Logotipo, nombre de la empresa

El contenido generado por IA puede ser incorrecto.

Facultad de Ingeniería

Ingeniería de Software I

**PROYECTO INGENIERÍA DE SOFTWARE SISTEMA DE**

**VENTAS E INVENTARIO PARA BOTILLERIA**

Autores:

Sebastián Mihael Atanasovici Díaz

Matías Nicolas Cáceres Ruz

Almendra Constanza Manríquez Torres

Joaquín Eduardo Soto Aguirre

David Héctor Orlando Zúñiga Jiménez

Profesor:  
Paulo Luis Francisco Quínsacara Jofré

Santiago, Chile

202

Tabla de contenido

[1. INTRODUCCIÓN 8](#_heading=h.4aeyzlio35c6)

[2. PROPUESTA DE SOLUCIÓN E INCREMENTO 9](#_heading=h.ac30asbhcqvk)

[3. CASOS DE USO DEL PRIMER INCREMENTO 10](#_heading=h.ke83sc5hmivk)

[4. PRODUCT BACKLOG DEL PRIMER INCREMENTO 27](#_heading=h.ci5pfnpjyzds)

[5. VISTA 4 + 1 28](#_heading=h.31cgz7f60gws)

[5.1. Vista lógica 28](#_heading=h.vb02augktg0w)

[5.2. Vista proceso 32](#_heading=h.3x57rlh8xcwf)

[5.3. Vista de desarrollo 49](#_heading=h.flz4r01xipap)

[5.4. Vista física 51](#_heading=h.nvlx8jg1ilm6)

[6. ÁRBOL DE NAVEGACIÓN 53](#_heading=h.u7l1zvgzy8br)

[7. CASOS DE PRUEBA 55](#_heading=h.xvcelxtwh1j3)

[7.1. Caso de prueba CU01 55](#_heading=h.uhhtqq22ei5b)

[7.2. Caso de prueba CU02 61](#_heading=h.lq41ihpra3qd)

[7.3. Caso de prueba CU04 64](#_heading=h.gg1q425sj3um)

[7.4. Caso de prueba CU05 67](#_heading=h.gfn8a3xw7kzk)

[7.5. Caso de prueba CU07 69](#_heading=h.mi3vwpupv38w)

[7.6. Caso de prueba CU10 73](#_heading=h.73q2ssyk5gee)

[7.7. Caso de prueba CU11 76](#_heading=h.wy8fxpri8l8n)

[7.8. Caso de prueba CU20 79](#_heading=h.mlamnaibel4h)

[7.9. Caso de prueba CU23 81](#_heading=h.ogjsorn8d17j)

[7.10. Caso de prueba CU24 85](#_heading=h.no041jit0t3e)

[7.11. Caso de prueba CU40 89](#_heading=h.a6ftnqak6ot1)

[7.12. Caso de prueba CU41 93](#_heading=h.i4e83u45qrnm)

[7.13. Caso de prueba CU42 96](#_heading=h.bunevk8qvll2)

[7.14. Caso de prueba CU43 99](#_heading=h.pj2vn34pe3iq)

[7.15. Caso de prueba CU45 102](#_heading=h.qzqf5q1gcdqp)

[8. REUNIONES SCRUM Y GESTIÓN DEL AVANCE DEL PRIMER INCREMENTO 107](#_heading=h.ozh5o1593z72)

[8.1. Reuniones diarias (Daily Scrum) 107](#_heading=h.jp450a4frl2p)

[8.2. Revisión del Sprint (Sprint Review) 107](#_heading=h.n94rn9jbjs3f)

[8.3. Retrospectiva del Sprint (Sprint Retrospective) 108](#_heading=h.aqs612b1ha6c)

[8.4. Finalización del Product Backlog 108](#_heading=h.rwutz69gjyun)

[9. TABLA DE ESFUERZO PRIMER INCREMENTO 110](#_heading=h.o6quekq0qhhk)

[9.1. Burn up y burn down horas hombre 111](#_heading=h.2nbhgfd5jmcl)

[9.2. Burn up y burn down tareas 113](#_heading=h.sxxek1uo21yg)

[10. BIBLIOGRAFIA 116](#_heading=h.xlej03jzvybk)

[11. ANEXOS 117](#_heading=h.t02szfsd5s8c)

[Anexo A: Detalle de vistas del sistema 117](#_heading=h.7rcoc481nydz)

Índice de Figuras

[Figura 4.1: Product backlog del primer incremento 26](#_heading=h.pwn5vfpw7h0y)

[Figura 5.1: Diagrama de Clases 29](#_heading=h.wvnwjcgo5jnh)

[Figura 5.2: Modelo de datos 31](#_heading=h.iqk9b1lai0xk)

[Figura 5.3: “Diagrama de secuencia del Caso de Uso N°1, Registrando una venta (Parte 1)” 33](#_heading=h.v1xg62lxna7d)

[Figura 5.4: “Diagrama de secuencia del Caso de Uso N°1, Registrando una venta (Parte 2)” 34](#_heading=h.4npqufvdy1ll)

[Figura 5.5: “Diagrama de secuencia del Caso de Uso N°2, Calculando el total de la venta”. 35](#_heading=h.v94ua9v1fddz)

[Figura 5.6: “Diagrama de secuencia del Caso de Uso N°4, Modificando cantidad de productos”. 36](#_heading=h.9o0160o9n4zh)

[Figura 5.7: “Diagrama de secuencia del Caso de Uso N°5, Eliminando productos de una venta”. 37](#_heading=h.db13g7i9vmq6)

[Figura 5.8: “Diagrama de secuencia del Caso de Uso N°7, Descontando automáticamente el inventario” 38](#_heading=h.7u3ajbali3w0)

[Figura 5.9: “Diagrama de secuencia del Caso de Uso N°10, Seleccionando método de pago” 39](#_heading=h.h9jossbhvn68)

[Figura 5.10: “Diagrama de secuencia del Caso de Uso N°11, Ingresando monto recibido y calcular cambio” 40](#_heading=h.acc1kqz8tb6h)

[Figura 5.11: “Diagrama de secuencia del Caso de Uso N°20, Visualizando stock disponible” 41](#_heading=h.xzb0f48g2s8n)

[Figura 5.12: “Diagrama de secuencia del Caso de Uso N°23, Registrando nuevos productos” 42](#_heading=h.ze84zfc7bq0w)

[Figura 5.13: “Diagrama de secuencia del Caso de Uso N°24, Registrando ingreso de stock de productos existentes” 43](#_heading=h.y16m4z47gisj)

[Figura 5.14: “Diagrama de secuencia del Caso de Uso N°40, Creando usuarios” 44](#_heading=h.s9c7c2kxy0n4)

[Figura 5.15: “Diagrama de secuencia del Caso de Uso N°41, Modificando usuarios” 45](#_heading=h.fihl927tumrj)

[Figura 5.16: “Diagrama de secuencia del Caso de Uso N°42, Eliminando usuario” 46](#_heading=h.ue1jfikxt8tt)

[Figura 5.17: “Diagrama de secuencia del Caso de Uso N°43, Cambiando rol de usuario” 47](#_heading=h.ue0sw9idmmw2)

[Figura 5.18: "Diagrama de secuencia del Caso de Uso N°45, Iniciando sesión en el sistema” 48](#_heading=h.kg8fc9qzfs03)

[Figura 5.19: Diagrama de componentes 49](#_heading=h.fj67y31xbsud)

[Figura 5.20: Diagrama de Despliegue 51](#_heading=h.gml6jp4iu98s)

[Figura 6.1: Árbol de Navegación 53](#_heading=h.8k0ql01uckr4)

[Figura 7.1: “Evidencia Caso de Prueba 1, Caso de Uso N°1” 56](#_heading=h.xwjocilvfxko)

[Figura 7.2: “Evidencia Caso de Prueba 2, Caso de Uso N°1” 57](#_heading=h.t98r2fg10l3n)

[Figura 7.3: “Evidencia Caso de Prueba 3, Caso de Uso N°1” 57](#_heading=h.yecx94fne5es)

[Figura 7.4: “Evidencia Caso de Prueba 4, Caso de Uso N°1” 58](#_heading=h.1g5giw4f6rtr)

[Figura 7.5: “Evidencia Caso de Prueba 5, Caso de Uso N°1” 59](#_heading=h.azrr0w1yysms)

[Figura 7.6: “Evidencia Caso de Prueba 6, Caso de Uso N°1” 60](#_heading=h.fs25q55oump1)

[Figura 7.7: “Evidencia Caso de Prueba 1, Caso de Uso N°2” 62](#_heading=h.ws14y8mxfi7b)

[Figura 7.8: " Evidencia Caso de Prueba 2, Caso de Uso N°2” 62](#_heading=h.nei5ea9vmzmz)

[Figura 7.9: "Evidencia Caso de Prueba 3, Caso de Uso N°2” 63](#_heading=h.9bshqe8sbttp)

[Figura 7.10: "Evidencia Caso de Prueba 1, Caso de Uso N°4” 65](#_heading=h.sw4cpb9vjh24)

[Figura 7.11: "Evidencia Caso de Prueba 2, Caso de Uso N°4” 66](#_heading=h.fcktk4uxtckm)

[Figura 7.12: "Evidencia Caso de Prueba 1, Caso de Uso N°5” 67](#_heading=h.6d1l06dqdrmi)

[Figura 7.13: "Evidencia Caso de Prueba 2, Caso de Uso N°5” 68](#_heading=h.a09ddlbqu3wj)

[Figura 7.14: "Evidencia Caso de Prueba 1, Caso de Uso N°7” 70](#_heading=h.wa3hoheqakhl)

[Figura 7.15: "Evidencia 1 Caso de Prueba 2, Caso de Uso N°7” 71](#_heading=h.mttv3itjqsaz)

[Figura 7.16: "Evidencia 2 Caso de Prueba 2, Caso de Uso N°7” 71](#_heading=h.bm079jkyxiqf)

[Figura 7.17: "Evidencia 3 Caso de Prueba 2, Caso de Uso N°7” 72](#_heading=h.63bnidkoe0cl)

[Figura 7.18: "Evidencia Caso de Prueba 1, Caso de Uso N°10” 74](#_heading=h.jetsenl3bnyr)

[Figura 7.19: "Evidencia Caso de Prueba 2, Caso de Uso N°10” 75](#_heading=h.c9z55thoepwe)

[Figura 7.20: "Evidencia Caso de Prueba 1, Caso de Uso N°11” 77](#_heading=h.8c8564gr5u5h)

[Figura 7.21: "Evidencia Caso de Prueba 2, Caso de Uso N°11” 77](#_heading=h.bxrgd8gnx1ol)

[Figura 7.22: "Evidencia Caso de Prueba 1, Caso de Uso N°20” 79](#_heading=h.47izhkqdkbtx)

[Figura 7.23: “Evidencia Caso de Prueba 2, Caso de Uso N°20” 80](#_heading=h.n576xzrf9u7o)

[Figura 7.24: "Evidencia Caso de Prueba 1, Caso de Uso N°23” 82](#_heading=h.f54puve1qwp8)

[Figura 7.25: "Evidencia Caso de Prueba 2, Caso de Uso N°23” 82](#_heading=h.qj8l6fzfvqc8)

[Figura 7.26: "Evidencia Caso de Prueba 3, Caso de Uso N°23” 83](#_heading=h.qu8zi7ehlkky)

[Figura 7.27: "Evidencia Caso de Prueba 4, Caso de Uso N°23” 84](#_heading=h.xummte48sew0)

[Figura 7.28: "Evidencia Caso de Prueba 1, Caso de Uso N°24” 85](#_heading=h.73vbpnlv9coz)

[Figura 7.29: "Evidencia Caso de Prueba 2, Caso de Uso N°24” 86](#_heading=h.sb68qsrtrg5c)

[Figura 7.30: "Evidencia Caso de Prueba 3, Caso de Uso N°24” 87](#_heading=h.81vq3cah7r6k)

[Figura 7.31: “Evidencia Caso de Prueba 4, Caso de Uso N°24” 87](#_heading=h.2igz05757aed)

[Figura 7.32: "Evidencia Caso de Prueba 1, Caso de Uso N°40” 89](#_heading=h.pby7gmabxrfh)

[Figura 7.33: "Evidencia Caso de Prueba 2, Caso de Uso N°40” 90](#_heading=h.zc8omxsnokhm)

[Figura 7.34: "Evidencia Caso de Prueba 3, Caso de Uso N°40” 91](#_heading=h.5dhlk759y3zu)

[Figura 7.35: "Evidencia Caso de Prueba 4, Caso de Uso N°40” 91](#_heading=h.9nyx70ozle6l)

[Figura 7.36:"Evidencia Caso de Prueba 5, Caso de Uso N°40” 92](#_heading=h.ot8agtsb70rp)

[Figura 7.37: "Evidencia Caso de Prueba 1, Caso de Uso N°41” 93](#_heading=h.82v1ppd6gfsy)

[Figura 7.38: "Evidencia Caso de Prueba 2, Caso de Uso N°41” 94](#_heading=h.p5ac7zawfyqj)

[Figura 7.39: "Evidencia Caso de Prueba 3, Caso de Uso N°41” 95](#_heading=h.7n7teqqmmcl7)

[Figura 7.40: "Evidencia Caso de Prueba 1, Caso de Uso N°42” 97](#_heading=h.n6hsajq12ue8)

[Figura 7.41: "Evidencia Caso de Prueba 2, Caso de Uso N°42” 97](#_heading=h.3s17tzyvnzd0)

[Figura 7.42: "Evidencia Caso de Prueba 3, Caso de Uso N°42” 98](#_heading=h.awx4ev9w9fwd)

[Figura 7.43: "Evidencia Caso de Prueba 1, Caso de Uso N°43” 100](#_heading=h.qgwngdk876n7)

[Figura 7.44: "Evidencia Caso de Prueba 2, Caso de Uso N°43” 100](#_heading=h.r2dcnzl682co)

[Figura 7.45: "Evidencia Caso de Prueba 3, Caso de Uso N°43” 101](#_heading=h.btlul2d1dax7)

[Figura 7.46: "Evidencia Caso de Prueba 1, Caso de Uso N°45” 103](#_heading=h.5rfxu3gn6pnh)

[Figura 7.47: "Evidencia Caso de Prueba 2, Caso de Uso N°45” 104](#_heading=h.9o7ss4r2q11v)

[Figura 7.48: "Evidencia Caso de Prueba 3, Caso de Uso N°45” 104](#_heading=h.gx1vt6fztox6)

[Figura 7.49: "Evidencia Caso de Prueba 4, Caso de Uso N°45” 105](#_heading=h.h5frbfww9ubc)

[Figura 8.1: Product backlog con tareas finalizadas 108](#_heading=h.eoqrlf153h04)

[Figura 9.1: Tabla de Esfuerzo 110](#_heading=h.imfx3qeapmul)

[Figura 9.2: Burn Up Incremento 1 (HH) 111](#_heading=h.y0rfbld31o8v)

[Figura 9.3: Burn Down Incremento 1 (HH) 112](#_heading=h.scc7gr9q94id)

[Figura 9.4: Progreso del Esfuerzo (Burn Up) 113](#_heading=h.cgy12xlu65k)

[Figura 9.5: Esfuerzo Pendiente (Burn Down) 114](#_heading=h.euunvnlfouiw)

Índice de Tablas

[Tabla 2.1: Distribución de Casos de Uso por Incremento 8](#_heading=h.8nf6r6xbbn3i)

[Tabla 3.1: “Caso de Uso N°1, Registrando una Venta”. 9](#_heading=h.d1tjgc3vvm4f)

[Tabla 3.2: “Caso de Uso N°2, Calculando el Total de la Venta”. 11](#_heading=h.30s59auntqib)

[Tabla 3.3: “Caso de Uso N°4, Modificando Cantidad de Productos”. 12](#_heading=h.oteevav26z8v)

[Tabla 3.4: “Caso de Uso N°5, Eliminando Productos de una Venta”. 13](#_heading=h.bqrlmqs1v2ob)

[Tabla 3.5: “Caso de Uso N°7, Descontando Automáticamente el Inventario” 14](#_heading=h.95i4xt4lrbze)

[Tabla 3.6: “Caso de Uso N°10, Seleccionando Método de Pago” 15](#_heading=h.834jlnsqp60n)

[Tabla 3.7: “Caso de Uso N°11, Ingresando Monto Recibido y Calculando Cambio” 16](#_heading=h.4cyp5fcybrze)

[Tabla 3.8: “Caso de Uso N°20, Visualizando Stock Disponible”. 17](#_heading=h.5gv9tv7ilagg)

[Tabla 3.9: “Caso de Uso N°23, Registrando Nuevos Productos”. 18](#_heading=h.v2bgor6wo4qr)

[Tabla 3.10: “Caso de Uso N°24, Registrando Ingreso de Stock de Productos Existentes”. 19](#_heading=h.bw6ozqbx35m6)

[Tabla 3.11: “Caso de Uso N°40, Creando Usuarios”. 20](#_heading=h.y9z75kuxoscg)

[Tabla 3.12: “Caso de Uso N°41, Modificando Usuarios”. 21](#_heading=h.5ik7yai33luo)

[Tabla 3.13: “Caso de Uso N°42, Eliminando Usuario”. 22](#_heading=h.ol7t3ecykni9)

[Tabla 3.14: “Caso de Uso N°43, Cambiando Rol de Usuario” 23](#_heading=h.t9yguu90r3g)

[Tabla 3.15: “Caso de Uso N°45, Iniciando Sesión en el Sistema”. 24](#_heading=h.thou80t7itp2)

# INTRODUCCIÓN

Este documento presenta el trabajo realizado durante el primer incremento del desarrollo del sistema para la Botillería Joaquín. En esta etapa se implementaron las funcionalidades más importantes para que el sistema comenzara a operar, como el registro de productos, el control de inventario y la gestión de ventas. Estas funciones constituyen la base del software y permiten realizar tareas esenciales del negocio desde las primeras fases del proyecto.

Para organizar el desarrollo, se utilizó la metodología Scrum++, la cual permite dividir el proyecto en etapas más pequeñas denominadas incrementos. Esta metodología facilita el avance progresivo mediante revisiones continuas y la incorporación de mejoras conforme se detectan oportunidades. En este primer incremento se alcanzó un progreso significativo en las funciones básicas, con el objetivo de contar con una versión inicial funcional del sistema.

El informe está estructurado siguiendo el modelo de arquitectura de software 4+1 vistas, lo que implica la incorporación de diversas perspectivas del sistema: la lógica interna (vista lógica), la forma en que se despliega en computadores y servidores (vista de desarrollo o física), la manera en que interactúan los usuarios con el sistema (vista de casos de uso), entre otras. Esta estructura permite una descripción más completa y comprensible del sistema, incluso para personas sin conocimientos técnicos especializados.

A lo largo del documento se detallan los desarrollos realizados, las pruebas efectuadas, los resultados obtenidos, y se incluyen imágenes reales del sistema en funcionamiento. De este modo, cualquier lector podrá comprender los avances logrados durante esta primera etapa del proyecto.

# PROPUESTA DE SOLUCIÓN E INCREMENTO

La solución propuesta consiste en el desarrollo de un sistema web personalizado que permita gestionar digitalmente las ventas y el inventario de la Botillería Joaquín, reemplazando los métodos manuales utilizados hasta ahora. Esta herramienta tiene como objetivo simplificar tareas como el ingreso de productos, el cálculo de montos, la visualización de stock y la generación de ventas.

El desarrollo fue dividido en cuatro incrementos, distribuidos en dos semestres, priorizando las funcionalidades más urgentes en las primeras etapas. Este documento detalla el Primer Incremento, en el cual se implementaron las funciones más críticas para que el sistema pudiera empezar a operar: registro de ventas, manejo de productos, actualización automática del stock, cálculo del total y visualización del inventario disponible.

Durante esta primera fase se desarrollaron 15 casos de uso, lo que representa un 33,3% del total planificado. La *Tabla 2.1* muestra cómo se distribuyen los casos de uso por incremento y su respectivo avance porcentual dentro del proyecto.

*Tabla 2.1: Distribución de Casos de Uso por Incremento*

|  |  |  |  |
| --- | --- | --- | --- |
| **Incremento** | **Porcentaje** | **Porcentaje Acumulado %** | **Cantidad de Casos de Uso** |
|  |
| 1 | 33.3% | 33.3% | 15 |  |
|  |
| 2 | 46.7% | 80% | 21 |  |
|  |
| 3 | 15% | 95% | 7 |  |
|  |
| 4 | 5% | 100% | 2 |  |
|  |

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

# CASOS DE USO DEL PRIMER INCREMENTO

En esta sección se presentan los 15 casos de uso desarrollados durante el Primer Incremento del proyecto. Estos casos fueron seleccionados por su relevancia dentro del funcionamiento básico de la Botillería Joaquín, ya que permiten registrar ventas, gestionar productos y llevar control del inventario, que son tareas fundamentales para el día a día del negocio.

Cada caso de uso corresponde a una acción que puede realizar el usuario dentro del sistema, como ingresar una venta, agregar productos, actualizar stock, entre otros. La selección de estos casos se basó en su prioridad y en el impacto directo que tienen en las operaciones principales de la botillería.

Con la implementación de estos 15 casos de uso, el sistema ya se puede utilizar de forma funcional, permitiendo evaluar si las soluciones desarrolladas responden de manera adecuada a las necesidades reales del negocio.

A continuación, se muestran las tablas correspondientes a cada caso de uso del Primer Incremento.

En la *Tabla 3.1* se presenta el Caso de Uso N°1: “Registrando una Venta”, que permite al vendedor y administrador registrar en el sistema una venta realizada, incluyendo los productos vendidos, las cantidades, el total, el método de pago y el cálculo del cambio.

*Tabla 3.1: “Caso de Uso N°1, Registrando una Venta”.*

|  |  |  |
| --- | --- | --- |
| **Caso de Uso N°1** | Registrando una Venta | |
| **Actores** | Vendedor, Administrador. | |
| **Propósito** | Registrar una venta, almacenando productos vendidos, cantidades, monto total y método de pago utilizado. | |
| **Precondiciones** | El actor debe haber iniciado sesión en el sistema como vendedor o administrador.  Debe existir stock disponible de los productos seleccionados. | |
| **Resumen** | El actor crea una venta seleccionando productos, indicando cantidades, eligiendo un método de pago e ingresando (cuando corresponde) el monto recibido. | |
| **Postcondiciones** | La venta queda registrada exitosamente en la base de datos del sistema.  El stock de cada producto se descuenta automáticamente.  El actor puede visualizar los detalles de la venta recién ingresada. | |
| **Tipos** | Principal. | |
| **Curso Normal de Eventos** | | |
| **Acción de los Actores** | | **Respuesta Sistema** |
| 1. El actor selecciona la opción "Crear Venta" desde el menú lateral.  3. El actor busca y agrega productos desde el listado de productos disponibles.  5. El actor selecciona el método de pago (efectivo o tarjeta).  7. El actor ingresa el monto recibido (si aplica) y confirma la venta.  9. El actor puede ir al módulo Listado de Ventas para verificar. | | 2. El sistema despliega el formulario de registro de venta, mostrando dos secciones: Productos en la venta y Agregar productos a la venta.  4. El sistema agrega los productos seleccionados a la sección de la venta y calcula automáticamente el subtotal.  6. El sistema habilita el campo de monto recibido en caso de pago en efectivo.  8. El sistema valida la información, descuenta el stock correspondiente y registra fecha, hora, productos, cantidades, monto total y método de pago.  10. El sistema muestra la nueva venta junto a las anteriores, con opción de visualizar detalles. |

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

En la *Tabla 3.2* se presenta el Caso de Uso N°2: “Calculando el Total de la Venta”, el cual permite al sistema sumar automáticamente los precios y cantidades de los productos seleccionados por el actor vendedor y administrador.

*Tabla 3.2: “Caso de Uso N°2, Calculando el Total de la Venta”.*

|  |  |  |
| --- | --- | --- |
| **Caso de Uso N°2** | Calculando el Total de la Venta | |
| **Actores** | Vendedor, Administrador. | |
| **Propósito** | Calcular automáticamente el valor total a cobrar en una venta, sumando precios y cantidades de productos seleccionados. | |
| **Precondiciones** | El actor debe haber iniciado sesión en el sistema como vendedor o administrador.  El actor debe haber iniciado una nueva venta.  El actor debe haber agregado productos al carrito de venta. | |
| **Resumen** | El actor agrega o modifica productos en la venta y el sistema actualiza automáticamente el subtotal y el total mostrado en pantalla. | |
| **Postcondiciones** | El total de la venta queda calculado y visible para el actor. | |
| **Tipos** | Principal | |
| **Curso Normal de Eventos** | | |
| **Acción de los Actores** | | **Respuesta Sistema** |
| 1. El actor agrega uno o más productos desde el listado de productos disponibles.  3. El actor modifica la cantidad de uno o más productos si es necesario.  5. El actor revisa el monto total antes de confirmar la venta. | | 2. El sistema suma los precios de los productos seleccionados y calcula automáticamente el subtotal.  4. El sistema recalcula automáticamente el subtotal reflejando los cambios en cantidad.  6. El sistema muestra el valor final actualizado. |

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

En la *Tabla 3.3* se presenta el Caso de Uso N°4: “Modificando Cantidad de Productos”, el cual permite al actor vendedor y administrador ajustar manualmente las unidades de productos dentro del carrito de venta antes de confirmar la transacción.

*Tabla 3.3: “Caso de Uso N°4, Modificando Cantidad de Productos”.*

|  |  |  |
| --- | --- | --- |
| **Caso de Uso N°4** | Modificando Cantidad de Productos | |
| **Actores** | Vendedor, Administrador. | |
| **Propósito** | Modificar la cantidad de productos agregados en la sección Productos en la venta antes de finalizar la venta. | |
| **Precondiciones** | El actor debe haber iniciado sesión en el sistema como vendedor o administrador.  El actor debe tener productos ya agregados al carrito de venta. | |
| **Resumen** | El actor ajusta las cantidades de productos ya agregados en la venta utilizando los botones + / –, y el sistema recalcula automáticamente el subtotal y el total de la venta. | |
| **Postcondiciones** | Las nuevas cantidades quedan registradas en la venta en curso.  El total actualizado se muestra en pantalla. | |
| **Tipos** | |  | | --- | |  |   Principal | |
| **Curso Normal de Eventos** | | |
| **Acción de los Actores** | | **Respuesta Sistema** |
| 1. El actor accede a la sección Productos en la venta.  3. El actor presiona los botones + o - para modificar la cantidad de uno o más productos.  5. El actor revisa los cambios antes de confirmar la venta. | | 2. El sistema muestra los productos agregados con sus cantidades actuales.  4. El sistema recalcula automáticamente el subtotal y muestra el nuevo total.  6. El sistema mantiene actualizadas las cantidades y el total de la venta. |

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

En la *Tabla 3.4* se presenta el Caso de Uso N°5: “Eliminando Productos de una Venta”, el cual permite al actor vendedor y administrador, quitar productos no deseados del carrito de venta antes de finalizar la transacción.

*Tabla 3.4: “Caso de Uso N°5, Eliminando Productos de una Venta”.*

|  |  |  |
| --- | --- | --- |
| **Caso de Uso N°5** | Eliminando Productos de una Venta | |
| **Actores** | Vendedor, Administrador. | |
| **Propósito** | Eliminar un producto específico del carrito antes de finalizar la venta. | |
| **Precondiciones** | El actor debe haber iniciado sesión en el sistema como vendedor o administrador.  El actor debe tener productos en el carrito de venta. | |
| **Resumen** | El actor elimina productos no deseados del carrito de venta antes de confirmar la transacción. | |
| **Postcondiciones** | El carrito queda actualizado sin el producto eliminado y el total de venta recalculado. | |
| **Tipos** | Principal | |
| **Curso Normal de Eventos** | | |
| **Acción de los Actores** | | **Respuesta Sistema** |
| 1. El actor accede a la sección Productos en la venta.  3. El actor hace clic en Eliminar del producto que desea quitar.  5. El actor visualiza el resumen de la venta actualizado. | | 2. El sistema muestra la lista de productos agregados con cantidad, precio y total parcial.  4. El sistema elimina el producto seleccionado de la venta en curso.  6. El sistema recalcula automáticamente subtotal y total, y muestra la venta sin el producto eliminado. |

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

En la *Tabla 3.5* se presenta el Caso de Uso N°7: “Descontando Automáticamente el Inventario”, el cual permite que, al finalizar una venta, el sistema descuente automáticamente del inventario las cantidades correspondientes a los productos vendidos.

*Tabla 3.5: “Caso de Uso N°7, Descontando Automáticamente el Inventario”*

|  |  |  |
| --- | --- | --- |
| **Caso de Uso N°7** | Descontando Automáticamente el Inventario | |
| **Actores** | Vendedor, Administrador. | |
| **Propósito** | Reflejar automáticamente en el inventario la disminución de stock al momento de registrar una venta. | |
| **Precondiciones** | El actor debe haber iniciado sesión en el sistema como vendedor o administrador.  Debe existir al menos un producto en la venta registrada.  El actor debe haber confirmado y completado una venta. | |
| **Resumen** | Cuando se confirma una venta, el sistema descuenta automáticamente del inventario las cantidades de productos vendidos, manteniendo el stock actualizado en tiempo real. | |
| **Postcondiciones** | Las cantidades de los productos vendidos se descuentan del inventario.  El nuevo stock disponible queda registrado y visible en el sistema. | |
| **Tipos** | Principal | |
| **Curso Normal de Eventos** | | |
| **Acción de los Actores** | | **Respuesta Sistema** |
| 1. El actor confirma y finaliza el registro de una venta. | | 2. El sistema identifica los productos vendidos y sus respectivas cantidades.  3. El sistema descuenta automáticamente las cantidades correspondientes de cada producto.  4. El sistema actualiza el stock en tiempo real y lo refleja en la vista de inventario |

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

En la *Tabla 3.6* se presenta el Caso de Uso N°10: “Seleccionando Método de Pago”, el cual permite al actor vendedor y administrador, elegir entre tarjeta o efectivo como medio de pago durante una venta.

*Tabla 3.6: “Caso de Uso N°10, Seleccionando Método de Pago”*

|  |  |  |
| --- | --- | --- |
| **Caso de Uso N°10** | Seleccionando Método de Pago | |
| **Actores** | Vendedor, Administrador. | |
| **Propósito** | Permitir que el actor seleccione el método de pago (Tarjeta o Efectivo) durante el registro de una venta. | |
| **Precondiciones** | El actor debe haber iniciado sesión en el sistema como vendedor.  El actor debe haber finalizado la selección de productos. | |
| **Resumen** | El actor elige el método de pago para completar la transacción, y si corresponde, ingresa el monto recibido para calcular el cambio. | |
| **Postcondiciones** | El método de pago queda registrado junto con la venta en la base de datos. | |
| **Tipos** | Principal | |
| **Curso Normal de Eventos** | | |
| **Acción de los Actores** | | **Respuesta Sistema** |
| 1. El actor selecciona una opción en Método de Pago.  3. El actor elige una de las opciones.  5. El actor ingresa el monto recibido si es que el pago fue en efectivo. | | 2. El sistema muestra las opciones de "Tarjeta" o "Efectivo".  4. El sistema registra el método de pago seleccionado.  6. El sistema calcula automáticamente el cambio y lo muestra en pantalla. |

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

En la *Tabla 3.7* se detalla el Caso de Uso N°11: “Ingresando Monto Recibido y Calculando Cambio”, el cual permite al actor vendedor y administrador ingresar el monto en efectivo entregado por el cliente para que el sistema calcule automáticamente el cambio a entregar.

*Tabla 3.7: “Caso de Uso N°11, Ingresando Monto Recibido y Calculando Cambio”*

|  |  |  |
| --- | --- | --- |
| **Caso de Uso N°11** | Ingresando Monto Recibido y Calculando Cambio | |
| **Actores** | Vendedor, Administrador. | |
| **Propósito** | Permitir al actor ingresar el monto entregado por el cliente en efectivo y obtener automáticamente el valor del cambio que debe entregarse. | |
| **Precondiciones** | El actor debe haber iniciado sesión en el sistema como vendedor o administrador.  El actor debe haber registrado todos los productos en la venta y tener visible el total a pagar.  La venta debe ser pago en efectivo. | |
| **Resumen** | Al momento de recibir un pago en efectivo, el actor puede ingresar el monto recibido y el sistema calcula de forma automática el cambio, mostrándolo en pantalla. | |
| **Postcondiciones** | El monto recibido queda registrado en la venta y el sistema informa el cambio correspondiente.  Se puede continuar con el cierre de la transacción. | |
| **Tipos** | Principal | |
| **Curso Normal de Eventos** | | |
| **Acción de los Actores** | | **Respuesta Sistema** |
| 1. El actor visualiza el total a pagar de la venta.  3. El actor ingresa el monto entregado por el cliente. | | 2. El sistema muestra un campo para ingresar el monto recibido en efectivo.  4. El sistema valida que el monto sea mayor o igual al total a pagar.  5. El sistema muestra en pantalla el valor exacto del cambio. |

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

En la *Tabla 3.8* se presenta el Caso de Uso N°20: “Visualizando stock disponible”, el cual permite al actor vendedor y administrador consultar el inventario actualizado de los productos registrados en el sistema.

*Tabla 3.8: “Caso de Uso N°20, Visualizando Stock Disponible”.*

|  |  |  |
| --- | --- | --- |
| **Caso de Uso N°20** | Visualizando Stock Disponible | |
| **Actores** | Vendedor, Administrador. | |
| **Propósito** | Permitir consultar el stock disponible y actualizado de cada producto en el sistema. | |
| **Precondiciones** | El actor debe haber iniciado sesión en el sistema como administrador o vendedor.  Deben existir productos registrados en la base de datos. | |
| **Resumen** | El actor accede a la opción Inventario, donde el sistema muestra la lista de productos con su información completa: código, nombre, tamaño, marca, cantidad disponible, precios, proveedor, fecha de ingreso y stock mínimo configurado. | |
| **Postcondiciones** | El actor visualiza correctamente el stock actualizado de todos los productos. | |
| **Tipos** | Principal | |
| **Curso Normal de Eventos** | | |
| **Acción de los Actores** | | **Respuesta Sistema** |
| 1. El actor selecciona la opción “Inventario” en el menú lateral. | | 2. El sistema despliega el listado de productos con su stock disponible, precios, proveedor, fecha de ingreso y stock mínimo. |

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

En la *Tabla 3.9* se presenta el Caso de Uso N°23: “Registrando Nuevos Productos”, el cual permite al administrador ingresar un nuevo producto al sistema, completando su información básica como nombre, marca, tamaño, precios y proveedor.

*Tabla 3.9: “Caso de Uso N°23, Registrando Nuevos Productos”.*

|  |  |  |
| --- | --- | --- |
| **Caso de Uso N°23** | Registrando Nuevos Productos | |
| **Actores** | Administrador | |
| **Propósito** | Permitir registrar un nuevo producto que aún no existe en el inventario, ingresando su información básica: nombre, tamaño, marca, precios de compra y venta, proveedor, código de barras, fecha de vencimiento. | |
| **Precondiciones** | El actor debe estar autenticado en el sistema como administrador. | |
| **Resumen** | Desde la sección Crear Productos, el actor selecciona la opción Agregar producto, completa el formulario con los datos requeridos y guarda. El sistema valida la información y registra el nuevo producto en la base de datos. | |
| **Postcondiciones** | El producto queda registrado en el inventario y aparece en la tabla de productos. | |
| **Tipos** | Principal | |
| **Curso Normal de Eventos** | | |
| **Acción de los Actores** | | **Respuesta Sistema** |
| 1. El actor ingresa a la sección Crear Productos.  3. El actor hace clic en Agregar producto.  5. El actor completa los campos y confirma. | | 2. El sistema muestra la tabla con los productos registrados y el botón Agregar producto.  4. El sistema despliega un formulario con campos: nombre, tamaño, marca, precios, proveedor, código de barras, fecha de vencimiento.  6. El sistema registra el nuevo producto en la base de datos, lo muestra en la tabla y confirma que fue agregado correctamente. |

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

En la *Tabla 3.10* se presenta el Caso de Uso N°24: “Registrando Nuevos Productos”, el cual permite al administrador ingresar un nuevo producto al sistema, completando su información básica como nombre, marca, tamaño, precios y proveedor.

*Tabla 3.10: “Caso de Uso N°24, Registrando Ingreso de Stock de Productos Existentes”.*

|  |  |  |
| --- | --- | --- |
| **Caso de Uso N°24** | Registrando Ingreso de Productos Existentes | |
| **Actores** | Administrador | |
| **Propósito** | Permitir al actor registrar el ingreso de unidades de productos ya existentes en el inventario, indicando proveedor, fecha, cantidad recibida y fecha de vencimiento. | |
| **Precondiciones** | El actor debe estar autenticado en el sistema como administrador.  Debe existir el producto previamente registrado en el inventario.  El actor debe haber recibido productos de un proveedor. | |
| **Resumen** | Desde la sección Añadir Stock, el actor selecciona un producto de la lista y hace clic en Añadir stock. Luego completa los datos requeridos (cantidad, fecha de recepción, fecha de vencimiento y proveedor) y confirma. | |
| **Postcondiciones** | El inventario se actualiza con las nuevas unidades ingresadas. | |
| **Tipos** | Principal | |
| **Curso Normal de Eventos** | | |
| **Acción de los Actores** | | **Respuesta Sistema** |
| 1. El actor accede a la sección Añadir Stock desde el menú lateral.  3. El actor selecciona el producto al que desea añadir unidades.  5. El actorcompleta los datos y confirma. | | 2. El sistema muestra la tabla de productos existentes con el botón Añadir stock en cada fila.  4. El sistema despliega un formulario para ingresar la cantidad, fecha de recepción, fecha de vencimiento y proveedor.  6. El sistema valida la información y suma la cantidad al stock del producto. |

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

En la *Tabla 3.11* se presenta el Caso de Uso N°40: “Creando Usuarios”, el cual permite al administrador registrar nuevos usuarios en el sistema, definiendo su nombre, credenciales y rol correspondiente.

*Tabla 3.11: “Caso de Uso N°40, Creando Usuarios”.*

|  |  |  |
| --- | --- | --- |
| **Caso de Uso N°40** | Creando Usuarios | |
| **Actores** | Administrador | |
| **Propósito** | Registrar un nuevo usuario en el sistema. | |
| **Precondiciones** | El actor debe estar autenticado en el sistema como administrador. | |
| **Resumen** | El administrador ingresa los datos del nuevo usuario (nombre, nombre usuario, contraseña, rol) y guarda el registro. | |
| **Postcondiciones** | El nuevo usuario queda registrado y puede iniciar sesión según su rol. | |
| **Tipos** | Extendido. | |
| **Curso Normal de Eventos** | | |
| **Acción de los Actores** | | **Respuesta Sistema** |
| 1. El actor selecciona "Crear nuevo usuario".  3. El actor completa los datos del nuevo usuario (nombre, nombre usuario, contraseña y rol) | | 2. El sistema muestra un formulario para ingresar los datos.  4. El sistema valida la información.  5. El sistema guarda el nuevo usuario y muestra un mensaje de éxito. |

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

En la *Tabla 3.12* se presenta el Caso de Uso N°41: “Modificando Usuarios”, que permite al administrador editar la información de un usuario previamente registrado, incluyendo su nombre, contraseña o rol.

*Tabla 3.12: “Caso de Uso N°41, Modificando Usuarios”.*

|  |  |  |
| --- | --- | --- |
| **Caso de Uso N°41** | Modificando Usuarios | |
| **Actores** | Administrador | |
| **Propósito** | Editar la información de un usuario existente. | |
| **Precondiciones** | El actor debe estar autenticado en el sistema como administrador.  El usuario por modificar debe estar registrado. | |
| **Resumen** | El administrador selecciona un usuario y puede editar su nombre, nombre de usuario o rol. | |
| **Postcondiciones** | La información del usuario queda actualizada. | |
| **Tipos** | Principal | |
| **Curso Normal de Eventos** | | |
| **Acción de los Actores** | | **Respuesta Sistema** |
| 1. El actor accede al módulo "Gestión de Usuarios".  3. El actor selecciona "Editar" sobre un usuario.  5. El actor modifica la información deseada. | | 2. El sistema muestra los usuarios registrados.  4. El sistema muestra un formulario con los datos actuales.  6. El sistema guarda los cambios y confirma la actualización. |

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

En la *Tabla 3.13* se presenta el Caso de Uso N°42: “Eliminando Usuario”, el cual permite al administrador eliminar a un usuario registrado del sistema.

*Tabla 3.13: “Caso de Uso N°42, Eliminando Usuario”.*

|  |  |  |
| --- | --- | --- |
| **Caso de Uso N°42** | Eliminando Usuario | |
| **Actores** | Administrador | |
| **Propósito** | Eliminar un usuario del sistema. | |
| **Precondiciones** | El actor debe estar autenticado en el sistema como administrador.  El usuario por eliminar debe estar registrado. | |
| **Resumen** | El administrador selecciona un usuario y lo elimina del sistema, previa confirmación. | |
| **Postcondiciones** | El usuario es eliminado y ya no puede acceder al sistema. | |
| **Tipos** | Principal. | |
| **Curso Normal de Eventos** | | |
| **Acción de los Actores** | | **Respuesta Sistema** |
| 1. El actor accede al módulo "Gestión de Usuarios".  3. El actor selecciona "Eliminar" sobre un usuario.  5. El actor confirma la eliminación. | | 2. El sistema muestra los usuarios registrados.  4. El sistema solicita confirmación.  6. El sistema elimina al usuario y muestra un mensaje de confirmación. |

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

En la *Tabla 3.14* se presenta el Caso de Uso N°43: “Cambiando Rol de Usuario”, que permite al administrador modificar el rol asignado a un usuario registrado. Este caso es un extendido del caso “Modificando usuario”, ya que forma parte del proceso de edición, específicamente enfocado en actualizar el perfil del usuario para que acceda a nuevas funciones según su nuevo rol.

*Tabla 3.14: “Caso de Uso N°43, Cambiando Rol de Usuario”*

|  |  |  |
| --- | --- | --- |
| **Caso de Uso N°43** | Cambiando Rol de Usuario | |
| **Actores** | Administrador | |
| **Propósito** | Permitir al administrador cambiar el rol de un usuario, de modo que este vea y acceda a funciones distintas dentro del sistema según su nuevo perfil. | |
| **Precondiciones** | El actor debe estar autenticado en el sistema como administrador.  El usuario por eliminar debe estar registrado. | |
| **Resumen** | El administrador accede a la edición de un usuario, modifica su rol (por ejemplo, de Vendedor a Administrador) y el sistema actualiza automáticamente los permisos y accesos del usuario según el nuevo rol. | |
| **Postcondiciones** | El usuario tiene habilitados los accesos correspondientes a su nuevo rol. En su próxima sesión verá los módulos y funciones asociadas a dicho rol. | |
| **Tipos** | Extendido. | |
| **Curso Normal de Eventos** | | |
| **Acción de los Actores** | | **Respuesta Sistema** |
| 1. El actor accede al módulo "Gestión de Usuarios".  3. El actor selecciona "Editar" sobre un usuario.  5. El administrador cambia el rol del usuario. | | 2. El sistema muestra los usuarios registrados.  4. El sistema carga los datos del usuario, incluyendo su rol actual.   |  | | --- | |  |   6. El sistema guarda el nuevo rol y actualiza automáticamente los permisos asociados.  7. El sistema guarda los cambios y confirma la actualización. |

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

En la *Tabla 3.15* se presenta el Caso de Uso N°45: “Iniciando Sesión en el Sistema”, el cual permite al usuario administrador y vendedor autenticarse mediante su nombre de usuario y contraseña.

*Tabla 3.15: “Caso de Uso N°45, Iniciando Sesión en el Sistema”.*

|  |  |  |
| --- | --- | --- |
| **Caso de Uso N°45** | Iniciando Sesión en el Sistema | |
| **Actores** | Administrador, Vendedor | |
| **Propósito** | Permitir que el usuario acceda al sistema utilizando sus credenciales. | |
| **Precondiciones** | El usuario debe estar registrado previamente en el sistema. | |
| **Resumen** | El usuario ingresa su nombre de usuario y contraseña.  El sistema valida las credenciales y, si son correctas, permite el acceso al panel correspondiente según su rol. | |
| **Postcondiciones** | El usuario queda autenticado y con acceso al sistema según su rol. | |
| **Tipos** | Principal. | |
| **Curso Normal de Eventos** | | |
| **Acción de los Actores** | | **Respuesta Sistema** |
| 1. El actor accede a la página de inicio de sesión.  3. El actor ingresa su nombre de usuario y contraseña.  5. El actor hace clic en "Iniciar sesión". | | 2. El sistema muestra el formulario de login.  4. El sistema valida las credenciales.  6. Si las credenciales son correctas, el sistema redirige al panel del usuario según su rol. Si son incorrectas, muestra un mensaje de error. |

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

# PRODUCT BACKLOG DEL PRIMER INCREMENTO

En esta sección se presenta el Product Backlog correspondiente al Primer Incremento del sistema. La *Figura 4.1* muestra la tabla que contiene todas las tareas planificadas por el equipo para esta etapa del proyecto. Cada fila representa una tarea específica, organizada según su nivel de prioridad. Además, se indica el caso de uso asociado (cuando corresponde), el nombre del responsable, la estimación del esfuerzo en horas, el estado de avance de la tarea, el objetivo del sprint relacionado y las fechas estimadas de inicio y término.

Esta planificación permitirá visualizar con claridad qué tareas se priorizarán, cómo se distribuirá el trabajo entre los integrantes del equipo y qué funcionalidades se decidirán para implementar en este primer incremento.

*Figura 4.1: Product backlog del primer incremento*

Una imagen de un periódico

El contenido generado por IA puede ser incorrecto.Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

Para una visualización más clara y detallada, se recomienda revisar directamente el archivo “Ingeniería de Software – Primer Incremento.xlsx”, específicamente en la pestaña Planificación.

# VISTA 4 + 1

En esta sección se documenta la arquitectura del sistema desarrollado utilizando el modelo 4+1 propuesto por Kruchten. Este enfoque organiza la arquitectura en cinco vistas complementarias, que permiten representar de manera clara tanto la estructura del sistema como su comportamiento. Además, está pensado para responder a las necesidades de distintos tipos de usuarios, desde programadores hasta usuarios finales o evaluadores técnicos. Gracias a esto, es posible entender mejor cómo está construido el sistema, cómo se comporta y cómo mantenerlo en el tiempo (Kruchten, 1995).

## Vista lógica

La vista lógica, según Kruchten (1995), está enfocada en los elementos del sistema que son visibles para el usuario final, y es la que describe la funcionalidad principal mediante estructuras como clases, objetos o paquetes, y busca representar las abstracciones más relevantes del sistema desde una perspectiva funcional. En otras palabras, muestra los objetos que el sistema expone al usuario final, permitiendo entender cómo se modelan las funciones clave del sistema.

En la *Figura 5.1* se muestra el diagrama de clases del sistema desarrollado para la Botillería Joaquín, donde se representan las principales entidades del sistema y sus relaciones. Un diagrama de clases permite visualizar de forma clara la estructura de un sistema, facilitando la comprensión de sus componentes, cómo se relacionan entre sí y qué funciones realiza cada uno (Lucidchart, s.f.)

Una de las clases principales representadas es Usuario, encargada de almacenar los datos de las personas que utilizan el sistema. Desde esta clase se derivan dos tipos: Administrador y Vendedor. Ambos perfiles pueden acceder al sistema, sin embargo, el Administrador tiene privilegios adicionales, como la creación de productos o la edición de usuarios, mientras que el Vendedor puede registrar ventas y consultar el inventario.

Otra clase importante es Venta, que guarda la información cada vez que se realiza una compra en la botillería. Allí se registra la fecha, la hora, cuánto se pagó, cómo se pagó y quién hizo la venta. Cada venta se divide en partes más pequeñas que están en la clase DetalleVenta, donde se indica qué productos se vendieron, cuántas unidades y cuánto costaron.

La clase Producto contiene la información relevante de cada producto, como nombre, precio, formato, código de barras y fecha de vencimiento. Además, cada DetalleVenta se vincula a un Producto específico, lo que permite saber exactamente qué se vendió en cada línea de la venta. Esta relación es de muchos a uno, es decir, un producto puede estar en muchas ventas distintas, pero cada detalle de venta corresponde a un solo producto. Así, el sistema puede identificar cuántas unidades se han vendido de cada producto y generar reportes o análisis más detallados.

Los productos también se van recibiendo en la botillería para mantener el stock. Para eso se usa la clase Stock, que registra cada vez que llega una nueva caja de productos, cuántas unidades son y cuándo vencen. Así, el sistema sabe cuánto stock hay disponible.

En cuanto a los pagos, cada venta debe pagarse con un único método: efectivo o tarjeta. Eso se representa en el sistema con la clase Método de Pago, que se divide en dos opciones: una para efectivo (donde se puede calcular el vuelto) y otra para tarjeta (que solo registra el monto).

Las líneas entre las clases muestran cómo se conectan entre sí y cuántas veces ocurre esa conexión. Estas conexiones son:

* Un usuario puede realizar muchas ventas, pero cada venta pertenece solo a un usuario.
* Una venta puede incluir varios productos (esto se refleja en varias líneas de DetalleVenta), y un producto puede estar en muchas ventas distintas.
* Un producto puede recibir varios ingresos a través de la clase Stock, ya que puede ser repuesto varias veces.
* Cada venta se paga con un único método, pero un mismo tipo de pago puede utilizarse en muchas ventas.

*Figura 5.1: Diagrama de Clases*

Diagrama, Esquemático

El contenido generado por IA puede ser incorrecto.

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

En la *Figura 5.2* se muestra el modelo de base de datos diseñado para el sistema de ventas e inventario de la Botillería Joaquín. Un modelo de base de datos, según Lucidchart (s.f.), describe la estructura lógica de una base de datos, incluyendo las relaciones e interacciones entre los datos. Es decir, permite visualizar cómo se distribuyen los datos en diferentes entidades (tablas) y cómo se conectan entre ellas para asegurar un almacenamiento ordenado y eficiente.

El modelo de datos implementado para este proyecto incluye tres tablas principales: usuario, producto y venta.

* Tabla usuario: almacena los datos de quienes utilizan el sistema. Cada usuario tiene un identificador único (usuario\_id) y se registran también su nombre, nombre de usuario, contraseña cifrada y rol (administrador o vendedor).
* Tabla producto: contiene la información detallada de cada producto disponible en el inventario. Se guarda un identificador único (producto\_id), el código de barras, nombre, tamaño, marca, precios de compra y venta, fechas de recepción y vencimiento, y el nombre del proveedor. Estos datos permiten registrar, actualizar y consultar el estado de los productos que se venden en la botillería.
* Tabla venta: registra cada transacción realizada. Incluye un identificador único (venta\_id), el total de la venta, el método de pago, la fecha y el identificador del usuario que realizó la operación. Un aspecto particular de esta tabla es el campo productos, de tipo TEXT, donde se almacena la lista de productos incluidos en la venta en formato JSON. Este texto guarda, por cada producto, su id, nombre, cantidad y subtotal. Por ejemplo: [{"id": "1", "nombre": "Coca-Cola", "cantidad": 2, "subtotal": 6000}]

Esta forma de almacenar los productos dentro de una venta simplifica la estructura de la base de datos para este primer incremento, ya que permite registrar múltiples productos sin crear una tabla adicional por ahora.

En cuanto a las relaciones, el modelo establece una relación de uno a muchos entre la tabla usuario y la tabla venta, ya que un mismo usuario puede realizar múltiples ventas en el sistema, pero cada venta está asociada únicamente a un usuario específico. