

1. **Fecha:** 11/06/2025

2. **Grupo base:** Grupo #1

3. **Integrantes:**

Apellidos y Nombres	Calificación
Marvin Pilay.	
Johnny Cambisaca.	
González George	

4. **Objetivos específicos** que persigue la realización del trabajo.

4.1. Crear Descripciones de Diseño de Software (DDS).

4.2. Comprender el rol y la importancia de una DDS dentro del proceso de desarrollo.

5. **Tareas específicas** que se deben realizar en este ejercicio.

5.1. Seguir las instrucciones del ejercicio.

6. **Entregables** que deberán ser cargados en el AVAC.

6.1. Archivo en PDF con el siguiente nombre: T02\_02\_GrupoXX\_Apellido1Nombre1.pdf

6.2. Incluso si la Tarea se realiza en grupo, la carga a la plataforma debe ser individual.

7. **Instrucciones**

7.1. Identificar la plantilla del Anexo C del IEEE Standard for Information Technology-Systems Design-Software Design Descriptions.

7.2. Desarrollar una Descripción de Diseño de Software (DDS)

7.2.1. Utilizar la plantilla identificada en el punto 7.1.

7.2.2. Los diagramas deben estar basados a la Especificación de Requerimientos de Software (SRS) realizada en la Tarea 02.01.

7.2.3. Dividir los diagramas según el punto de vista definido en el estándar.

7.3. Los diagramas que deben ser desarrollados son:

7.3.1. Diagrama de Clases.

7.3.2. Diagrama Entidad Relación.

7.3.3. Diagrama de Secuencia.

7.3.4. Diagrama de Estados.

8. **Rúbrica de calificación.**

Criterios	No Presenta	Nivel Bajo	Nivel Medio	Nivel Alto	Total
Los diagramas de clases utilizan correctamente la notación UML.	0	0.5	1	2	
Los diagramas entidad-relación, de secuencia y de estados utilizan correctamente la notación	0	0.2	0.5	1	
Los diagramas se encuentran ordenados de acuerdo con el estándar 1016.	0	0.2	0.5	1	

# Software Design Description (SDD)

**Sistema:** Gestión de Minimercado

**Date of Issue and Status:** 11/06/2025 – Versión 0.1

**Issuing Organization:** Universidad Politécnica Salesiana

Carrera de Ingeniería de Software

## Authorship:

Johnny Cambisaca

Marvin Pilay

González George

## Change History:

Versión	Fecha	Descripción	Autor
0.1	11/06/2025	Documento Software Design Description (SSD)	Johnny Cambisaca Marvin Pilay González George

## Índice del Documento

<b>Portada</b> .....	2
Fecha de emisión y estado.....	2
Organización emisora.....	2
Autoría .....	2
Historial de cambios.....	2
<b>Introducción</b> .....	4
Propósito .....	4
Alcance .....	4
Contexto .....	4
Resumen.....	4
<b>Referencias</b> .....	5
<b>Glosario</b> .....	5
<b>Cuerpo del Documento</b> .....	5
Partes interesadas y preocupaciones de diseño .....	5
Punto de vista de diseño 1.....	6
Punto de vista de diseño 2 .....	7
Punto de vista de diseño 3 .....	7
Punto de vista de diseño 4.....	7
Razonamiento del diseño.....	8

# Introducción:

## Propósito

El propósito de nuestro documento es poder abiertamente describir y detallar el diseño del software del sistema de Gestión de Minimercado en el cual se busca proporcionar una base técnica para la implementación de su diseño asegurando que el sistema cumpla con los requerimientos funcionales y no funcionales establecidos según el documento de Especificación de Requerimientos de Software (SRS). Además que este mismo sirva como guía o referencia para nosotros los desarrolladores, probadores y demás partes que estarán involucradas en el proceso del desarrollo.

## Alcance

Este sistema será diseñado para automatizar y gestionar eficientemente los procesos operativos de un minimercado. El sistema incluye módulos para gestionar la configuración del sistema, sus proveedores, productos e inventario, ventas, caja, clientes y reportes. En este documento se abarcará el diseño de la arquitectura del sistema, componentes principales, interacciones, estructuras de datos, y los diagramas UML necesarios que representan su comportamiento y estructura.

## Contexto

Nuestro diseño se basa en el contexto definido en el documento SRS en el cual se identificó la necesidad de un sistema integrado para el control y supervisión de flujo de productos, transacciones de venta, control de caja y su generación de reportes.

El sistema está destinado a ser usado por los administradores del minimercado y sus empleados, permitiendo un manejo seguro y organizado de la información del negocio.

## Resumen

En ese documento se pone a disposición:

- Vista general de los stakeholders y sus intereses.
- Descripción del diseño del software con respecto a su arquitectura
- Diagramas UML:
  - Diagramas de Clases, Entidad-Relación, Secuencia, Estados
- Justificación de decisiones de diseño.
- Glosario y referencias normativas.

## Referencias:

IEEE 1016-2009 - Standard for Information Technology-Systems Design-Software Design Descriptions

## Glosario:

<b>Término</b>	<b>Definición</b>
SDD	Documento de Diseño – Software Design Description
UML	Lenguaje de modelado Unificado – Unified Modeling Language
ER	Entidad-Relación. Modelo que representa datos
Módulo	Unidad funcional del sistema que agrupa funcionalidades que están relacionadas
Caja	Parte del sistema que se encarga de registrar los pagos

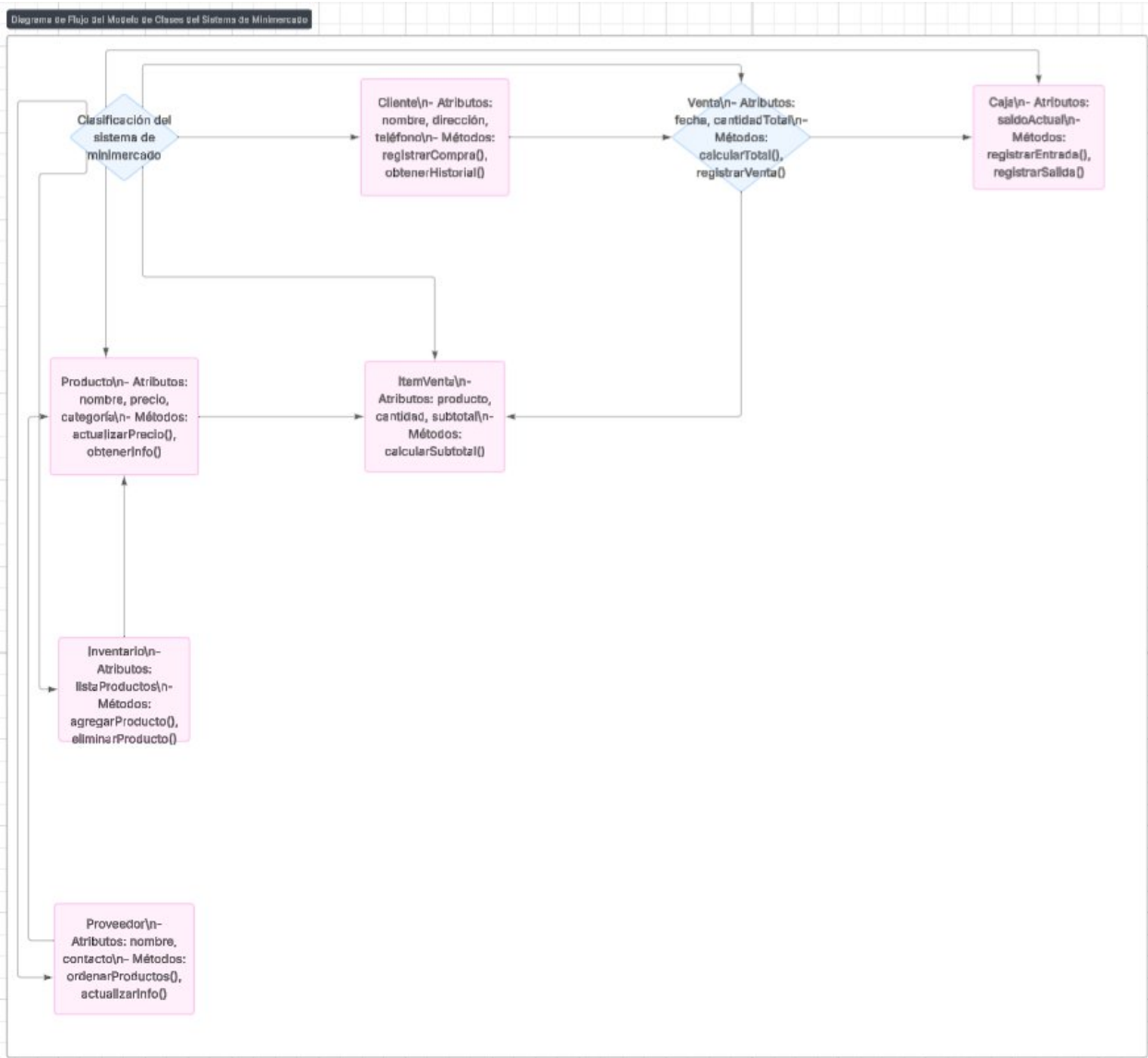
## Cuerpo del Documento:

### Partes interesadas y preocupaciones de diseño

<b>Stakeholder</b>	<b>Rol / Interés</b>	<b>Preocupaciones de Diseño</b>
Administrador	Supervisa la operación del sistema y toma decisiones	Seguridad de los datos, acceso total a módulos, generación de reportes
Cajero	Registra ventas y maneja la caja	Facilidad de uso, velocidad de respuesta, interfaz clara
Encargado de inventario	Registra entradas y salidas de productos	Gestión simple de inventario, alertas por stock bajo
Desarrolladores	Implementan el sistema	Claridad del diseño, modularidad, facilidad de mantenimiento
Equipo de pruebas	Verifica calidad del sistema	Trazabilidad de funcionalidades, cobertura de pruebas
Cliente/usuario final	Utiliza el sistema en un entorno real	Facilidad de uso, confiabilidad, estabilidad del sistema

