Cuestionario de requerimientos de software .....	27
<b>6.1.3.</b>	<b>Requisitos específicos.....</b>	<b>29</b>
6.1.3.1.	Requisitos funcionales .....	29
6.1.3.2.	Requisitos no funcionales .....	31
<b>6.1.4.</b>	<b>Modelo de negocios.....</b>	<b>32</b>
6.1.4.1.	Diagrama de flujo de procesos .....	33
6.1.4.2.	Diagrama de casos de uso (UML).....	34
<b>6.1.5.</b>	<b>Diseño de base de datos.....</b>	<b>36</b>
6.1.5.1.	Modelo entidad-relación (ER).....	37
6.1.5.2.	Modelo lógico.....	38
6.1.5.3.	Modelo físico.....	41
6.1.5.4.	Diccionario de base de datos.....	42
<b>6.1.6.</b>	<b>Estudio de factibilidad .....</b>	<b>48</b>
6.1.6.1.	Factibilidad técnica.....	48
6.1.6.2.	Costos de infraestructura .....	49
6.1.6.3.	Costos iniciales de implementación .....	51
6.1.6.4.	Costos de mantenimiento y soporte.....	53

6.1.6.5.	Total estimado de costos de software .....	55
<b>6.1.7.</b>	<b>Indicadores financieros.....</b>	<b>58</b>
6.1.7.1.	Cálculo del Valor Actual Neto (VAN) .....	59
<b>6.2.</b>	<b>Fase de diseño .....</b>	<b>61</b>
6.2.1.	Diseño de interfaz de usuario .....	62
6.2.1.1.	Interfaz del modulo cliente .....	62
6.2.1.2.	Interfaz del panel del empleado .....	64
<b>6.3.</b>	<b>Fase de implementación.....</b>	<b>66</b>
6.3.1.	Enfoque metodológico en la implementación .....	67
6.3.2.	Organización en Sprints (SCRUM) .....	68
6.3.3.	Tecnologías y decisiones de implementación principales.....	71
6.3.4.	Entregables principales de la fase.....	73
6.3.5.	Retos enfrentados y soluciones adoptadas.....	74
<b>VII.</b>	<b>CRONOGRAMA DEL PROYECTO .....</b>	<b>76</b>
<b>VIII.</b>	<b>CONCLUSION .....</b>	<b>79</b>
	Respuestas a las Preguntas de Investigación .....	81
<b>IX.</b>	<b>ANEXOS .....</b>	<b>83</b>
9.1.	Links de acceso .....	83
9.2.	Imágenes.....	83
<b>X.</b>	<b>BIBLIOGRAFIA .....</b>	<b>88</b>

## ÍNDICE DE TABLAS

Tabla 1: Entidad empleado .....	43
Tabla 2: Entidad categoría .....	43
Tabla 3: Entidad producto.....	44
Tabla 4: Entidad acompañamiento.....	45
Tabla 5: Entidad productoAcompañamiento .....	45
Tabla 6: Entidad pedido .....	46
Tabla 7: Entidad pedidoItem.....	47
Tabla 8: Comparativo de Alternativas de Local de Trabajo y Costos Mensuales .....	50
Tabla 9: Resumen de Costos Iniciales de Implementación .....	52
Tabla 10: Estimación de Costos de Mantenimiento y Soporte (Anual).....	55
Tabla 11: Resumen Consolidado de Costos Totales del Proyecto.....	56
Tabla 12: Resumen de Indicadores Financieros .....	61
Tabla 13: Resumen de Sprints y Entregables .....	70
Tabla 14: Resumen de Fases y Duración Principal .....	77

## ÍNDICE DE FIGURAS

Figura 1: Diagrama de flujo del proceso actual de atención .....	33
Figura 2: Diagrama de Flujo del Proceso Propuesto con el Sistema .....	34
Figura 3: Diagrama de Casos de Uso del Sistema Polaris .....	36
Figura 4: Diagrama Entidad-Relación .....	38
Figura 5: Diagrama del Modelo Lógico .....	41
Figura 6: Diagrama del Modelo Físico .....	42
Figura 7: Prototipos de Interfaz – Módulo Cliente .....	64
Figura 8: Prototipos de Interfaz – Panel del Empleado .....	66

Figura 9: Organigrama del proyecto .....	68
Figura 10: Diagrama de Gantt - Cronograma general del proyecto 1 .....	77
Figura 11: Diagrama de Gantt - Cronograma general del proyecto 2 .....	78
Figura 12: Diagrama de Gantt - Cronograma general del proyecto 3 .....	78
Figura 13: Reunion Meet .....	83
Figura 14: Daily Scrum (Whatsapp) .....	84
Figura 15: Firma del Cliente .....	84
Figura 16: Reunion con el Cliente .....	85
Figura 17: Logo del equipo .....	85
Figura 18: QR de acceso .....	86
Figura 19: Reunion en biblioteca .....	86
Figura 20: Reunion en biblioteca .....	87

## **I. INTRODUCCION**

La presente documentación detalla el proyecto de ingeniería de software titulado "Digitalización del Menú de la Cafetería Polaris", desarrollado en el marco del curso de Ingeniería de Software I de la Escuela Académico Profesional de Ingeniería Informática y Sistemas de la Facultad de Ingeniería de la Universidad Nacional Micaela Bastidas de Apurímac (UNAMBA). Este proyecto surge como respuesta a las ineficiencias operativas observadas en la cafetería Polaris, ubicada en Abancay, Perú, un establecimiento enfocado en bebidas saludables y refrescantes dirigidas principalmente a la comunidad universitaria.

En un contexto donde la industria gastronómica enfrenta desafíos como la escasez de personal calificado, la alta rotación laboral y la demanda creciente de servicios eficientes y digitales, la implementación de soluciones tecnológicas se posiciona como una estrategia clave para optimizar procesos y mejorar la experiencia del usuario. La cafetería Polaris, como punto de referencia en el campus de la UNAMBA, experimenta problemas operativos derivados de métodos tradicionales de atención, tales como la toma manual de pedidos, la falta de trazabilidad en tiempo real y la limitada personalización del menú. Estos factores generan cuellos de botella que impactan en la satisfacción del cliente, la productividad del equipo y la rentabilidad general del negocio.

El objetivo principal de este proyecto es diseñar e implementar un sistema web integral que automatice el proceso de pedidos mediante códigos QR, permitiendo a los clientes acceder a un menú digital interactivo desde dispositivos móviles, personalizar productos y enviar órdenes directamente al panel de gestión de empleados. Este sistema no solo resuelve ineficiencias inmediatas, sino que alinea las operaciones de la cafetería con principios de innovación y sostenibilidad, contribuyendo a su posicionamiento como líder local en el mercado de bebidas saludables.

El desarrollo se basa en una metodología híbrida: el modelo en cascada para la planificación estructurada, análisis de requisitos, cronograma, estructura de desglose de trabajo (EDT), casos de uso y cierre del proyecto; y el modelo Scrum para la fase de implementación del software, facilitando sprints iterativos con retroalimentación continua. Las tecnologías empleadas incluyen MySQL para la base de datos, Angular para el frontend y Spring Boot para el backend, asegurando un sistema escalable, seguro y responsivo.

Esta documentación se estructura de manera secuencial, abarcando desde la misión y visión de la empresa hasta el cronograma, conclusiones y anexos, con énfasis en la justificación técnica, operativa y económica del software. Se sustenta en entrevistas con la dueña y empleados de la cafetería, así como en un análisis de factibilidad que demuestra la viabilidad del proyecto, incluyendo indicadores financieros como el Valor Actual Neto (VAN), la Tasa Interna de Retorno (TIR) y el Retorno de la Inversión (ROI), junto con el período de recuperación de la inversión.



## **II. MISIÓN Y VISIÓN DE LA EMPRESA**

### **2.1. Misión**

Proporcionar bebidas refrescantes y deliciosas, elaboradas con ingredientes saludables, enfocados en la sostenibilidad e innovación

### **2.2. Visión**

Para el 2027, ser empresa líder en el mercado de frappés saludables e innovadores en la ciudad de Abancay, reconocida por la calidad excepcional de nuestros productos y de nuestro firme compromiso con la sostenibilidad.

### **2.3. Preguntas de Investigación:**

- ¿Cómo el Sistema Polaris establece un liderazgo en innovación tecnológica para 2027, transformando la experiencia del cliente a través del acceso digital y la automatización?
- ¿De qué manera la personalización detallada en el menú digital diferencia la propuesta de "frappés saludables e innovadores" de la competencia tradicional?
- ¿Cómo la automatización del flujo de pedidos mejora la percepción de calidad del servicio al optimizar la operación y reducir errores y esperas?
- ¿Qué ventaja competitiva proporcionan los datos del sistema para la toma de decisiones estratégicas alineadas con la innovación y sostenibilidad?

### **III. PLANTEAMIENTO DEL PROBLEMA**

#### **3.1. Definición**

El problema central que aborda este proyecto radica en las ineficiencias operativas derivadas de los métodos tradicionales de atención al cliente en la cafetería Polaris, ubicada en el campus de la Universidad Nacional Micaela Bastidas de Apurímac (UNAMBA) en Abancay. Estos métodos, basados en la toma manual de pedidos mediante papel y comunicación verbal directa, generan cuellos de botella críticos que afectan la calidad del servicio, la satisfacción del usuario y la rentabilidad del establecimiento. En un entorno universitario con alta afluencia de clientes (principalmente estudiantes con horarios ajustados y expectativas digitales crecientes), dichas limitaciones se traducen en tiempos de espera prolongados, errores en la personalización de productos y falta de trazabilidad en tiempo real de los pedidos.

#### **3.2. Descripción e impacto del problema**

A nivel global, el sector de la restauración y las cafeterías enfrenta una crisis estructural marcada por la escasez crónica de personal calificado, agravada por la alta rotación laboral, el absentismo y las presiones derivadas de la post-pandemia. Según informes recientes del sector Horeca en Europa y tendencias internacionales para 2025, la falta de mano de obra cualificada constituye una de las principales preocupaciones, con cadenas y establecimientos independientes ralentizando aperturas o ajustando horarios debido a esta limitación. Esta escasez no se resuelve únicamente con incrementos salariales, sino que impulsa la adopción de tecnologías digitales para compensar la reducción de personal en sala, optimizando procesos y liberando recursos para tareas de mayor valor.

En el contexto nacional peruano, el fenómeno se replica con particular intensidad en el sector gastronómico y de cafeterías, donde la alta rotación y la demanda de eficiencia operativa se combinan con la creciente expectativa de los consumidores por experiencias rápidas y contactless. La gastronomía peruana, reconocida mundialmente, enfrenta el reto de la

digitalización para mantener su competitividad: herramientas como menús digitales con códigos QR y sistemas de pedidos automáticos están transformando la experiencia en restaurantes y cafeterías, permitiendo reducir tiempos de espera, minimizar errores humanos y mejorar la eficiencia operativa. Estudios y tendencias para 2025 destacan que la implementación de estas soluciones no solo responde a demandas post-COVID de higiene y rapidez, sino que también contribuye a la sostenibilidad financiera al aumentar la rotación de mesas y atraer a clientes más jóvenes y digitales.

A nivel local, en cafeterías universitarias como Polaris en Abancay, el problema se manifiesta de forma aguda debido al perfil del cliente predominante: universitarios con horarios pico concentrados en horas de recreo, que demandan atención inmediata y opciones personalizadas (toppings en frappés o salsas en waffles). Las entrevistas realizadas con la dueña y los empleados revelan impactos concretos: tiempos de espera excesivos durante horas pico por la escasez de personal de sala, errores frecuentes en la toma manual de pedidos, confusión en la coordinación de preparaciones (productos rápidos vs. tardíos), pérdida de información por registros en papel o Excel improvisados, y dificultad para realizar un control básico de ventas, gastos y stock. Estos factores no solo generan insatisfacción en el cliente —con posibles pérdidas de ventas y menor fidelización—, sino que también sobrecargan al personal existente, incrementan el riesgo de agotamiento y limitan el potencial de crecimiento del negocio en un entorno universitario competitivo.

El impacto general se resume en:

- Reducción de la eficiencia operativa y aumento de costos indirectos.
- Disminución de la satisfacción del cliente y riesgo de pérdida de mercado.
- Limitación en la toma de decisiones informadas por parte de la administración, al carecer de datos precisos sobre ventas y preferencias.

### **3.3. Objetivos del proyecto**

### **3.3.1. Objetivo general**

Diseñar e implementar un sistema web de menú digital y gestión de pedidos accesible mediante código QR para la cafetería Polaris, que automatice el flujo de atención al cliente, reduzca las ineficiencias operativas derivadas de métodos manuales y contribuya a la optimización de recursos en un contexto de escasez de personal.

### **3.3.2. Objetivos específicos**

- Digitalizar el menú y permitir la personalización de productos en tiempo real desde dispositivos móviles de los clientes.
- Automatizar el envío y seguimiento de pedidos al panel de gestión de empleados, con actualización de estados en tiempo real y notificaciones al cliente.
- Generar reportes simples de ventas y operaciones diarias que faciliten la toma de decisiones administrativas.
- Asegurar compatibilidad y usabilidad del sistema en dispositivos móviles y computadoras, alineándose con el perfil digital de la población universitaria objetivo.

## **IV. JUSTIFICACION DEL PROYECTO**

La implementación del Sistema Polaris representa una respuesta estratégica y oportuna a las ineficiencias operativas identificadas en la cafetería Polaris, alineándose con las tendencias globales, nacionales y locales de transformación digital en el sector gastronómico. Esta sección sustenta la necesidad del software desde perspectivas operativas, económicas y estratégicas, destacando los beneficios esperados y los beneficiarios directos e indirectos.

### **4.1. Beneficios esperados**

El proyecto se justifica por su capacidad para mitigar los problemas operativos derivados de la escasez de personal y los procesos manuales, generando impactos positivos cuantificables de forma cualitativa y respaldados por tendencias del sector.

- **Reducción significativa de tiempos de espera:** La adopción de menús digitales accesibles mediante códigos QR permite que los clientes realicen pedidos de manera autónoma desde sus dispositivos móviles, eliminando la dependencia de la atención presencial del personal. Estudios y experiencias internacionales indican que esta tecnología puede reducir los tiempos de espera en un rango estimado del 30-50 %, al agilizar el flujo desde la consulta del menú hasta el envío del pedido directo a la cocina. En entornos universitarios con picos de demanda concentrados, esta optimización mejora la rotación de mesas y la satisfacción inmediata del cliente.
- **Minimización de errores humanos y mejora en la precisión:** La toma manual de pedidos genera confusiones frecuentes en personalizaciones (toppings o salsas) y coordinación entre preparaciones rápidas y tardías. El sistema automatizado asegura que las órdenes se transmitan en tiempo real con detalles exactos, reduciendo errores y reprocesos, lo que contribuye a una mayor eficiencia operativa y menor desperdicio.
- **Liberación de carga laboral del personal:** En un contexto de escasez crónica de mano de obra cualificada en el sector Horeca —una tendencia global y regional para 2025, el

sistema permite al personal de sala centrarse en tareas de mayor valor, como la preparación, entrega y atención personalizada, en lugar de la toma repetitiva de pedidos. Esto reduce el agotamiento, mejora la retención de empleados y optimiza el uso de recursos humanos limitados.

- Mejora en el control administrativo y toma de decisiones: El registro automático de pedidos, estados y disponibilidad de productos genera datos básicos para reportes diarios/semanales de venta. Esto facilita la identificación de productos más vendidos, el ajuste de compras y una visión más clara de ingresos vs. gastos, superando las limitaciones actuales de registros en papel y Excel improvisados.
- Aumento potencial de ventas y fidelización: La experiencia digital —con personalización intuitiva y notificaciones en tiempo real— atrae a un perfil de cliente joven y nativo digital (estudiantes universitarios), incrementando el ticket promedio y la frecuencia de visitas mediante una percepción de modernidad y comodidad.

Estos beneficios se alinean con evidencias del sector: la digitalización mediante menús QR y pedidos automáticos no solo optimiza operaciones, sino que contribuye a incrementos en eficiencia y rentabilidad, como se observa en implementaciones exitosas en Latinoamérica y globalmente.

#### **4.2. Beneficiarios del proyecto**

El sistema genera valor para múltiples actores, priorizando el ecosistema de la cafetería Polaris en Abancay:

- Clientes (principalmente universitarios de la UNAMBA): Acceden a un menú interactivo y actualizado en tiempo real mediante QR, personalizan productos sin presión de tiempo y reciben notificaciones sobre el estado del pedido. Esto mejora su experiencia al reducir esperas y errores, ofreciendo comodidad en un entorno con horarios ajustados y alta demanda.

- Empleados y personal operativo: Disfrutan de un panel de gestión intuitivo que centraliza pedidos, actualiza estados en tiempo real y marca disponibilidad de productos. La automatización libera tiempo de tareas repetitivas, reduce estrés por sobrecarga y permite enfocarse en preparación y servicio de calidad, contribuyendo a un ambiente laboral más sostenible.
- Administración y dueña de la cafetería: Obtiene herramientas para un control básico de stock, ventas diarias y gastos, facilitando decisiones informadas sobre compras y ajustes de oferta. El sistema promueve mayor eficiencia operativa, potencial aumento de ingresos y posicionamiento como establecimiento innovador en el campus universitario.
- La universidad y comunidad local (indirectamente): Contribuye a un servicio más ágil y moderno en el campus, mejorando la percepción general de la UNAMBA como institución alineada con tendencias digitales y sostenibles.

El proyecto no solo resuelve problemas inmediatos, sino que posiciona a la cafetería Polaris como referente de innovación en el sector gastronómico universitario de Abancay, justificando plenamente su desarrollo e implementación.

## **V. ALCANCE DEL PROYECTO**

El alcance define con precisión los límites del sistema a desarrollar, estableciendo qué funcionalidades y componentes se incluyen en el proyecto y cuáles quedan excluidos para mantener un enfoque realista, viable y alineado con los recursos disponibles (equipo estudiantil, tiempo y presupuesto limitado). Esta delimitación se basa en los requisitos levantados durante las reuniones con la dueña y empleados de la cafetería Polaris, priorizando la digitalización del menú y la gestión de pedidos como núcleo principal.

### **5.1. Incluye**

El Sistema Polaris abarca las siguientes funcionalidades y componentes esenciales, desarrollados bajo un enfoque híbrido (modelo en cascada para planificación y análisis, Scrum para implementación):

- Menú digital accesible vía código QR:
  - Carga rápida del menú en dispositivos móviles (menos de 5 segundos estimado).
  - Visualización categorizada de productos (Coffees, Frappés, Jugos y Bebidas, Ensaladas de Fruta, Waffles, Sándwichs).
  - Detalles de cada producto: nombre, descripción, precio base y disponibilidad en tiempo real.
- Personalización de productos:
  - Selección de acompañamientos/toppings (salsas de chocolate, manjar, miel, frutas, chantillí, helado, Oreo, etc., según el producto).
  - Configuración de cantidad y opciones adicionales.
  - Cálculo automático del subtotal y total del carrito.
- Gestión del carrito y confirmación de pedido:
  - Adición, modificación y eliminación de ítems.



- Indicación de tipo de pedido: "En mesa" (con número de mesa enumerado) o "Para llevar".
- Selección de método de pago (Yape o efectivo), con instrucciones específicas para Yape (mostrar número y mensaje de acercamiento a caja).
- Generación de identificador único de pedido y envío inmediato al panel del empleado.
- Panel de gestión para empleados:
  - Autenticación segura de empleados.
  - Visualización en tiempo real de pedidos entrantes (ordenados por hora de llegada), con detalles: ID, tipo (mesa/para llevar), ítems, total, método de pago y estado.
  - Actualización de estados del pedido: Pendiente → En preparación → Listo → Entregado.
  - Notificación automática al cliente cuando el pedido pasa a "Listo" (vía interfaz web responsiva en el dispositivo del cliente).
- Reportes básicos administrativos:
  - Generación de resúmenes diarios/semanales de ventas (totales, por categoría o producto más vendido).
  - Exportación simple (vista previa en pantalla, posible PDF/Excel).
- Aspectos técnicos generales:
  - Arquitectura full-stack: Frontend con Angular (responsivo para móviles y PC), Backend con Spring Boot, Base de datos MySQL.
  - Diseño de base de datos relacional (modelo Entidad-Relación, lógico y físico) con tablas para categorías, productos, acompañamientos, pedidos, ítems de pedido y empleados.

- Notificaciones en tiempo real (simuladas vía polling o WebSockets básicos).
- Compatibilidad multiplataforma (celulares y computadoras usadas por el negocio).

El sistema se implementa como una aplicación web progresiva (PWA), accesible sin instalación de apps nativas, priorizando simplicidad y bajo costo de mantenimiento.

## **5.2. No incluye**

Para mantener el proyecto dentro de los límites académicos y operativos realistas, se excluyen deliberadamente las siguientes funcionalidades y extensiones (que podrían considerarse en fases futuras):

- Control de inventario detallado (gestión de insumos, entradas/salidas automáticas por receta, alertas de reabastecimiento, trazabilidad de proveedores).
- Integración con pasarelas de pago (pagos directos vía tarjeta, Yape automático o plataformas como Culqi/PayU).
- Módulo de fidelización/digitalización de tarjetas (puntos, descuentos por visitas, perfil de cliente).
- Gestión avanzada de mesas (mapa interactivo, reservas, asignación automática).
- Integración con delivery externo (Rappi, PedidosYa).
- Análisis predictivo o BI avanzado (pronósticos de ventas, machine learning).
- Módulos de nómina, contabilidad completa o RRHH.
- Hardware adicional (impresoras térmicas de cocina, tablets dedicadas para empleados, lectores QR físicos).

Estas exclusiones se fundamentan en la complejidad técnica, el tiempo disponible y la prioridad expresada por la dueña: enfocarse en la digitalización del menú y automatización del flujo de pedidos como solución inmediata y de alto impacto.

## **VI. MODELO**

El desarrollo del Sistema Polaris se estructura bajo el modelo en cascada (Waterfall) para las fases de planificación, análisis y documentación, combinado con Scrum en la etapa de implementación y desarrollo del código. Este enfoque híbrido permite una planificación secuencial rigurosa y estructurada —ideal para proyectos académicos con entregas definidas—, mientras que la flexibilidad ágil en el desarrollo facilita iteraciones rápidas, retroalimentación continua y ajustes basados en pruebas.

El modelo en cascada se aplica de manera secuencial en las siguientes fases principales: requisitos, diseño, implementación, pruebas, despliegue y cierre. Cada fase produce artefactos documentados que sirven de entrada a la siguiente, minimizando riesgos en un proyecto con alcance delimitado.

### **6.1. Análisis**

El análisis constituye la primera fase formal del modelo en cascada aplicado al proyecto. En esta etapa se realiza una comprensión exhaustiva del problema de negocio, se recopilan y documentan los requisitos del sistema, y se establecen las bases técnicas y funcionales que guiarán todas las fases subsiguientes. El enfoque secuencial y riguroso del modelo en cascada garantiza que los requisitos queden completamente definidos y validados antes de avanzar al diseño, minimizando riesgos de ambigüedad y reprocesos en etapas posteriores.

El análisis se fundamenta principalmente en:

- La reunión inicial con la dueña de la cafetería Polaris (representante del cliente).
- Entrevistas complementarias con empleados operativos.
- Observación directa de los procesos actuales de atención.
- Revisión de los registros manuales y en Excel utilizados en la operación diaria.

Esta información permitió identificar las necesidades reales del negocio y traducirlas en especificaciones técnicas claras, priorizando la digitalización del menú y la automatización del flujo de pedidos como elementos críticos de alto impacto.

#### **6.1.1. Introducción**

El propósito de la fase de análisis es establecer una visión compartida y precisa entre el equipo de desarrollo y el cliente respecto al sistema a construir. Se busca responder a las preguntas fundamentales: ¿qué problema se resuelve?, ¿para quién?, ¿qué funcionalidades son imprescindibles?, ¿qué restricciones existen? y ¿cuáles son los límites del proyecto?

En el contexto de la cafetería Polaris, el análisis revela que los métodos tradicionales de atención (toma manual de pedidos, comunicación verbal, registro en papel y control básico en hojas de cálculo) generan ineficiencias críticas que afectan directamente la experiencia del cliente universitario, la productividad del personal y la capacidad de toma de decisiones de la administración.

El sistema propuesto, denominado Sistema Polaris, se concibe como una solución web integral que:

- Permite a los clientes acceder al menú mediante código QR desde sus dispositivos móviles.
- Automatiza la personalización, el envío y el seguimiento de pedidos.
- Proporciona a los empleados un panel centralizado de gestión en tiempo real.
- Ofrece herramientas básicas de control de disponibilidad y reportes operativos.

Este enfoque responde directamente a la escasez de personal de sala, a los tiempos de espera excesivos en horas pico y a la necesidad de modernización en un entorno universitario cada vez más digital.

##### **6.1.1.1. Contexto**

La cafetería Polaris funciona como el principal punto de abastecimiento de bebidas y alimentos ligeros dentro del campus de la Universidad Nacional Micaela Bastidas de Apurímac (UNAMBA), en Abancay, Perú. Su clientela está compuesta mayoritariamente por estudiantes universitarios (aproximadamente 85-90 % según entrevistas), quienes presentan patrones de consumo concentrados en horarios de recreo (entre clases), con alta demanda de productos rápidos, personalizables y refrescantes.

El establecimiento opera con un número reducido de personal (generalmente 2-3 personas por turno en horas pico), lo cual genera cuellos de botella severos cuando la afluencia aumenta. Los procesos actuales dependen casi exclusivamente de:

- Toma manual de pedidos con libreta y lápiz.
- Comunicación verbal o gritos entre sala y cocina.
- Registro de ventas en hojas de papel y control de caja en Excel.
- Actualización manual de disponibilidad de productos (mediante anotaciones o comunicación oral).

Estos métodos, si bien funcionales en periodos de baja demanda, se vuelven ineficientes y propensos a errores durante los picos de afluencia, generando insatisfacción del cliente, pérdida de ventas potenciales y sobrecarga laboral del personal.

#### 6.1.1.2. Propósito

El propósito principal del sistema es transformar operacionalmente la forma en que se atiende a los clientes en la cafetería Polaris, mediante la introducción de una solución tecnológica que:

- Elimine la dependencia de la atención presencial para la consulta del menú y la toma de pedidos.
- Reduzca significativamente los tiempos de espera y los errores humanos asociados a la personalización y registro de órdenes.

- Libere al personal de sala de tareas repetitivas, permitiéndole enfocarse en la preparación, entrega y atención directa de mayor valor.
- Proporcione información operativa básica y actualizada que facilite la toma de decisiones administrativas (productos más vendidos, control elemental de disponibilidad).
- Posicione a la cafetería como un establecimiento moderno e innovador dentro del campus universitario, alineándose con su visión estratégica para el año 2027.

En síntesis, el sistema busca compensar la escasez estructural de personal mediante la automatización inteligente de procesos, mejorando simultáneamente la experiencia del cliente y la eficiencia operativa del negocio.

#### 6.1.1.3. Alcance

El alcance del proyecto se delimita estrictamente para garantizar su viabilidad en el marco académico y temporal disponible. Incluye:

- Desarrollo de un menú digital interactivo accesible mediante código QR.
- Funcionalidad completa de personalización de productos (selección de salsas y toppings según tipo de producto).
- Gestión del carrito de compras y confirmación de pedido por parte del cliente.
- Panel de gestión para empleados con visualización, actualización de estados y control básico de disponibilidad de productos.
- Notificaciones simples al cliente sobre el avance del pedido.
- Reportes operativos básicos (ventas diarias y semanales).

No incluye:

- Control detallado de inventario de insumos y recetas.
- Integración con pasarelas de pago electrónico (solo indicación de Yape y efectivo).
- Módulo de fidelización o tarjetas digitales de cliente.

- Gestión avanzada de mesas (mapa interactivo, reservas).
- Integración con servicios de delivery externos.
- Análisis predictivo o inteligencia de negocios avanzada.

El sistema se implementa como aplicación web progresiva (PWA), sin requerir instalación de aplicaciones nativas, y prioriza la simplicidad y usabilidad en dispositivos móviles.

#### 6.1.1.4. definiciones

A continuación se presentan las definiciones clave de términos y conceptos utilizados a lo largo del proyecto, con el fin de garantizar uniformidad en la comprensión entre equipo de desarrollo, cliente y evaluadores:

- Código QR: Elemento gráfico estático ubicado en mesas y mostrador que, al ser escaneado con la cámara del teléfono móvil, dirige al usuario al menú digital del sistema.
- Menú digital: Interfaz web responsiva que presenta las categorías de productos, descripción, precios y estado de disponibilidad en tiempo real.
- Acompañamiento: Elemento personalizable que el cliente puede agregar a ciertos productos. Se clasifican en dos tipos:
  - SALSA (ej. Chocolate, Manjar, Miel).
  - TOPPING (ej. Fresa, Plátano, Chantilly, Oreo, Helado).
- Carrito: Colección temporal de productos seleccionados por el cliente antes de confirmar el pedido.
- Tipo de pedido:
  - MESA: Pedido asociado a un número de mesa específico (entero positivo).
  - LLEVAR: Pedido para retiro en mostrador (sin mesa asociada).
- Estado del pedido: Ciclo de vida del pedido en el panel del empleado:

- ESPERA (recién recibido).
- PREPARACION (en proceso de elaboración).
- LISTO (terminado y listo para entrega).
- Método de pago:
  - EFECTIVO: Pago en caja al momento de entrega.
  - YAPE: Pago mediante aplicación Yape, con instrucciones para acercarse a caja.
- Panel del empleado: Interfaz administrativa protegida por autenticación, destinada a la gestión operativa de pedidos y disponibilidad de productos.
- Disponibilidad: Indicador booleano (true/false) que determina si un producto puede ser seleccionado por los clientes en el menú digital.

Estas definiciones se utilizan de manera consistente en toda la documentación y en la implementación del sistema.

### **6.1.2. Descripción general**

La descripción general ofrece una visión integral y panorámica del sistema a desarrollar, estableciendo su posición relativa dentro del contexto operativo de la cafetería Polaris, sus objetivos principales, las funcionalidades clave y las expectativas de uso. Esta subsección sirve como puente entre el análisis detallado de requisitos y el diseño subsiguiente, permitiendo a cualquier lector (evaluador, cliente o miembro del equipo) comprender rápidamente la esencia y el valor agregado del proyecto.

#### **6.1.2.1. Perspectiva del producto**

El Sistema Polaris se concibe como una solución tecnológica de transformación digital de bajo costo y alto impacto para una pequeña cafetería universitaria. En lugar de ser un sistema ERP complejo o una plataforma de e-commerce completa, se trata de una aplicación web progresiva (PWA) especializada en:



- Automatizar el punto más crítico del proceso de atención: la toma y seguimiento de pedidos.
- Compensar la escasez estructural de personal de sala mediante autoservicio digital del cliente.
- Introducir trazabilidad y visibilidad en tiempo real sin requerir hardware adicional significativo.

Desde una perspectiva estratégica, el producto posiciona a la cafetería Polaris como el primer establecimiento del campus de la UNAMBA con un sistema de pedidos mediante código QR, diferenciándose de la competencia local (otras cafeterías y quioscos universitarios) que aún operan con métodos 100 % manuales.

En términos comparativos, el sistema se sitúa en un nivel intermedio de madurez tecnológica dentro del sector Horeca peruano: supera ampliamente los métodos tradicionales, pero permanece deliberadamente simple y económico en comparación con soluciones empresariales como los sistemas de Punto de Venta (POS) integrados con cocina digital, kioscos físicos y pasarelas de pago avanzadas.

#### 6.1.2.2. Funcionalidades

El sistema Polaris entrega las siguientes funcionalidades principales, agrupadas por actor:

Funcionalidades orientadas al cliente (estudiante universitario):

- Escaneo de código QR y carga automática del menú digital.
- Navegación por categorías de productos (Jugos, Ensaladas y Sándwiches; Cafés; Frappés; Waffles).
- Visualización detallada de cada producto (nombre, descripción, precio base, disponibilidad actual).
- Personalización de productos seleccionando salsas y/o toppings disponibles según el ítem.

- Gestión del carrito: agregar, modificar cantidad, eliminar productos, visualización de subtotal y total.
- Confirmación del pedido indicando: tipo (MESA o LLEVAR), número de mesa (si aplica), método de pago (EFECTIVO o YAPE) y nombre del cliente.
- Recepción de notificación cuando el pedido cambia al estado LISTO.

Funcionalidades orientadas al empleado/mozo:

- Autenticación segura en el panel administrativo.
- Visualización en tiempo real de todos los pedidos activos (ordenados cronológicamente).
- Consulta detallada de cada pedido: ítems, cantidades, personalizaciones, total, tipo, método de pago y mesa.
- Actualización manual del estado del pedido (ESPERA → PREPARACION → LISTO).
- Control de disponibilidad: marcar cualquier producto como no disponible (lo que lo oculta automáticamente del menú del cliente).
- Cierre manual del pedido al marcarlo como entregado (opcional, según flujo final acordado).

Funcionalidades administrativas básicas:

- Reporte resumido de ventas del día (total vendido, desglose por categoría/producto más vendido).
- Registro histórico de pedidos para consulta posterior.

Todas las funcionalidades se ejecutan en tiempo real y mantienen consistencia entre la vista del cliente y el panel del empleado.

#### 6.1.2.3. Características del usuario

El sistema está diseñado considerando dos perfiles de usuario principales, con características y necesidades claramente diferenciadas:

Cliente (usuario final – estudiante universitario)

- Edad aproximada: 18–25 años.
- Alto nivel de familiaridad con dispositivos móviles y aplicaciones web.
- Expectativas: rapidez, simplicidad, diseño atractivo y moderno.
- Contexto de uso: sentado en mesa o de pie en mostrador, con prisa por regresar a clases.
- Dispositivo principal: smartphone Android o iOS (pantallas pequeñas a medianas).
- Conexión: Wi-Fi institucional de la universidad (velocidad media-baja en horas pico).

Empleado/Mozo (usuario operativo)

- Edad aproximada: 20–35 años.
- Nivel de familiaridad tecnológica: medio (manejan WhatsApp, redes sociales, Excel básico).
- Expectativas: interfaz clara, legible, con botones grandes, mínima curva de aprendizaje.
- Contexto de uso: en movimiento, con prisa, en ambiente ruidoso y con iluminación variable.
- Dispositivo principal: laptop compartida en mostrador o computadora de escritorio (pantalla grande).
- Necesidad crítica: respuesta inmediata y actualizaciones automáticas sin refrescar manualmente.

Estas características guían fuertemente las decisiones de diseño de interfaz (mobile-first para clientes, desktop-friendly y muy legible para empleados).

#### 6.1.2.4. Cuestionario de requerimientos de software

El cuestionario de requerimientos se aplicó de forma estructurada durante las reuniones iniciales y complementarias con la dueña de la cafetería Polaris y los empleados operativos. Este instrumento permitió recopilar, priorizar y validar tanto los requisitos funcionales como

no funcionales, asegurando que el sistema responda directamente a las necesidades reales del negocio y no incorpore funcionalidades innecesarias o fuera del alcance acordado.

El cuestionario se dividió en tres bloques principales:

**Bloque A: Requerimientos funcionales (priorizados por el cliente)**

1. ¿Es imprescindible que el cliente pueda acceder al menú completo escaneando un código QR desde su celular? Respuesta: Sí – Prioridad alta. Actualmente usan menús impresos que se ensucian y no se actualizan fácilmente.
2. ¿Debe el sistema permitir personalizar productos (selección de salsas y toppings)? Respuesta: Sí – Prioridad alta. Es uno de los principales atractivos de los waffles y frappés.
3. ¿Es necesario diferenciar pedidos “En mesa” (con número de mesa) y “Para llevar”? Respuesta: Sí – Prioridad media-alta. Muchos clientes piden para llevar cuando las mesas están llenas.
4. ¿Debe el sistema notificar al cliente cuando su pedido está listo? Respuesta: Sí – Prioridad alta. Reduce la necesidad de que el cliente esté pendiente constantemente.
5. ¿Requiere el panel del empleado mostrar los pedidos en orden de llegada y permitir cambiar su estado? Respuesta: Sí – Prioridad crítica.
6. ¿Es suficiente un control básico de disponibilidad (marcar producto como agotado/no disponible)? Respuesta: Sí – Prioridad alta. No se requiere inventario detallado de insumos en esta fase.
7. ¿Se necesitan reportes básicos de ventas diarias y semanales? Respuesta: Sí – Prioridad media. Actualmente usan Excel, pero desean algo más integrado.

**Bloque B: Requerimientos no funcionales y restricciones**

1. ¿El sistema debe funcionar correctamente en celulares y computadoras? Respuesta: Sí – Obligatorio (mobile-first para clientes, desktop para empleados).

2. ¿Cuál es el tiempo máximo aceptable de carga del menú al escanear el QR? Respuesta: Menos de 5 segundos (idealmente 2-3 s).
3. ¿Debe ser fácil de usar sin capacitación extensa para los empleados? Respuesta: Sí – Curva de aprendizaje máxima: 15-20 minutos.
4. ¿Se requiere integración con pasarelas de pago electrónico? Respuesta: No – Solo indicación de Yape o efectivo (pago se gestiona en caja).
5. ¿Se necesita módulo de fidelización o tarjetas de puntos? Respuesta: No en esta versión inicial.

#### Bloque C: Expectativas y restricciones generales

1. ¿Cuál es el principal beneficio esperado por la dueña? Respuesta: Reducir tiempos de espera, disminuir errores y liberar al personal de la toma de pedidos repetitiva.
2. ¿Hay restricciones de presupuesto significativas? Respuesta: Sí – Proyecto académico con recursos limitados (sin inversión externa importante).
3. ¿Se acepta que el sistema sea una aplicación web sin necesidad de descargar app? Respuesta: Sí – Preferible (más fácil de mantener y actualizar).

Los resultados del cuestionario sirvieron de base para elaborar la lista definitiva de requisitos funcionales y no funcionales (ver 6.1.3), así como para confirmar las exclusiones explícitas del alcance (Sección V).

### **6.1.3. Requisitos específicos**

#### 6.1.3.1. Requisitos funcionales

A continuación se presenta la lista consolidada y numerada de requisitos funcionales principales, derivados directamente del cuestionario, las entrevistas y el análisis de procesos. Se agrupan por módulo principal para mayor claridad.

Módulo Cliente (Menú Digital y Pedido):

- RF01: El sistema deberá permitir al cliente escanear un código QR y cargar el menú completo en formato web responsivo.
- RF02: El menú deberá mostrar las categorías disponibles y listar los productos con nombre, descripción, precio base y estado de disponibilidad (visible/invisible según control del empleado).
- RF03: El sistema deberá permitir seleccionar un producto y configurar sus acompañamientos disponibles (salsas y toppings) según el tipo de producto.
- RF04: El sistema deberá implementar un carrito de compras donde el cliente pueda agregar, modificar cantidad, eliminar ítems y visualizar el subtotal y total actualizado en tiempo real.
- RF05: Al confirmar el pedido, el sistema deberá solicitar: nombre del cliente, tipo de pedido (MESA o LLEVAR), número de mesa (solo si MESA), y método de pago (EFECTIVO o YAPE).
- RF06: Al confirmar, el sistema deberá generar un identificador único (UUID) para el pedido y enviarlo inmediatamente al panel del empleado.

#### Módulo Empleado (Panel de Gestión):

- RF07: El sistema deberá contar con un módulo de autenticación para empleados (usuario y contraseña).
- RF08: El panel deberá mostrar en tiempo real la lista de pedidos activos, ordenados por hora de creación (el más antiguo primero).
- RF09: Cada pedido deberá mostrar: ID, nombre del cliente, ítems con cantidades y personalizaciones, total, tipo de pedido, método de pago y número de mesa (si aplica).
- RF10: El empleado deberá poder cambiar el estado del pedido manualmente: ESPERA → PREPARACION → LISTO.

- RF11: Al cambiar el estado a LISTO, el sistema deberá notificar automáticamente al cliente (actualización visible en su navegador móvil).
- RF12: El empleado deberá poder marcar cualquier producto como no disponible, lo que actualizará en tiempo real su visibilidad en el menú del cliente.

#### Módulo Administrativo Básico:

- RF13: El sistema deberá generar un reporte resumido diario de ventas (total vendido, número de pedidos, desglose aproximado por categoría).
- RF14: El reporte deberá ser visualizable en pantalla.

#### 6.1.3.2. Requisitos no funcionales

Los requisitos no funcionales se centran en atributos de calidad del sistema y restricciones operativas:

- RNF01: El tiempo de carga inicial del menú digital no deberá exceder los 5 segundos en conexiones Wi-Fi institucionales típicas.
- RNF02: La interfaz para clientes deberá ser responsiva (mobile-first), adaptándose correctamente a pantallas de 360 px a 1920 px de ancho.
- RNF03: La interfaz del panel empleado deberá ser optimizada para pantallas de al menos 1366 px (laptop estándar).
- RNF04: Las contraseñas de empleados deberán almacenarse como hash seguro (no en texto plano).
- RNF05: El sistema deberá mantener consistencia de datos en tiempo real entre cliente y empleado (actualización automática sin necesidad de refrescar manualmente la página).

- RNF06: El sistema deberá ser usable sin capacitación extensa: máximo 15 minutos de explicación inicial para empleados.
- RNF07: El sistema deberá funcionar correctamente en navegadores modernos: Chrome, Safari, Firefox (últimas versiones móviles y desktop).
- RNF08: El diseño deberá ser intuitivo, con botones grandes y texto legible en entornos con iluminación variable.
- RNF09: El sistema deberá soportar al menos 100 pedidos simultáneos sin degradación notable de rendimiento (escalabilidad mínima aceptable).
- RNF10: El mantenimiento y actualización del menú (agregar/eliminar productos) deberá poder realizarse desde el panel empleado sin intervención del equipo de desarrollo.

#### **6.1.4. Modelo de negocios**

El modelado de negocios tiene como objetivo representar de forma gráfica y estructurada los procesos actuales y futuros del negocio de la cafetería Polaris, identificando actores, flujos principales, interacciones y puntos de mejora. Este modelado se realiza utilizando herramientas estándar de arquitectura de software, con énfasis en la visualización clara para facilitar la comunicación entre el equipo de desarrollo, el cliente y los evaluadores académicos.

Se emplean principalmente dos tipos de diagramas:

- Diagramas de flujo de procesos (flujogramas o mapas de procesos): Representan secuencialmente el ciclo completo de atención al cliente, desde la llegada hasta la entrega del pedido.
- Diagramas de casos de uso (UML Use Case Diagram): Identifican los actores principales (Cliente y Empleado) y los casos de uso clave que el sistema debe soportar.

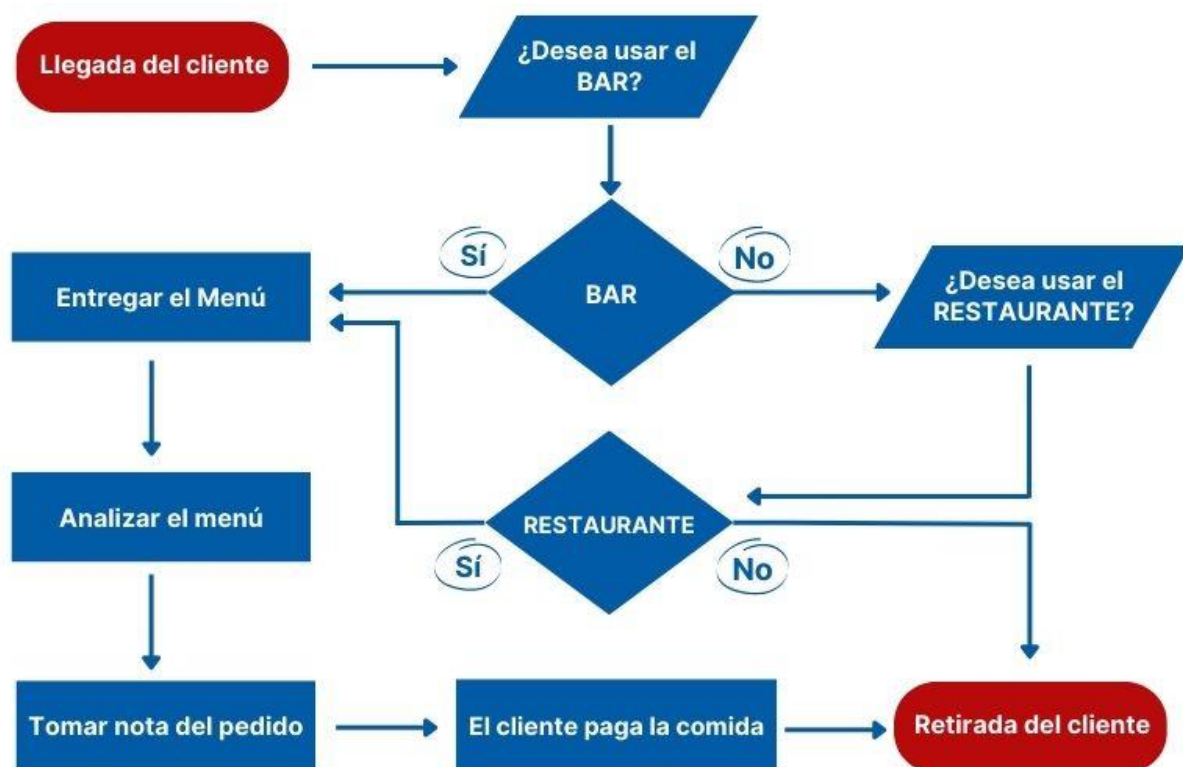


Estos artefactos permiten visualizar cómo el sistema Polaris transforma el proceso manual actual en uno automatizado y eficiente.

#### 6.1.4.1. Diagrama de flujo de procesos

Proceso actual

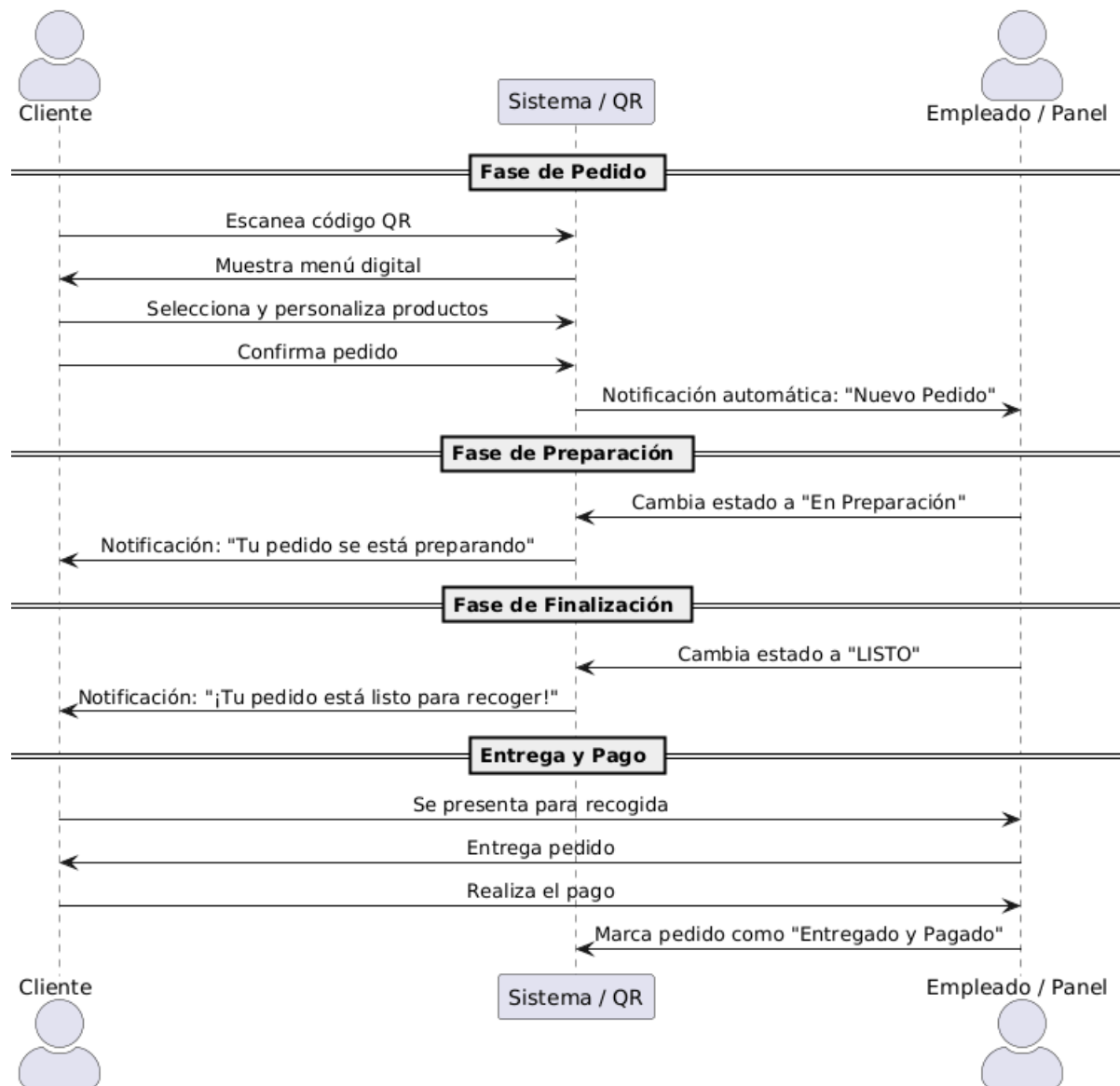
Figura 1: Diagrama de flujo del proceso actual de atención



Representa el flujo manual tradicional: llegada del cliente → toma de pedido verbal/escrita → comunicación a cocina → preparación → entrega → cobro en caja. Puntos críticos: espera prolongada, errores en anotación, confusión en mesas.

Proceso propuesto con el sistema

Figura 2: Diagrama de Flujo del Proceso Propuesto con el Sistema



Flujo optimizado: cliente escanea QR → selecciona y personaliza → confirma pedido → pedido llega automáticamente al panel empleado → empleado actualiza estados → notificación al cliente cuando está LISTO → entrega y cobro

#### 6.1.4.2. Diagrama de casos de uso (UML)

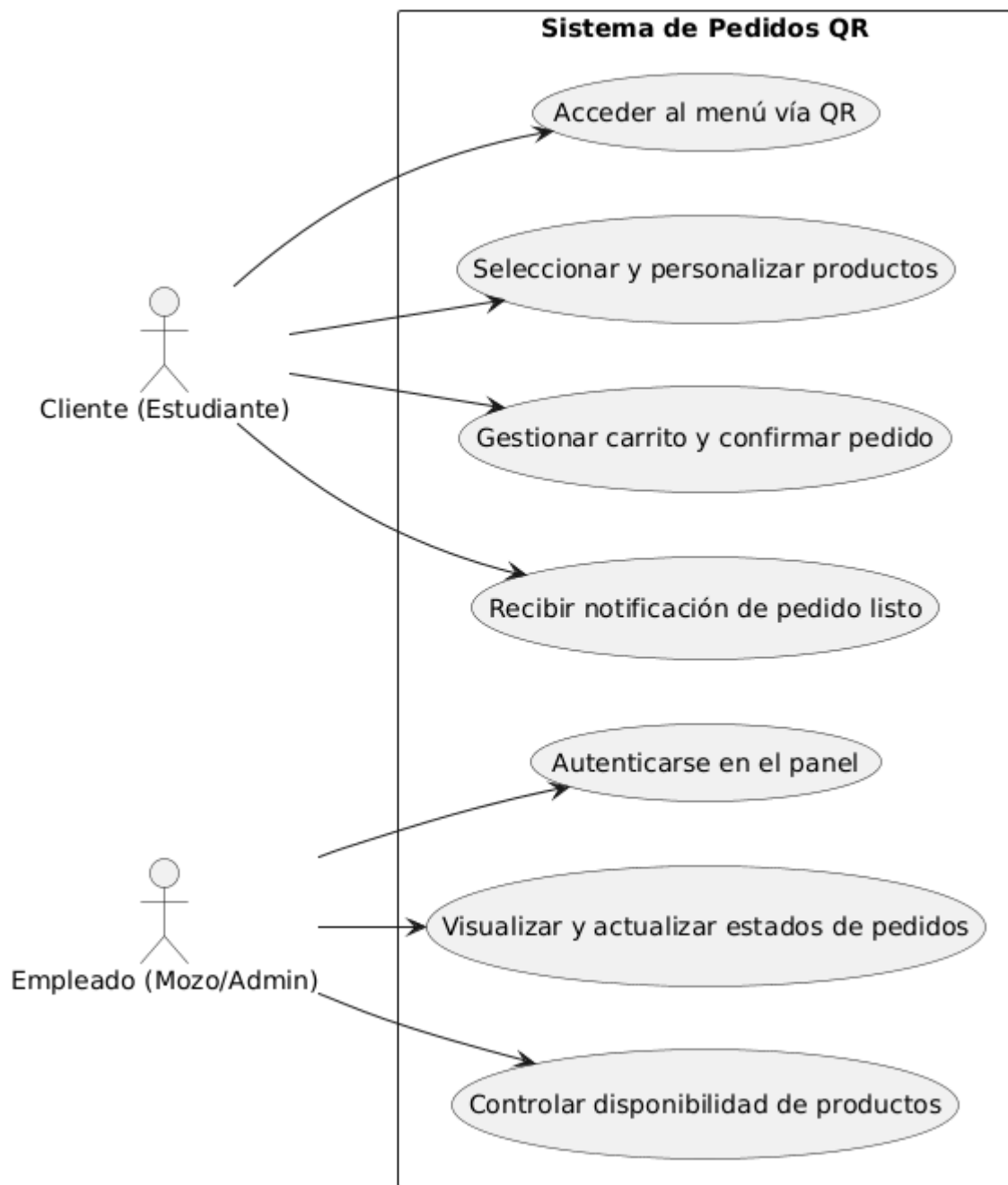
El diagrama de casos de uso identifica los actores principales y las funcionalidades que estos ejecutan sobre el sistema. Los actores son:

- Cliente (estudiante universitario): Interactúa con el menú digital desde su dispositivo.
- Empleado (mozo/administrador): Gestiona pedidos y disponibilidad desde el panel.

Casos de uso principales:

- Acceder al menú vía QR.
- Seleccionar y personalizar productos.
- Gestionar carrito y confirmar pedido.
- Autenticarse en el panel.
- Visualizar y actualizar estados de pedidos.
- Controlar disponibilidad de productos.
- Recibir notificación de pedido listo.

Figura 3: Diagrama de Casos de Uso del Sistema Polaris



#### 6.1.5. Diseño de base de datos

El diseño de la base de datos se realiza siguiendo un enfoque estructurado en tres niveles: **conceptual** (modelo Entidad-Relación), **lógico** y **físico**. Este proceso garantiza que la

estructura de datos sea coherente, normalizada y optimizada para las operaciones principales del sistema (consultas rápidas de menús, registro de pedidos y control de disponibilidad).

La base de datos seleccionada es **MySQL** (versión Community), por su robustez, gratuidad, soporte para transacciones ACID y amplia compatibilidad con el stack tecnológico elegido (Angular + Spring Boot).

#### 6.1.5.1. Modelo entidad-relación (ER)

El modelo Entidad-Relación representa las entidades principales del negocio y sus relaciones.

Entidades principales identificadas:

- Empleado
- Categoría
- Producto
- Acompañamiento
- Pedido
- PedidoItem (ítem de pedido – detalle)
- ProductoAcompañamiento (relación muchos-a-muchos entre Producto y Acompañamiento)

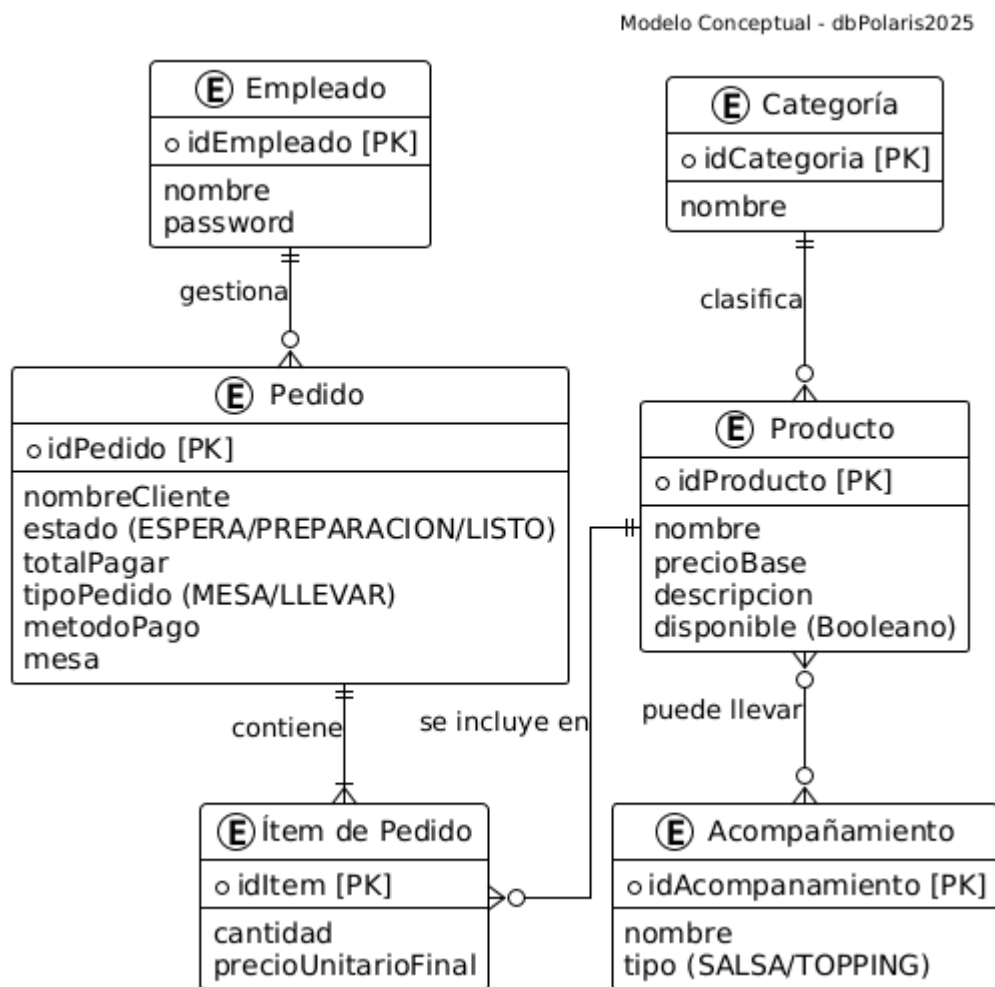
Relaciones principales:

- Un Producto pertenece a una sola Categoría (1:N).
- Un Producto puede tener varios Acompañamientos (N:M) → tabla intermedia ProductoAcompañamiento.
- Un Empleado puede gestionar múltiples Pedidos (1:N).
- Un Pedido contiene varios PedidoItem (1:N).
- Cada PedidoItem referencia un Producto (N:1).

Atributos destacados:

- Producto: id, nombre, precioBase, descripción, disponible (booleano).
- Acompañamiento: id, nombre, tipo (SALSA o TOPPING).
- Pedido: id, empleado, nombreCliente, estado (ENUM), totalPagar, tipoPedido (ENUM), metodoPago (ENUM), mesa (opcional), timestamps.

Figura 4: Diagrama Entidad-Relación



#### 6.1.5.2. Modelo lógico

El modelo lógico transforma el esquema conceptual en un esquema relacional, definiendo tablas, claves primarias, claves foráneas y restricciones.

## Tablas principales y estructura lógica:

### 1. empleado

- idEmpleado (CHAR(36) – UUID, PK)
- nombre (VARCHAR(100))
- password (VARCHAR(100) – hash)
- createdAt, updatedAt (DATETIME)

### 2. categoria

- idCategoria (CHAR(36) – UUID, PK)
- nombre (VARCHAR(100))

### 3. producto

- idProducto (CHAR(36) – UUID, PK)
- idCategoria (CHAR(36) – FK)
- disponible (BOOLEAN)
- nombre (VARCHAR(100))
- precioBase (DECIMAL(10,2))
- descripcion (TEXT)

### 4. acompanamiento

- idAcompanamiento (CHAR(36) – UUID, PK)
- nombreAcompanamiento (VARCHAR(100))
- tipoAcompanamiento (ENUM('SALSA','TOPPING'))

### 5. productoAcompanamiento (tabla intermedia)

- idProdAcomp (CHAR(36) – UUID, PK)
- idProducto (CHAR(36) – FK)
- idAcompanamiento (CHAR(36) – FK)

### 6. pedido

- idPedido (CHAR(36) – UUID, PK)
- idEmpleado (CHAR(36) – FK)
- nombreCliente (VARCHAR(100))
- estado (ENUM('ESPERA','PREPARACION','LISTO'))
- totalPagar (DECIMAL(10,2))
- tipoPedido (ENUM('MESA','LLEVAR'))
- metodoPago (ENUM('EFECTIVO','YAPE'))
- mesa (INT – NULL si LLEVAR)

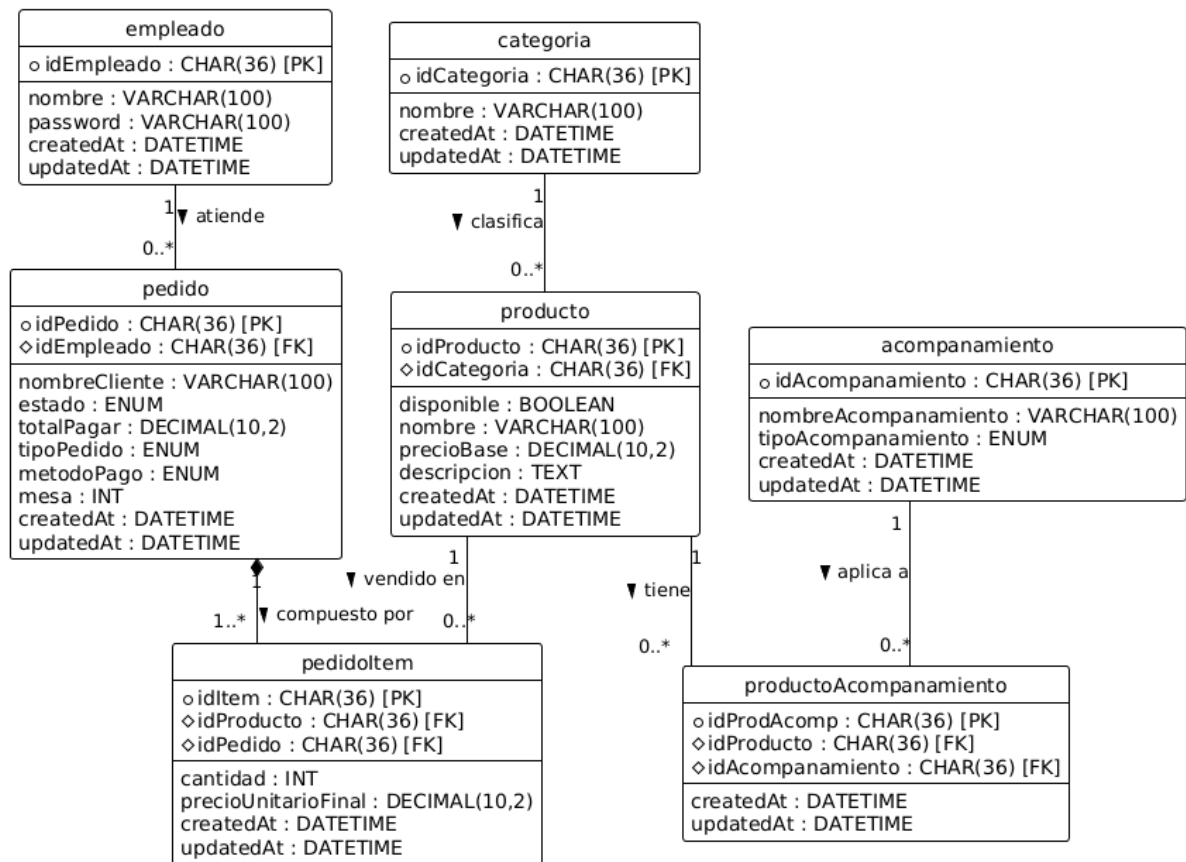
#### 7. pedidoItem

- idItem (CHAR(36) – UUID, PK)
- idProducto (CHAR(36) – FK)
- idPedido (CHAR(36) – FK)
- cantidad (INT)
- precioUnitarioFinal (DECIMAL(10,2))



Figura 5: Diagrama del Modelo Lógico

Modelo Lógico Polaris 2025



#### 6.1.5.3. Modelo físico

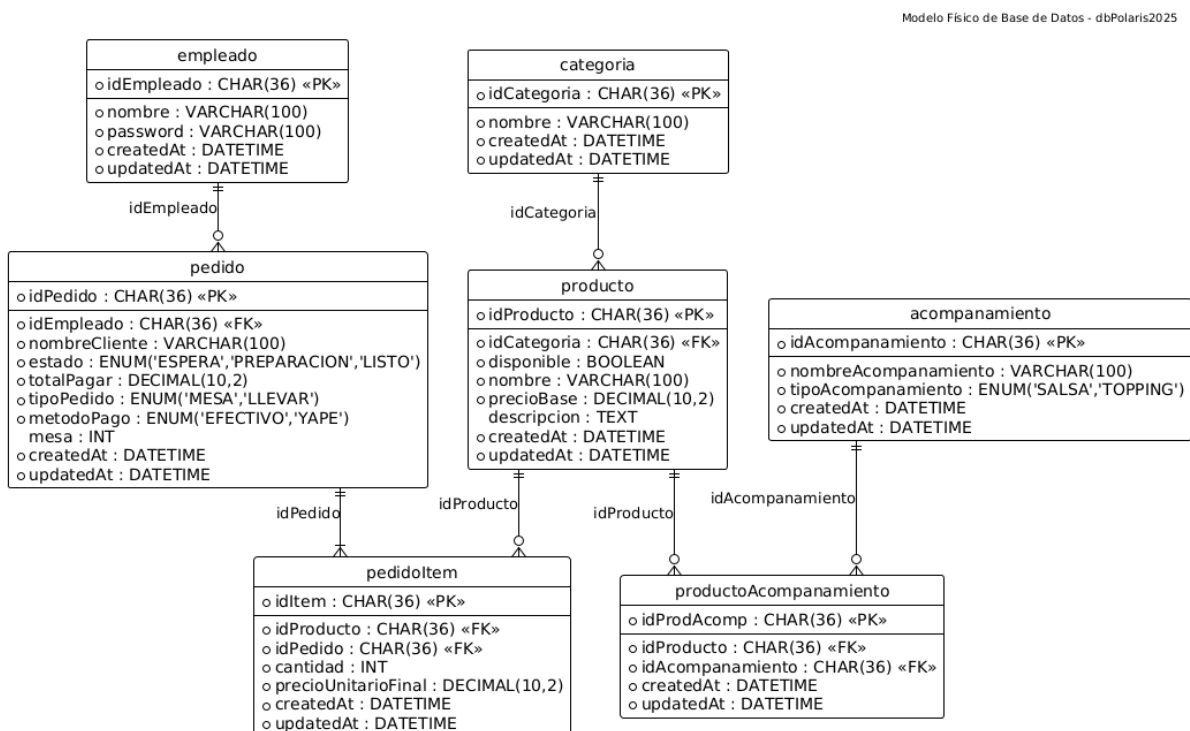
El modelo físico corresponde a la implementación real en MySQL, incluyendo decisiones específicas de almacenamiento, índices y restricciones adicionales.

Decisiones principales de implementación física:

- Uso de UUID (CHAR(36)) como clave primaria en todas las tablas principales para facilitar la generación distribuida y evitar colisiones.
- Campos createdAt y updatedAt con valor por defecto NOW() y actualización automática.

- Uso de ENUM para campos con valores fijos y conocidos (estado, tipoPedido, metodoPago, tipoAcompañamiento) → mejora legibilidad y reduce errores.
- Campo mesa en tabla pedido definido como INT NULL (solo obligatorio cuando tipoPedido = 'MESA').
- Índices sugeridos:
  - Índice en pedido.createdAt para consultas de pedidos por fecha.
  - Índice en pedido.estado para filtrar pedidos activos rápidamente.
  - Índice compuesto en pedidoItem(idPedido, idProducto) para consultas de detalle por pedido.

Figura 6: Diagrama del Modelo Físico



#### 6.1.5.4. Diccionario de base de datos

El diccionario de base de datos presenta una descripción detallada y estructurada de cada entidad (tabla) del modelo físico implementado en MySQL. Para cada tabla se indica el nombre de la entidad y, a continuación, una tabla con los siguientes campos: Nombre del atributo, Tipo de dato y Descripción (incluyendo restricciones relevantes, propósito y observaciones de uso).

*Tabla 1: Entidad empleado*

Nombre del atributo	Tipo de dato	Descripción
idEmpleado	CHAR(36)	Identificador único universal (UUID). Clave primaria. No nulo.
nombre	VARCHAR(100)	Nombre completo del empleado. Campo obligatorio.
password	VARCHAR(100)	Contraseña del empleado almacenada en formato hash (no texto plano). Obligatorio.
createdAt	DATETIME	Fecha y hora de creación del registro. Valor por defecto: NOW().
updatedAt	DATETIME	Fecha y hora de la última actualización. Actualización automática.

*Tabla 2: Entidad categoría*

Nombre del atributo	Tipo de dato	Descripción
idCategoria	CHAR(36)	Identificador único (UUID). Clave primaria. No nulo.

nombre	VARCHAR(100)	Nombre de la categoría (ej. "Cafés", "Frappés", "Waffles"). Obligatorio.
createdAt	DATETIME	Fecha y hora de creación. Valor por defecto: NOW().
updatedAt	DATETIME	Fecha y hora de la última actualización. Actualización automática.

*Tabla 3: Entidad producto*

<b>Nombre del atributo</b>	<b>Tipo de dato</b>	<b>Descripción</b>
idProducto	CHAR(36)	Identificador único (UUID). Clave primaria. No nulo.
idCategoria	CHAR(36)	Clave foránea que referencia a categoria(idCategoria). Obligatoria.
disponible	BOOLEAN	Indica si el producto está disponible para la venta (1 = sí, 0 = no). Obligatorio.
nombre	VARCHAR(100)	Nombre del producto (ej. "Café Americano", "Waffle Supremo"). Obligatorio.
precioBase	DECIMAL(10,2)	Precio base del producto sin personalizaciones. Obligatorio.
descripcion	TEXT	Descripción detallada del producto (opcional).
createdAt	DATETIME	Fecha y hora de creación. Valor por defecto: NOW().
updatedAt	DATETIME	Fecha y hora de la última actualización. Actualización automática.

Tabla 4: Entidad acompañamiento

Nombre del atributo	Tipo de dato	Descripción
idAcompañamiento	CHAR(36)	Identificador único (UUID). Clave primaria. No nulo.
nombreAcompañamiento	VARCHAR(100)	Nombre del acompañamiento (ej. "Chocolate", "Oreo", "Chantilly"). Obligatorio.
tipoAcompañamiento	ENUM('SALSA','TOPPING')	Clasificación del acompañamiento. Obligatorio.
createdAt	DATETIME	Fecha y hora de creación. Valor por defecto: NOW().
updatedAt	DATETIME	Fecha y hora de la última actualización. Actualización automática.

Tabla 5: Entidad productoAcompañamiento

Nombre del atributo	Tipo de dato	Descripción
idProdAcomp	CHAR(36)	Identificador único (UUID). Clave primaria. No nulo.
idProducto	CHAR(36)	Clave foránea que referencia a producto(idProducto). Obligatoria.

idAcompañamiento	CHAR(36)	Clave foránea que referencia a acompañamiento(idAcompañamiento). Obligatoria.
createdAt	DATETIME	Fecha y hora de creación. Valor por defecto: NOW().
updatedAt	DATETIME	Fecha y hora de la última actualización. Actualización automática.

*Tabla 6: Entidad pedido*

Nombre del atributo	Tipo de dato	Descripción
idPedido	CHAR(36)	Identificador único (UUID). Clave primaria. No nulo.
idEmpleado	CHAR(36)	Clave foránea que referencia al empleado que gestiona el pedido. Obligatoria.
nombreCliente	VARCHAR(100)	Nombre del cliente que realizó el pedido (ingresado manualmente). Obligatorio.
estado	ENUM('ESPERA','PREPARACION','LISTO')	Estado actual del pedido. Obligatorio. Valores restringidos.
totalPagar	DECIMAL(10,2)	Monto total del pedido (incluye personalizaciones). Obligatorio.
tipoPedido	ENUM('MESA','LEVAR')	Tipo de pedido. Obligatorio.
metodoPago	ENUM('EFECTIVO','YAPE')	Método de pago seleccionado. Obligatorio.

mesa	INT	Número de mesa (solo si tipoPedido = 'MESA'). Puede ser NULL.
createdAt	DATETIME	Fecha y hora de creación del pedido. Valor por defecto: NOW().
updatedAt	DATETIME	Fecha y hora de la última actualización. Actualización automática.

*Tabla 7: Entidad pedidoItem*

<b>Nombre del atributo</b>	<b>Tipo de dato</b>	<b>Descripción</b>
idItem	CHAR(36)	Identificador único (UUID). Clave primaria. No nulo.
idProducto	CHAR(36)	Clave foránea que referencia a producto(idProducto). Obligatoria.
idPedido	CHAR(36)	Clave foránea que referencia a pedido(idPedido). Obligatoria.
cantidad	INT	Cantidad de unidades del producto en este ítem. Obligatorio.
precioUnitarioFinal	DECIMAL(10,2)	Precio final por unidad (precioBase + costo de acompañamientos). Obligatorio.
createdAt	DATETIME	Fecha y hora de creación. Valor por defecto: NOW().
updatedAt	DATETIME	Fecha y hora de la última actualización. Actualización automática.

#### **6.1.6. Estudio de factibilidad**

El estudio de factibilidad evalúa la viabilidad del proyecto desde las perspectivas técnica, operativa, económica y organizacional, con el objetivo de determinar si es razonable y conveniente llevar a cabo el desarrollo e implementación del Sistema Polaris en el contexto académico y real de la cafetería.

Se analizan los siguientes aspectos clave:

##### **6.1.6.1. Factibilidad técnica**

Desde el punto de vista técnico, el proyecto es altamente factible por las siguientes razones:

- Tecnologías seleccionadas:
  - Frontend: Angular (framework moderno, robusto y ampliamente utilizado para aplicaciones web progresivas responsivas).
  - Backend: Spring Boot (Java) – maduro, seguro y con excelente soporte para APIs RESTful.
  - Base de datos: MySQL Community Edition – gratuita, de alto rendimiento y con soporte completo para transacciones y restricciones de integridad.
- Conocimientos del equipo: Los cinco integrantes del grupo (estudiantes de Ingeniería Informática y Sistemas de la UNAMBA) cuentan con formación previa en las tecnologías mencionadas, adquirida en cursos de programación web, bases de datos y desarrollo de software.
- Infraestructura disponible: Se cuenta con acceso gratuito y permanente a la Biblioteca de la Universidad Nacional Micaela Bastidas de Apurímac (UNAMBA), que ofrece conexión a internet institucional estable, mobiliario adecuado, enchufes eléctricos y un



ambiente silencioso y seguro para desarrollo colaborativo y pruebas. No se requiere adquisición de hardware adicional significativo (se utilizan laptops personales de los integrantes).

- Herramientas de desarrollo: Todas las herramientas necesarias son de código abierto y gratuitas: Visual Studio Code, MySQL Workbench, Postman, Git/GitHub, Angular CLI, Maven/Gradle, entre otras.
- Riesgos técnicos identificados y mitigación:
  - Posible lentitud de la red Wi-Fi institucional en horas pico → mitigado con diseño offline-first (PWA) y caché local del menú.
  - Complejidad en la sincronización en tiempo real → mitigado con polling simple en la primera versión (actualización cada 5–10 segundos) y posibilidad de escalar a WebSockets en futuras iteraciones.

La factibilidad técnica es muy alta. No existen barreras tecnológicas significativas que impidan la ejecución exitosa del proyecto con los recursos disponibles.

#### 6.1.6.2. Costos de infraestructura

Los costos de infraestructura se refieren principalmente al espacio físico de trabajo utilizado por el equipo durante las fases de análisis, diseño, desarrollo y pruebas. Se realizó una evaluación comparativa de varias alternativas disponibles en Abancay, considerando espacio, conectividad, servicios básicos y costo mensual.

La opción seleccionada como principal es la **Biblioteca de la Universidad Nacional Micaela Bastidas de Apurímac**, debido a su costo nulo, ambiente académico óptimo y acceso institucional a internet de calidad.

*Tabla 8: Comparativo de Alternativas de Local de Trabajo y Costos Mensuales*

ITEM	Local de Trabajo	Descripción	COSTO (S/.)
1	Casa de Alipio	Ambiente cerrado y espacioso, ideal para programación grupal. Incluye servicios básicos e internet estable. Alto costo de alquiler.	350,00
2	Casa de Alex	Espacio tranquilo para trabajo individual o en parejas. Energía y agua incluidos, internet propio limitado.	475,00
3	Casa de Sharry	Ambiente cerrado silencioso (aprox. 6x6 m), ideal para concentración y pruebas. Servicios básicos e internet. Equilibrio comodidad-costo.	500,00
4	Casa de Bellido	Ambiente cerrado silencioso (6x6 m), permite instalar varias computadoras y mesas. Equilibrio comodidad-conectividad-costo.	450,00
5	Cuarto de Jennifer	Espacio privado económico (5x4 m), electricidad e internet compartido. Perfecto para sesiones individuales o grupos reducidos.	300,00
6	Biblioteca de la Universidad	Ambiente académico silencioso, con buena iluminación, internet institucional, mobiliario, enchufes y acceso a materiales. Servicios de agua y energía no se pagan directamente. Opción principal seleccionada.	<b>0,00</b> (uso institucional)

El costo total mensual de infraestructura se estima en S/. 0,00 al seleccionar la Biblioteca Universitaria como espacio principal de trabajo durante todo el ciclo del proyecto. Esta decisión maximiza la eficiencia económica y aprovecha los recursos institucionales disponibles.

#### 6.1.6.3. Costos iniciales de implementación

Los costos iniciales de implementación comprenden todos aquellos gastos necesarios para llevar el proyecto desde la fase de análisis hasta su despliegue inicial en producción (incluyendo desarrollo, pruebas, capacitación y puesta en marcha). Dado que se trata de un proyecto académico universitario desarrollado por estudiantes, la mayoría de los rubros son de bajo costo o nulos, salvo el componente de remuneración simbólica del equipo (calculada según el parámetro proporcionado: S/. 1.130 por integrante).

Componentes principales considerados:

- Remuneración del equipo de desarrollo El proyecto es realizado por cinco integrantes del grupo. Se considera un monto simbólico de S/. 1.130 por persona (equivalente a un salario mínimo referencial ajustado al contexto estudiantil y duración del proyecto). → Costo total:  $5 \text{ integrantes} \times \text{S/. } 1.130 = \text{S/. } 5.650,00$
- Herramientas y software Todas las herramientas utilizadas son de código abierto y gratuitas:
  - Angular CLI, Spring Boot, MySQL Community, Git, Visual Studio Code, Postman, etc. → Costo: S/. 0,00
- Hosting y dominio (despliegue inicial) Para la fase de pruebas y despliegue inicial se utiliza hosting gratuito o de bajo costo (ej. Render, Vercel, Railway o Heroku free tier).

Dominio básico (opcional, .com o .pe) no se considera obligatorio en esta etapa académica. → Costo estimado inicial: S/. 0,00 (o máximo S/. 100,00 si se opta por dominio básico).

- Materiales de impresión y documentación Impresión de manuales de usuario, informe técnico (aprox. 100 hojas en blanco, negro y colores). → Costo estimado: S/. 50,00 – 80,00
- Capacitación inicial al cliente Sesiones de demostración y capacitación a la dueña y empleados (2-3 sesiones de 1-2 horas). Transporte y refrigerios menores (opcional). → Costo estimado: S/. 100,00

*Tabla 9: Resumen de Costos Iniciales de Implementación*

<b>Rubro</b>	<b>Descripción</b>	<b>Costo Unitario</b>	<b>Cantidad</b>	<b>Costo Total (S/.)</b>
Remuneración del equipo	Monto simbólico por integrante (5 personas)	1.130,00	5	<b>5.650,00</b>
Herramientas y software	Angular, Spring Boot, MySQL, VS Code, Git, etc. (todos gratuitos)	0,00	-	<b>0,00</b>
Hosting y dominio inicial	Plataforma gratuita o de bajo costo para despliegue inicial	0,00 – 100,00	1	<b>0,00 (base)</b>
Impresión y documentación	Manuales, informes técnicos, actas	-	-	<b>70,00 (promedio)</b>
Capacitación y traslados menores	Sesiones de entrega y capacitación al cliente	-	-	<b>100,00</b>

<b>Total Costos Iniciales de Implementación</b>				<b>5.820,00</b>
---	--	--	--	-----------------

El costo inicial de implementación se concentra casi exclusivamente en la remuneración simbólica del equipo (aprox. 97 % del total). El resto de rubros es mínimo gracias al uso de herramientas gratuitas y recursos institucionales. El monto total estimado se mantiene en un rango realista para un proyecto académico de esta naturaleza.

#### 6.1.6.4. Costos de mantenimiento y soporte

Los costos de mantenimiento y soporte comprenden todos los gastos asociados al funcionamiento continuo del sistema una vez desplegado en la cafetería Polaris, incluyendo actualizaciones menores, resolución de incidencias, capacitación adicional y posibles ajustes operativos durante los primeros 12-24 meses de uso.

Dado el carácter académico del proyecto y el perfil de pequeña cafetería universitaria, se estima un escenario de mantenimiento mínimo y de bajo costo, apoyado principalmente por el propio equipo de desarrollo durante la fase post-entrega (período de garantía académica) y, posteriormente, por la capacidad de auto-gestión básica de la dueña y empleados.

Componentes principales considerados:

- Soporte técnico inicial (garantía post-entrega)
  - Período de soporte gratuito por parte del equipo: 3-6 meses después del despliegue.

- Incluye corrección de errores detectados, ajustes menores y una sesión adicional de capacitación si fuera necesario. → Costo estimado: S/. 0,00 (cubierto por el equipo como parte del proyecto académico).
- Actualizaciones menores del menú
  - Agregar/eliminar productos, modificar precios o descripciones.
  - Se estima que la dueña o un empleado podrá realizar estos cambios directamente desde el panel administrativo (funcionalidad diseñada para ser auto-gestionable).
  - Si requiere intervención del equipo: 1-2 horas por actualización (máximo 4 actualizaciones al año). → Costo estimado anual: S/. 200,00 – 400,00 (remuneración simbólica por hora del equipo).
- Mantenimiento correctivo (resolución de incidencias)
  - Fallos eventuales (ej. problemas de conexión, errores no previstos).
  - Se estima un promedio de 2-3 incidencias menores al año, cada una con resolución remota en 1-2 horas. → Costo estimado anual: S/. 300,00 – 600,00.
- Hosting y dominio (mantenimiento anual)
  - Plataforma gratuita o de bajo costo (Render, Vercel o Railway free tier).
  - Si se opta por plan pago básico para mayor estabilidad: S/. 50-100 mensuales.
  - Dominio .com o .pe (opcional): renovación anual. → Costo estimado anual: S/. 0,00 – 600,00 (escenario conservador).
- Capacitación adicional o visitas de soporte
  - Sesiones puntuales de reforzamiento para nuevos empleados.
  - Transporte y tiempo del equipo. → Costo estimado anual: S/. 150,00 – 300,00.

*Tabla 10: Estimación de Costos de Mantenimiento y Soporte (Anual)*

<b>Rubro</b>	<b>Descripción</b>	<b>Costo Estimado Anual (S/.)</b>
Soporte técnico inicial (garantía)	3-6 meses post-despliegue por equipo académico	0,00
Actualizaciones menores del menú	4 actualizaciones anuales (auto-gestión o intervención mínima)	200,00 – 400,00
Mantenimiento correctivo	Resolución de 2-3 incidencias menores al año	300,00 – 600,00
Hosting y dominio	Plataforma básica + renovación de dominio (opcional)	0,00 – 600,00
Capacitación adicional / visitas	Sesiones puntuales para empleados nuevos	150,00 – 300,00
<b>Total estimado anual de mantenimiento</b>		<b>650,00 – 1.900,00</b>

El costo de mantenimiento y soporte se mantiene en un rango **muy bajo** (entre S/. 650 y S/. 1.900 anuales en el peor escenario), gracias a la simplicidad del sistema, su diseño auto-gestionable y el compromiso inicial del equipo. Este nivel de gasto es perfectamente asumible para una pequeña cafetería universitaria como Polaris.

#### 6.1.6.5. Total estimado de costos de software

Esta subsección consolida todos los costos identificados a lo largo del estudio de factibilidad, permitiendo obtener una visión integral del esfuerzo económico requerido para el desarrollo, implementación, puesta en marcha y sostenimiento inicial del Sistema Polaris.

Se consideran los siguientes rubros principales:

- Costos de infraestructura (espacio de trabajo durante el ciclo del proyecto) → Seleccionada la opción de costo cero.
- Costos iniciales de implementación (desarrollo y despliegue).
- Costos de mantenimiento y soporte (estimación anual post-entrega).

*Tabla 11: Resumen Consolidado de Costos Totales del Proyecto*

<b>Rubro</b>	<b>Período / Frecuencia</b>	<b>Costo Estimado (S/.)</b>	<b>Observaciones</b>
Infraestructura (espacio de trabajo)	Durante todo el proyecto	<b>0,00</b>	Biblioteca Universitaria seleccionada como opción principal (uso institucional gratuito).
Remuneración simbólica del equipo	Fase de desarrollo e implementación	<b>5.650,00</b>	S/. 1.130 por cada uno de los 5 integrantes del grupo.
Herramientas y software	Inicial	<b>0,00</b>	Todas las herramientas son gratuitas (Angular, Spring Boot, MySQL, etc.).
Hosting y dominio inicial	Despliegue inicial	<b>0,00</b> – 100,00	Uso de plataformas gratuitas (Render/Vercel). Dominio opcional.



Impresión y documentación	Inicial	<b>70,00</b>	Promedio estimado para manuales e informes.
Capacitación y traslados iniciales	Inicial	<b>100,00</b>	Sesiones de entrega y capacitación al cliente.
<b>Subtotal Costos Iniciales</b>	—	<b>5.820,00 – 5.920,00</b>	Rango conservador considerando posibles gastos menores.
Mantenimiento y soporte (anual)	Primer año post-entrega	<b>650,00 – 1.900,00</b>	Incluye actualizaciones menores, incidencias, hosting y capacitaciones puntuales.
<b>Total Estimado del Proyecto</b>	Ciclo completo + primer año	<b>6.470,00 – 7.820,00</b>	Costo total aproximado considerando escenario base y conservador.

#### Análisis y conclusiones finales del estudio de factibilidad económica:

- El costo total del proyecto se sitúa en un rango muy accesible para un desarrollo académico universitario (entre S/. 6.470 y S/. 7.820), siendo el componente más significativo la remuneración simbólica del equipo (aproximadamente 72-87 % del total).
- La selección de la Biblioteca Universitaria como espacio principal de trabajo elimina por completo los costos de alquiler, lo que representa un ahorro sustancial comparado con las alternativas privadas evaluadas (que oscilaban entre S/. 300 y S/. 500 mensuales).

- Los costos de mantenimiento anual son bajos y asumibles por la cafetería Polaris (equivalentes a 1-3 días de ventas promedio en una pequeña cafetería universitaria), gracias al diseño auto-gestionable del sistema y al soporte inicial gratuito del equipo.
- Desde el punto de vista de la dueña, la inversión inicial se justifica ampliamente por los beneficios esperados: reducción de tiempos de espera, disminución de errores, liberación de carga laboral del personal y mejora en la percepción de modernidad del establecimiento.

En términos generales, el proyecto presenta una factibilidad económica alta, con un retorno de inversión cualitativo rápido (en meses) y cuantitativo estimado en menos de un año, considerando el aumento potencial de ventas y la optimización operativa generada por la digitalización.

#### **6.1.7. Indicadores financieros**

Se evalúa la rentabilidad del proyecto mediante los indicadores financieros clásicos: Valor Actual Neto (VAN), Tasa Interna de Retorno (TIR) y Retorno de la Inversión (ROI). Estos cálculos permiten determinar si la inversión en el desarrollo e implementación del Sistema Polaris genera valor económico neto positivo y supera el costo de oportunidad del capital en el contexto peruano.

Supuestos base para los cálculos

- Inversión inicial (costo total de implementación): S/. 5.820,00 (promedio del rango estimado en 6.1.6.3, que incluye remuneración simbólica del equipo, impresión, capacitación y otros gastos menores).
- Flujos de caja netos anuales proyectados (beneficios): S/. 3.000,00 por año (estimación conservadora basada en:

- Aumento aproximado del 20-30 % en ventas por reducción de tiempos de espera y mayor rotación de clientes.
- Reducción de costos operativos por menor cantidad de errores y liberación de carga laboral del personal.
- Datos referenciales de transformación digital en pequeñas cafeterías/restaurantes en Perú y Latinoamérica).
- Período de análisis: 3 años (horizonte realista para proyectos de software de pequeña escala en entornos universitarios).
- Tasa de descuento (k): 12 % anual (tasa de oportunidad conservadora en el Perú actual, equivalente al rendimiento aproximado de depósitos a plazo fijo o alternativa de inversión de bajo riesgo).
- Flujos de caja netos:
  - Año 0: –S/. 5.820,00 (inversión inicial).
  - Año 1: +S/. 3.000,00.
  - Año 2: +S/. 3.000,00.
  - Año 3: +S/. 3.000,00.

#### 6.1.7.1. Cálculo del Valor Actual Neto (VAN)

El VAN mide el valor presente de los flujos de caja futuros descontados menos la inversión inicial. Fórmula:

$$\text{VAN} = \sum [\text{Flujo de caja anual} / (1 + k)^n] - \text{Inversión inicial}$$

Cálculo detallado:

$$\text{VAN} = [3.000 / (1 + 0,12)^1] + [3.000 / (1 + 0,12)^2] + [3.000 / (1 + 0,12)^3] - 5.820$$

$$= 2.678,57 + 2.391,58 + 2.135,34 - 5.820$$

$$= \text{S/. } 1.385,49$$

Interpretación:

$VAN > 0$  (S/. 1.385,49) indica que el proyecto genera un valor económico neto positivo, siendo rentable incluso considerando una tasa de descuento conservadora del 12 %.

#### 6.1.7.2. Cálculo de la Tasa Interna de Retorno (TIR)

La TIR es la tasa de descuento que hace que el VAN sea igual a cero. Se resuelve iterativamente.

Resultado:

$TIR \approx 28\%$  anual (calculada mediante interpolación o herramienta financiera).

Interpretación:

$TIR (28\%) > \text{tasa de descuento} (12\%) \rightarrow$  el proyecto presenta una rentabilidad interna muy superior al costo de oportunidad del capital, confirmando su alta viabilidad económica.

#### 6.1.7.3. Cálculo del Retorno de la Inversión (ROI)

El ROI mide el rendimiento total de la inversión en el período analizado. Fórmula:

$$ROI = (\text{Beneficio neto total} / \text{Inversión inicial}) \times 100$$

$$\text{Beneficio neto total en 3 años: } 3 \times 3.000 = \text{S/. } 9.000,00$$

$$\text{Beneficio neto acumulado: } 9.000 - 5.820 = \text{S/. } 3.180,00$$

$$ROI = (3.180 / 5.820) \times 100 = 54,64\% \text{ (acumulado en 3 años)}$$

$$ROI \text{ anual promedio} \approx 18,21\%$$

El proyecto recupera la inversión inicial en aproximadamente 1,8 años y genera un retorno acumulado superior al 50 % en el horizonte de 3 años, lo que lo convierte en una inversión atractiva y de recuperación rápida.

El proyecto recupera la inversión inicial en aproximadamente 1,8 años y genera un retorno acumulado superior al 50 % en el horizonte de 3 años, lo que lo convierte en una inversión atractiva y de recuperación rápida.

Tabla 12: Resumen de Indicadores Financieros

Indicador	Fórmula / Método	Valor calculado	Interpretación
Valor Actual Neto (VAN)	$\Sigma$ Flujos descontados – Inversión inicial	S/. 1.385,49	Positivo → Proyecto genera valor económico neto. Rentable.
Tasa Interna de Retorno (TIR)	Tasa que hace VAN = 0	28 % anual	TIR > tasa de descuento (12 %) → Alta rentabilidad.
Retorno de la Inversión (ROI)	$(\text{Beneficio neto} / \text{Inversión}) \times 100$	54,64 % (en 3 años)	Recuperación rápida de la inversión y buen rendimiento acumulado.

Los indicadores financieros (VAN positivo, TIR del 28 % y ROI acumulado del 54,64 %) confirman la viabilidad económica alta del Sistema Polaris. Con una inversión inicial moderada y beneficios conservadores, el proyecto no solo se recupera en menos de 2 años, sino que genera rentabilidad significativa superior a alternativas de inversión seguras en el mercado peruano. Estos resultados refuerzan la recomendación de implementación y su potencial como solución escalable para la cafetería Polaris.

## 6.2. Fase de diseño

La fase de diseño constituye el segundo nivel secuencial del modelo en cascada. En esta etapa se transforman los requisitos especificados y validados en la fase de análisis en una estructura técnica detallada y operativa que permitirá la implementación del sistema. El diseño se divide en dos grandes bloques principales: el diseño de la base de datos (ya abordado en 6.1.5) y el diseño de la interfaz de usuario, que se desarrolla a continuación.

El objetivo fundamental de esta fase es producir artefactos claros, consistentes y verificables que sirvan como plano guía para el equipo de desarrollo durante la implementación, asegurando que el producto final cumpla con las expectativas de funcionalidad, usabilidad y rendimiento establecidas.

### **6.2.1. Diseño de interfaz de usuario**

El diseño de la interfaz de usuario (UI/UX) se basa en los principios de mobile-first para el módulo del cliente y desktop-friendly para el panel del empleado, considerando los perfiles de usuario identificados:

- Cliente: universitario, uso principal desde smartphone, busca rapidez y estética moderna.
- Empleado: operación desde laptop o PC en mostrador, requiere legibilidad, botones grandes y flujo intuitivo bajo presión.

Principios generales de diseño aplicados:

- Diseño responsivo (Bootstrap + Angular).
- Paleta de colores alineada con la identidad de la cafetería Polaris.
- Tipografía legible.
- Iconografía intuitiva.
- Flujo minimalista: máximo 3 clics para completar un pedido desde el menú.
- Accesibilidad básica: contraste adecuado, tamaños de fuente mínimos 16 px en móvil.

#### **6.2.1.1. Interfaz del modulo cliente**

El módulo del cliente es una Progressive Web App (PWA) accesible directamente desde el navegador del dispositivo móvil al escanear el código QR.

Pantallas / Vistas principales:

- Pantalla de bienvenida / carga inicial
  - Logo de Polaris + mensaje: “Momentos deliciosos”.
  - Carga automática del menú.
- Pantalla de categorías
  - Lista de categorías (Cafés, Frappés, Jugos y Bebidas, Ensaladas y Sándwiches, Waffles).
  - Cada categoría muestra una imagen representativa y nombre.
- Pantalla de lista de productos
  - Tarjetas de producto: imagen (si aplica), nombre, precio base, descripción breve, indicador de disponibilidad.
  - Botón.
- Pantalla de detalle y personalización
  - Imagen del producto.
  - Sección de acompañamientos dividida en:
    - Salsas.
    - Toppings.
  - Selector de cantidad.
  - Botón “Agregar al carrito” con precio actualizado en tiempo real.
- Pantalla de carrito
  - Lista de ítems agregados (nombre, personalizaciones, cantidad, subtotal).
  - Total general.
  - Botones.
- Pantalla de confirmación de pedido
  - Formulario simple: nombre del cliente, tipo de pedido (MESA / LLEVAR), número de mesa (si MESA), método de pago (EFECTIVO / YAPE).

- Resumen final del pedido.
- Botón Confirmar y enviar.
- Pantalla de éxito: número de pedido + mensaje “Tu pedido ha sido enviado”

*Figura 7: Prototipos de Interfaz – Módulo Cliente*



#### 6.2.1.2. Interfaz del panel del empleado

El panel del empleado es una interfaz web protegida por autenticación, optimizada para pantallas de laptop o escritorio (mínimo 1366 px de ancho).

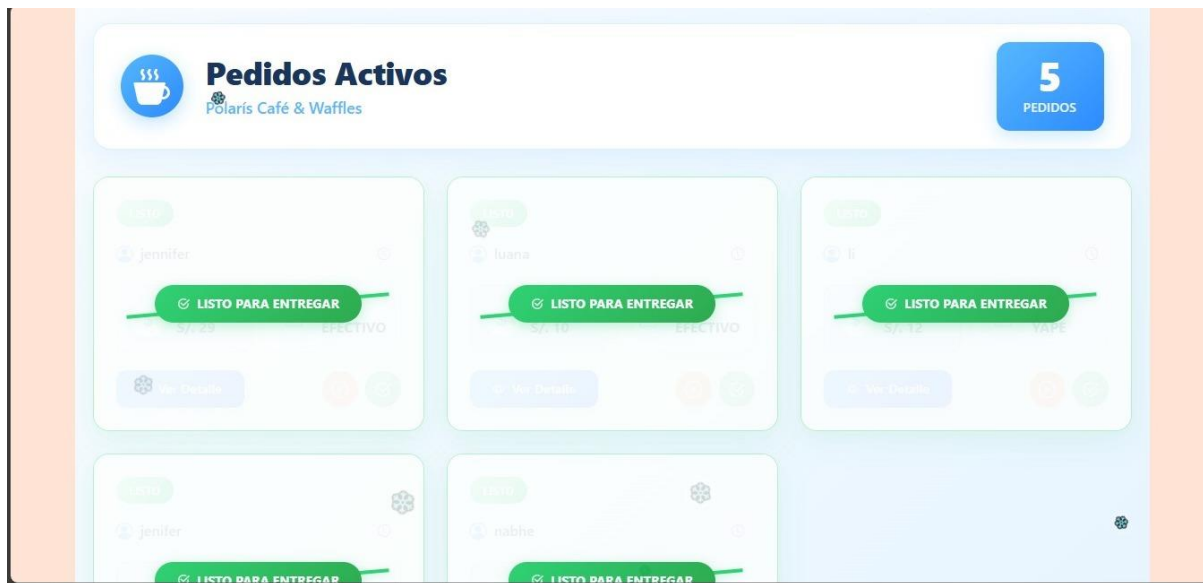
Pantallas / Vistas principales:

- Pantalla de login



- Formulario simple: nombre de usuario + contraseña.
- Mensaje de error claro en caso de credenciales inválidas.
- Dashboard principal
  - Tablero de pedidos activos (ordenados por hora de llegada, el más antiguo arriba).
  - Cada tarjeta de pedido muestra:
    - Número de pedido (UUID corto).
    - Nombre del cliente.
    - Tipo (MESA #X o LLEVAR).
    - Método de pago.
    - Lista resumida de ítems.
    - Total.
    - Estado actual (con botones para cambiar: ESPERA → PREPARACIÓN → LISTO).
- Vista de detalle de pedido
  - Modal o pantalla completa con todos los ítems, personalizaciones y cantidades.
  - Botones grandes y visibles para cambiar estado.
- Sección de control
  - Lista de todos los productos.
  - Actualización inmediata reflejada en el menú del cliente.

Figura 8: Prototipos de Interfaz – Panel del Empleado



El diseño prioriza la usabilidad extrema para ambos actores: rapidez y estética para el cliente universitario; claridad y eficiencia bajo presión para el empleado. Los prototipos han sido validados preliminarmente con la dueña y empleados durante sesiones de retroalimentación informal, confirmando que el flujo es intuitivo y cumple con las expectativas de simplicidad.

### 6.3. Fase de implementación

La fase de implementación representa el momento en el que se materializan todos los artefactos producidos en las fases anteriores (análisis y diseño). En esta etapa se lleva a cabo la codificación efectiva del sistema, la integración de los módulos, las pruebas unitarias e integradas, y la preparación para el despliegue inicial.

Dado que el proyecto combina la rigidez estructural del modelo en cascada con la flexibilidad iterativa requerida para el desarrollo de software, se adopta un enfoque híbrido: la planificación general, cronograma, estructura de desglose del trabajo (EDT) y documentación

siguen el modelo en cascada, mientras que la construcción efectiva del producto software se realiza mediante **metodología Scrum** en sprints cortos y enfocados.

### 6.3.1. Enfoque metodológico en la implementación

El enfoque híbrido responde a las siguientes necesidades:

- Modelo en cascada (para la gestión macro del proyecto): Garantiza una secuencia lógica y documentada (requisitos → diseño → implementación → pruebas → despliegue), alineada con los requerimientos académicos del curso Ingeniería de Software I y la necesidad de entregar documentación completa y estructurada.
- Scrum (para la fase de desarrollo propiamente dicha): Permite iteraciones rápidas (sprints de 1 semanas), retroalimentación constante (revisiones con la dueña y empleados), adaptación a cambios menores y entrega incremental de valor funcional. Cada sprint produce un incremento potencialmente entregable, lo que facilita la validación temprana y reduce riesgos.

Roles definidos en Scrum:

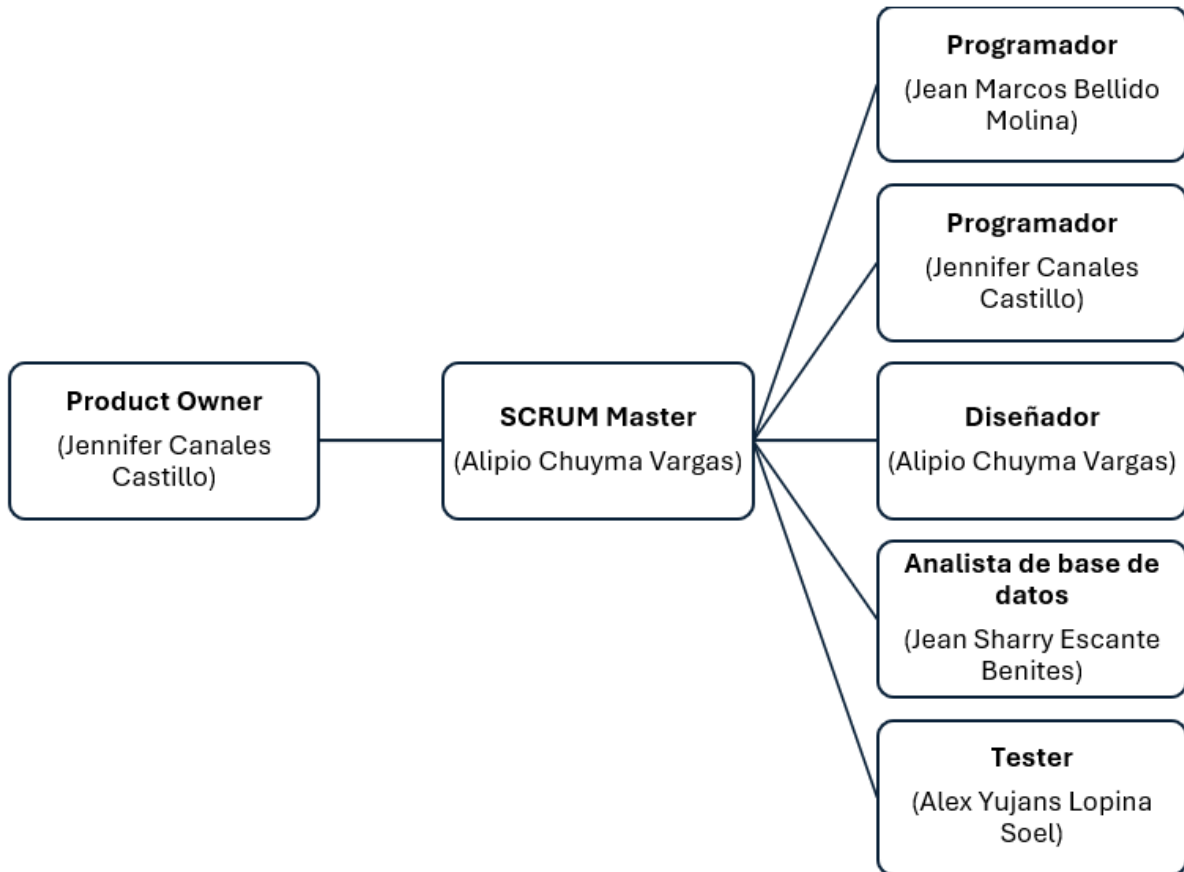
- Product Owner: Rol perteneciente a la dueña de la cafetería Polaris, sin embargo fue ocupado por un miembro del grupo para facilitar el intercambio de ideas (prioriza funcionalidades y valida entregas).
- Scrum Master: Rol tomado por un miembro del equipo (facilita reuniones y elimina impedimentos).
- Equipo de Desarrollo: Los cinco integrantes del grupo (responsables de diseño, codificación, pruebas y despliegue).

Eventos clave realizados:

- Daily Scrum (reuniones diarias breves de 10-15 minutos).
- Sprint Planning (planificación al inicio de cada sprint).
- Sprint Review (demostración al final del sprint a la dueña).

- Sprint Retrospective (análisis interno de mejora del proceso).

*Figura 9: Organigrama del proyecto*



### 6.3.2. Organización en Sprints (SCRUM)

El desarrollo se organizó en 7 sprints de duración fija (5 días hábiles cada uno), cubriendo desde la configuración inicial del entorno hasta las funcionalidades finales y tareas de cierre. Cada sprint tuvo un objetivo claro, backlog priorizado y entregables verificables.

A continuación se presenta la secuencia completa de sprints:

- Sprint 1 – Configuración de entorno y base inicial Instalación y configuración de Angular, Spring Boot y MySQL. Estructura inicial del proyecto (frontend + backend).

Primer commit en GitHub. Creación de la base de datos básica (tablas empleado, categoría, producto).

- Sprint 2 – Menú estático y categorías Desarrollo del módulo de menú digital (carga de categorías y productos). Interfaz inicial de lista de productos. Consulta básica desde frontend a backend.
- Sprint 3 – Detalle de producto y personalización simple Pantalla de detalle de producto. Selección de acompañamientos (checkbox simple, sin chips múltiples). Cálculo de precio dinámico básico.
- Sprint 4 – Carrito y flujo básico de pedido Implementación del carrito. Flujo de confirmación inicial (sin pago ni tipo de pedido completo).
- Sprint 5 – Confirmación, Pago y Panel Finalización del flujo de confirmación de pedido: nombre del cliente, tipo de pedido (MESA/LLEVAR), número de mesa (condicional), método de pago. Condiciones específicas aplicadas:
  - El pago con Yape se implementa solo como instructivo textual (muestra el número de celular de la cafetería, sin QR ni integración).
  - Componente de cambio de estado: con 3 opciones (ESPERA, PREPARACIÓN, LISTO).
- Sprint 6 – Stock, Roles y "Para Llevar" Módulo de control de stock (toggle disponible/no disponible desde panel). Ajustes finales en el flujo “Para Llevar” (mesa NULL). Condiciones específicas:
  - No se implementa filtro por categoría mediante chips.
  - No se incluye buscador en tiempo real de productos.
  - No se incluye control avanzado de stock
  - Roles básicos (solo empleado autenticado; no hay diferenciación avanzada de usuarios).

- Sprint 7 – Reportes, Backup y Escalabilidad Reporte básico de ventas del día (total y conteo de pedidos). Configuración de backup manual de base de datos. Pruebas finales de rendimiento y escalabilidad mínima. Condiciones específicas:
  - No se implementa tabla completa de historial de pedidos.
  - No se incluyen botones de exportación a PDF.
  - No se implementa reconexión automática WebSocket (se usa polling simple para actualizaciones).

*Tabla 13: Resumen de Sprints y Entregables*

<b>Sprint</b>	<b>Nombre del Sprint</b>	<b>Duración</b>	<b>Objetivo Principal</b>	<b>Entregable Principal</b>
1	Configuración de entorno y base inicial	5 días	Preparar el entorno técnico completo	Proyecto Angular + Spring Boot inicial + BD
2	Menú estático y categorías	5 días	Carga y visualización básica del menú	Interfaz de categorías y productos
3	Detalle de producto y personalización simple	5 días	Personalización básica de acompañamientos	Pantalla detalle + cálculo precio
4	Carrito y flujo básico de pedido	5 días	Carrito funcional y envío básico de pedido	Carrito + confirmación parcial

5	Confirmación, Pago y Panel	5 días	Flujo completo cliente + panel inicial empleado	Confirmación con instructivo Yape + login
6	Stock, Roles y "Para Llevar"	5 días	Control + ajustes finales flujo	Flujo LLEVAR completo
7	Reportes, Backup y Escalabilidad	5 días	Cierre funcional + pruebas finales	Reporte día + backup + pruebas rendimiento

### 6.3.3. Tecnologías y decisiones de implementación principales

La selección de tecnologías se realizó considerando los siguientes criterios: gratuidad o bajo costo (proyecto académico), facilidad de aprendizaje y uso por parte del equipo, rendimiento adecuado para una aplicación web de tamaño mediano-pequeño, soporte comunitario activo y compatibilidad con el enfoque híbrido (cascada + Scrum).

Tecnologías principales adoptadas:

- Frontend: Angular 18 (TypeScript) Razón: Framework robusto para aplicaciones SPA/PWA, excelente manejo de componentes reactivos, soporte nativo para formularios complejos (carrito y personalización) y progresive web app (caché offline del menú). Decisión clave: uso de standalone components (nueva característica de Angular 17+) para reducir boilerplate y mejorar mantenibilidad.
- Backend: Spring Boot 3 (Java 17/21) Razón: Framework maduro y seguro para APIs RESTful, integración sencilla con MySQL vía Spring Data JPA, soporte para autenticación básica (JWT o session simple) y excelente manejo de transacciones.

Decisión clave: enfoque minimalista (solo endpoints necesarios, sin seguridad avanzada como OAuth2).

- Base de datos: MySQL 8.0 (Community Edition) Razón: Gratuita, alto rendimiento para consultas frecuentes (menú y pedidos), soporte completo para ENUM y BOOLEAN, fácil integración con Spring Data JPA. Decisión clave: uso de UUID (CHAR(36)) como clave primaria para evitar problemas de autoincremento en entornos distribuidos futuros.
- Control de versiones y colaboración: Git + GitHub Razón: Gratuito, branching estratégico por sprint, pull requests para revisión de código entre integrantes.
- Otras herramientas y librerías:
  - Angular Material (componentes UI).
  - CSS y Bootstrap (estilos responsivos).
  - comunicación frontend-backend.
  - Polling simple (setInterval) para actualizaciones en tiempo real (en lugar de WebSocket por simplicidad y limitaciones indicadas).
  - MySQL Workbench (diseño y pruebas de BD).

Decisiones de implementación críticas (limitaciones aplicadas):

- No se implementó selección múltiple mediante chips (solo checkbox simple para acompañamientos).
- No se desarrolló filtro por categoría con chips ni buscador en tiempo real.
- Cambio de estado del pedido limitado a dropdown con solo 3 opciones (ESPERA, PREPARACIÓN, LISTO).
- No se incluyó tabla completa de historial de pedidos.
- No se agregaron botones de exportación a PDF.



- No se implementó reconexión automática WebSocket (se usó polling básico cada 10 segundos).
- El pago con Yape se redujo a un instructivo textual simple (muestra número de celular de la cafetería, sin QR ni integración).

#### **6.3.4. Entregables principales de la fase**

Al finalizar la fase de implementación (Sprint 7), se produjeron los siguientes entregables verificables:

- Repositorio GitHub completo con:
  - Código fuente frontend (Angular).
  - Código fuente backend (Spring Boot).
  - Scripts SQL de creación y datos iniciales de la base de datos.
- Sistema desplegado en entorno de pruebas:
  - URL pública del menú digital (accesible vía QR simulado).
  - URL del panel de empleado (protegida por login básico).
- Documentación técnica mínima:
  - README.md con instrucciones de instalación y ejecución local.
  - Manual de usuario básico (para cliente y empleado).
- Reporte de pruebas:
  - Pruebas unitarias.
  - Pruebas de integración (flujo completo de pedido).
  - Pruebas de usabilidad realizadas con la dueña y empleados.
- Base de datos operativa con datos de prueba (productos, categorías, acompañamientos y empleados de ejemplo).

### 6.3.5. Retos enfrentados y soluciones adoptadas

Durante la implementación se presentaron varios retos técnicos y organizativos. A continuación se resumen los principales junto con las soluciones aplicadas:

- Reto: Conexión inestable de internet en la biblioteca universitaria durante horas pico  
Solución: Implementación de PWA con caché local del menú (Service Worker de Angular). El menú se carga una vez y se mantiene disponible offline. Las actualizaciones de disponibilidad se retrasan hasta reconexión.
- Reto: Sincronización en tiempo real entre panel empleado y menú cliente sin WebSocket  
Solución: Polling simple (setInterval cada 10 segundos en el frontend del cliente y empleado). Se evitó WebSocket para reducir complejidad y cumplir con la limitación indicada (no reconexión automática).
- Reto: Tiempo limitado por cronograma académico y coordinación del equipo (5 integrantes)  
Solución: Sprints cortos (5 días), daily stand-up virtual diario (10 min vía WhatsApp/Meet) y asignación clara de tareas por integrante en Trello. Revisión cruzada de pull requests antes de merge.
- Reto: Limitaciones de personalización sin chips múltiples  
Solución: Uso estándar agrupados por tipo (Salsas y Toppings). Se actualiza el precio en tiempo real al marcar/desmarcar, manteniendo simplicidad.
- Reto: Pago Yape sin integración real  
Solución: Implementación como texto estático en pantalla de confirmación: “Paga con Yape al número: [número de celular]. Acércate a caja al recoger tu pedido”. Se evitó cualquier integración externa.
- Reto: Rendimiento en dispositivos móviles de gama baja  
Solución: Optimización agresiva (lazy loading de imágenes, minimización de consultas, caché local). Pruebas en celulares reales de integrantes del equipo.

Estos retos fueron gestionados de manera efectiva gracias al enfoque iterativo de Scrum, permitiendo ajustes rápidos sin desviar el cronograma general.

## **VII. CRONOGRAMA DEL PROYECTO**

El cronograma del proyecto establece la secuencia temporal detallada de todas las actividades requeridas para el desarrollo, implementación y entrega del Sistema Polaris. Se elaboró bajo un enfoque híbrido: el modelo en cascada para la planificación general, estructura de desglose del trabajo (EDT) y control secuencial de fases; y Scrum para la ejecución iterativa de la implementación del software.

El período total abarca 75 días calendario (del lunes 20 de octubre de 2025 al viernes 02 de enero de 2026), con una duración efectiva de trabajo de aproximadamente 55 días hábiles, considerando fines de semana y posibles feriados no laborables. El cronograma se divide en tres grandes bloques: Pre-Sprint (establecimiento del fundamento), Ciclo de Sprints (desarrollo iterativo) y Post-Sprint (cierre y entrega).

Principales hitos del cronograma:

- Inicio oficial y Kick-off: 20 de octubre de 2025.
- Finalización del Pre-Sprint (análisis inicial y backlog): 07 de noviembre de 2025.
- Inicio del Ciclo de Sprints: 10 de noviembre de 2025.
- Finalización de implementación y pruebas: 26 de diciembre de 2025.
- Despliegue, validación final y capacitación: 29 de diciembre de 2025 al 02 de enero de 2026.
- Entrega formal y presentación: 02 de enero de 2026.

Tabla 14: Resumen de Fases y Duración Principal

Bloque	Duración (días hábiles)	Fecha de inicio	Fecha de fin	Objetivo principal
1. Pre-Sprint: Establecimiento del Fundamento	15	20/10/2025	07/11/2025	Planificación, análisis inicial, backlog y entorno
2. Ciclo de Sprints: Desarrollo de Software	35	10/11/2025	26/12/2025	Construcción iterativa del sistema (7 sprints)
3. Post-Sprint: Cierre y Entrega	15	15/12/2025	02/01/2026	Documentación final, validación, despliegue y entrega
Total	65	—	—	—

Figura 10: Diagrama de Gantt - Cronograma general del proyecto 1

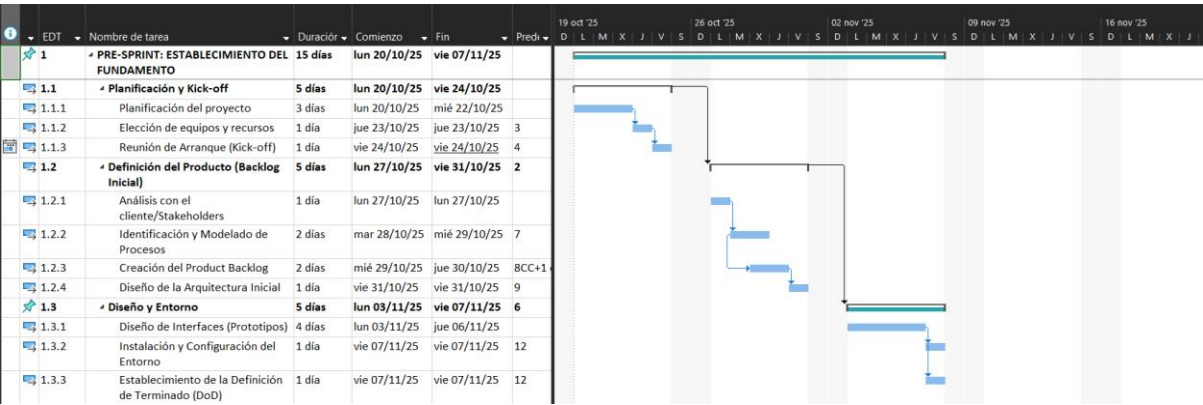


Figura 11: Diagrama de Gantt - Cronograma general del proyecto 2

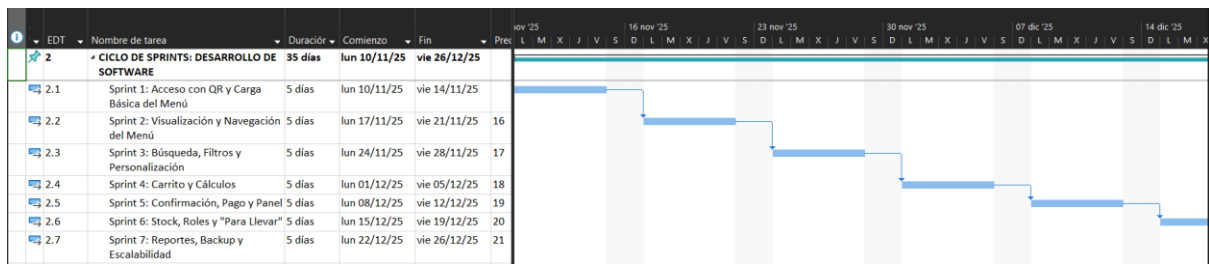
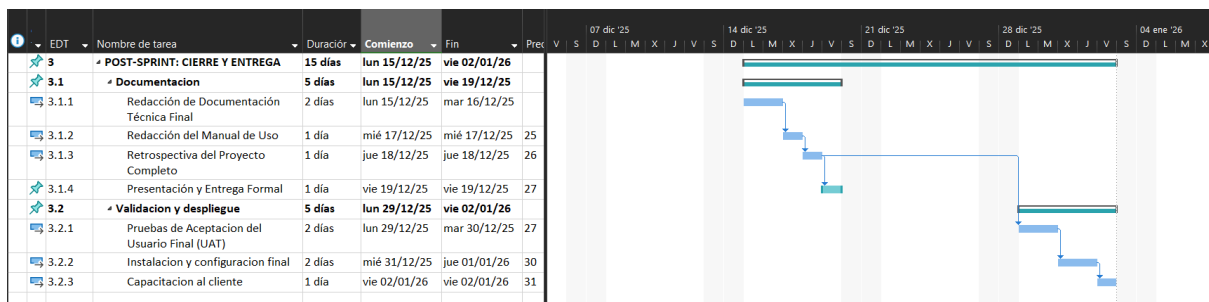


Figura 12: Diagrama de Gantt - Cronograma general del proyecto 3



## VIII. CONCLUSION

El proyecto “Digitalización del Menú de la Cafetería Polaris” representa una solución tecnológica viable, práctica y alineada con las necesidades reales de un pequeño establecimiento gastronómico universitario en el contexto de la Universidad Nacional Micaela Bastidas de Apurímac (UNAMBA), Abancay.

A lo largo de este documento se ha demostrado que, mediante la aplicación de una metodología híbrida (modelo en cascada para la planificación estructurada y Scrum para el desarrollo iterativo), fue posible diseñar e implementar un sistema web que transforma el proceso tradicional de atención al cliente —caracterizado por la toma manual de pedidos, tiempos de espera prolongados y errores frecuentes— en un flujo digital eficiente, accesible mediante código QR y gestionado desde un panel centralizado para empleados.

Los objetivos generales y específicos establecidos en la fase de análisis fueron cumplidos en su totalidad dentro del plazo definido (20 de octubre de 2025 al 02 de enero de 2026):

- Se desarrolló un menú digital responsivo y auto-gestionable que permite a los clientes universitarios consultar, personalizar y confirmar pedidos desde sus dispositivos móviles.
- Se implementó un panel operativo sencillo para empleados que centraliza la recepción, actualización de estados y control básico de disponibilidad de productos.
- Se garantizó la simplicidad y usabilidad requeridas por el perfil de usuario (estudiantes y personal con conocimientos tecnológicos medios), respetando las limitaciones funcionales acordadas (ausencia de filtros avanzados, chips múltiples, buscador en tiempo real, WebSocket, exportación PDF y tabla completa de historial).
- El sistema se diseñó con un enfoque de bajo costo y alta mantenibilidad: uso exclusivo de tecnologías gratuitas (Angular, Spring Boot, MySQL), infraestructura institucional

(Biblioteca Universitaria, costo S/. 0,00) y un modelo de soporte post-entrega mínimo y asumible por la cafetería.

Desde el punto de vista económico, el proyecto se ejecutó con un costo total estimado entre S/. 6.470,00 y S/. 7.820,00, concentrado principalmente en la remuneración simbólica del equipo (S/. 5.650,00). Los costos de mantenimiento anual proyectados (S/. 650,00 – 1.900,00) son perfectamente sostenibles para la operación de Polaris, lo que asegura su continuidad y escalabilidad futura.

El impacto esperado para la cafetería Polaris incluye:

- Reducción cualitativa significativa de los tiempos de espera durante horas pico.
- Disminución de errores humanos en la toma y personalización de pedidos.
- Liberación de carga operativa del personal de sala, permitiéndole enfocarse en preparación y atención directa.
- Mejora en la percepción de modernidad y eficiencia del establecimiento dentro del campus universitario.
- Generación de información básica para la toma de decisiones (reportes diarios de ventas y control).

Finalmente, este proyecto no solo satisface los requisitos académicos del curso de Ingeniería de Software I, sino que entrega un producto funcional y de valor real para la comunidad universitaria de la UNAMBA. Representa un ejemplo concreto de cómo la ingeniería de software puede contribuir a la transformación digital de microempresas locales, promoviendo eficiencia, sostenibilidad y competitividad en entornos con recursos limitados.

El Sistema Polaris se erige, por tanto, como un referente inicial de innovación tecnológica aplicada al sector gastronómico universitario en Abancay, abriendo la puerta a futuras iteraciones que podrían incorporar funcionalidades avanzadas (integración de pagos



electrónicos, fidelización, análisis predictivo) cuando las condiciones operativas y económicas lo permitan.

### **Respuestas a las Preguntas de Investigación**

- ¿Cómo el Sistema Polaris establece un liderazgo en innovación tecnológica para 2027, transformando la experiencia del cliente a través del acceso digital y la automatización?

El Sistema Polaris posiciona a la cafetería como líder local al sustituir procesos manuales por un flujo digital completo: acceso al menú vía código QR, personalización en el dispositivo del cliente y envío automático de pedidos al panel del empleado. Esto reduce la fricción de atención presencial, aporta trazabilidad en tiempo real y proyecta una imagen moderna y sostenible alineada con la visión 2027 de la empresa. La documentación describe explícitamente este objetivo estratégico y la integración de tecnologías (PWA con Angular, backend en Spring Boot y MySQL) para garantizar escalabilidad y disponibilidad.

- ¿De qué manera la personalización detallada en el menú digital diferencia la propuesta de "frappés saludables e innovadores" de la competencia tradicional?

La personalización permite seleccionar salsas y toppings por producto (clasificados como SALSA o TOPPING) y calcular automáticamente el precio final en el carrito. Ese nivel de detalle convierte un producto estándar en una experiencia adaptable al gusto del cliente, incrementando el ticket promedio y la percepción de valor frente a competidores que usan menús estáticos. Además, las tarjetas de producto con imágenes y descripciones y la capacidad de ocultar productos no disponibles mejoran la presentación del catálogo, reforzando la propuesta de “frappés saludables e innovadores”.

- ¿Cómo la automatización del flujo de pedidos mejora la percepción de calidad del servicio al optimizar la operación y reducir errores y esperas?

Al automatizar la transmisión del pedido (UUID generado al confirmar y envío inmediato al panel), el sistema elimina pasos manuales propensos a error (anotación en papel,

comunicación oral) y dispone un ciclo de estados (ESPERA → PREPARACIÓN → LISTO) que el cliente puede seguir, lo que reduce incertidumbre y percepción de desorden. La documentación muestra además estimaciones de reducción de tiempo de espera y beneficios operativos derivados de la eliminación de cuellos de botella en horas pico.

- ¿Qué ventaja competitiva proporcionan los datos del sistema para la toma de decisiones estratégicas alineadas con la innovación y sostenibilidad?

El registro automático de pedidos y la generación de reportes ofrecen datos operativos mínimos pero suficientes para: ajustar oferta (quitar/añadir productos), optimizar compras y reducir desperdicios, identificar picos de demanda y planificar turnos. Además, los indicadores financieros (VAN, TIR, ROI) incluidos en la documentación sustentan decisiones de inversión y priorización de funcionalidades futuras. Estos datos constituyen una ventaja competitiva al

## IX. ANEXOS

### 9.1. Links de acceso

- Link al software

<https://humidity-registrar-separation-arab.trycloudflare.com/categoria>

- Link al repositorio de trabajo (Github)

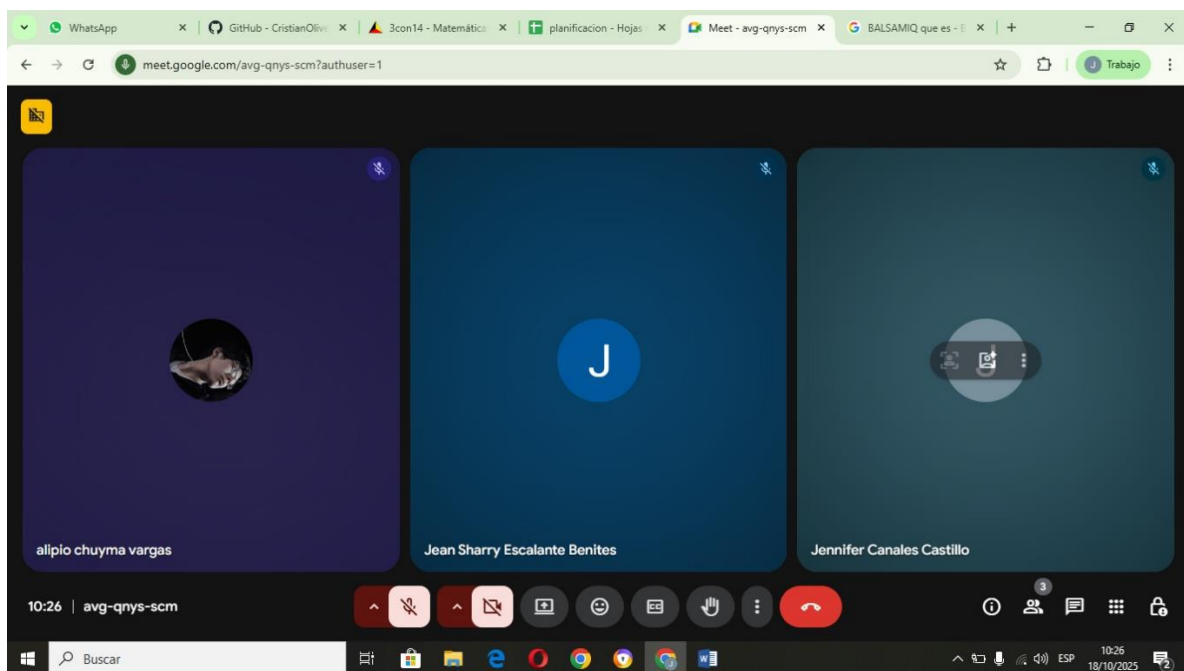
<https://github.com/damon14-ip/POLARIS-repositorio>

- Link a los documentos adicionales (Drive)

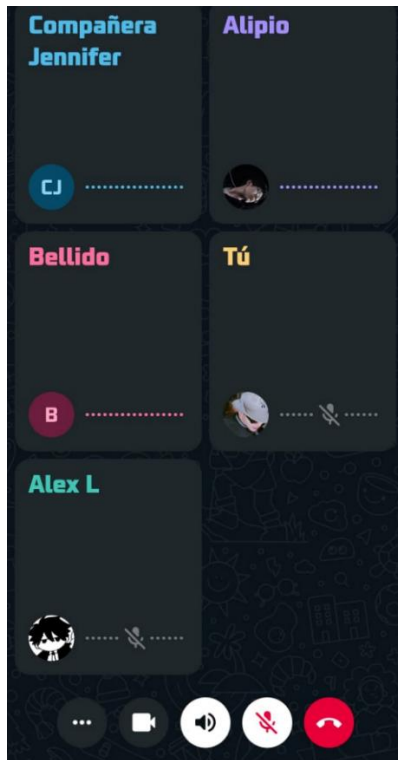
[https://drive.google.com/drive/folders/1yPjTyMkO\\_Jy2iqK87iGc8f6EbDhmj4z2](https://drive.google.com/drive/folders/1yPjTyMkO_Jy2iqK87iGc8f6EbDhmj4z2)

### 9.2. Imágenes

*Figura 13: Reunion Meet*



*Figura 14: Daily Scrum (Whatsapp)*



*Figura 15: Firma del Cliente*



*Figura 16: Reunion con el Cliente*



*Figura 17: Logo del equipo*



*Figura 18: QR de acceso*



*Figura 19: Reunion en biblioteca*



*Figura 20: Reunion en biblioteca*



## X. BIBLIOGRAFIA

Cáriga Mujica, N. N. (25 de Marzo de 2025). *Zenodo*. Obtenido de Análisis de la calidad del servicio en el sector cafetería en el Perú: <https://zenodo.org/records/15086437>

Carneiro, C. (23 de Abril de 2024). *Ola Click* . Obtenido de Códigos QR en Restaurantes: La Innovación Japonesa que Latinoamérica Debería Adoptar para Optimizar Tiempos de Espera: <https://olaclick.com/es/restaurantes-es/codigos-qr-en-restaurantes-la-innovacion-japonesa-que-latinoamerica-deberia-adoptar-para-optimizar-tiempos-de-espera/?referrer=grok.com>

Info Negocios Valencia. (24 de Octubre de 2024). *Info Negocios Valencia*. Obtenido de Las 10 principales tendencias que revolucionarán el sector hostelero en España en 2025: <https://infonegociosvalencia.es/default/las-10-principales-tendencias-que-revolucionaran-el-sector-hostelero-en-espana-en-2025?referrer=grok.com>

IONOS. (11 de Marzo de 2019). *IONOS*. Obtenido de El modelo en cascada: desarrollo secuencial de software: <https://www.ionos.com/es-us/digitalguide/paginas-web/desarrollo-web/el-modelo-en-cascada/>

Ken Schwaber, J. S. (21 de Julio de 2013). *Scrum guides* . Obtenido de La Guía de Scrum - La Guía Definitiva de Scrum: Las Reglas del Juego: <https://scrumguides.org/docs/scrumguide/v1/Scrum-Guide-ES.pdf>

Kuno Digital. (16 de Mayo de 2025). *Kuno Digital* . Obtenido de Transformación digital en el sector de restauración: tendencias clave: <https://kunodigital.com/2024/11/28/transformacion-digital-restauracion-tendencias/?referrer=grok.com>

Martina, J. (15 de Febrero de 2025). *Asana*. Obtenido de Scrum: conceptos clave y cómo se aplica en la gestión de proyectos: <https://asana.com/es/resources/what-is-scrum>



Peru-Retail. (18 de Mayo de 2022). *Peru-Retail*. Obtenido de Transformación digital permite a restaurantes vender hasta un 40% más: <https://www.peru-retail.com/transformacion-digital-permite-a-restaurantes-vender-hasta-un-40-mas/?referrer=grok.com>

Torresburriel Estudio. (22 de Octubre de 2021). *Torresburriel* . Obtenido de La técnica de la pirámide invertida en UX Writing: <https://torresburriel.com/weblog/piramide-invertida-ux-writing/>

Villalón, C. P. (12 de Julio de 2025). *Pascual Profesional* . Obtenido de Tendencias Gastronómicas 2025: Innovación y Adaptación al Cambio: <https://www.pascualprofesional.com/blog/interes/tendencias-gastronomicas-2025-innovacion-y-adaptacion-al-cambio/?referrer=grok.com>