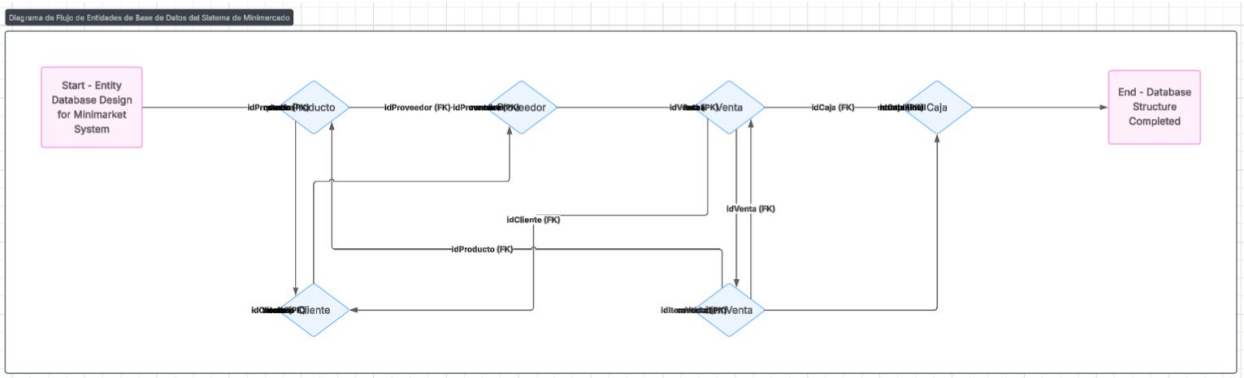
# Punto de vista de diseño 1

## Estructural



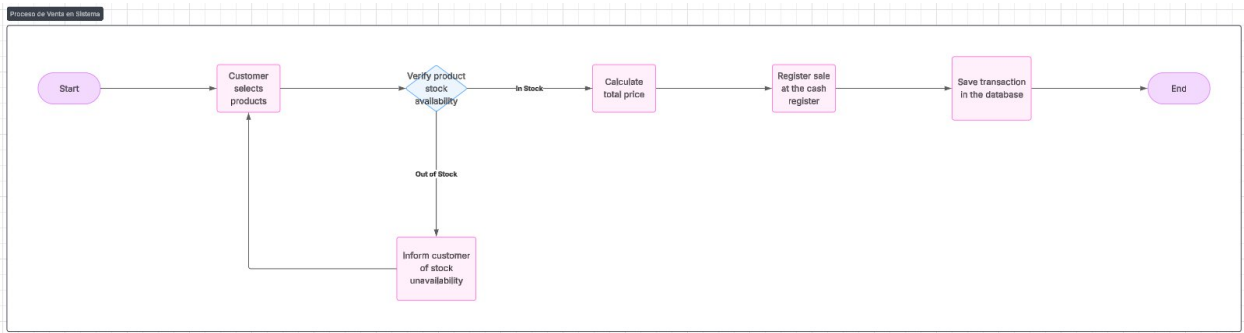
## Punto de vista de diseño 2

### Datos



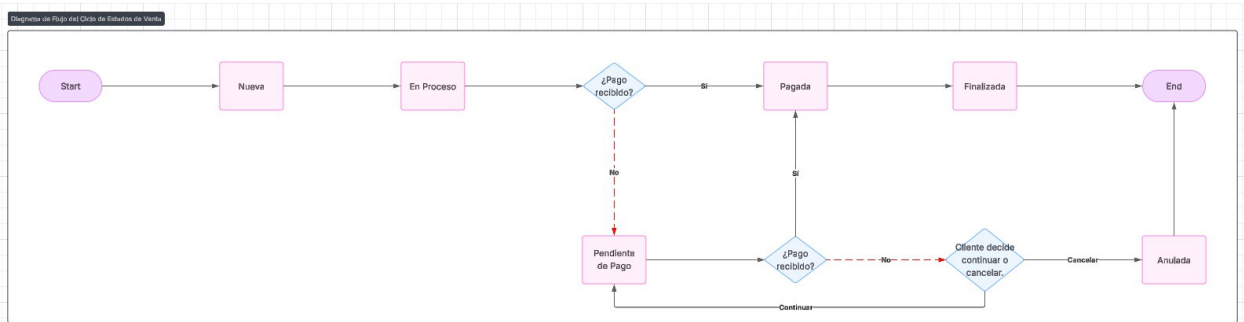
## Punto de vista de diseño 3

### Comportamiento Interactivo



## Punto de vista de diseño 4

### Estados del sistema



## Razonamiento del Diseño

En el diseño propuesto se sigue los principios de modularidad, bajo acoplamiento y alta cohesión y por supuesto la reutilización. Se utiliza la arquitectura, lógica de negocio y acceso a datos lo cual permite escalar el sistema fácilmente y realizar el mantenimiento y extenderlo con nuevas funciones.

Las decisiones que fueron claves son:

- Uso de controladores y servicios para poder aislar la lógica
- Reutilización de clases para venta, productos y reportes.
- Separación clara entre entidades persistentes y componentes de UI.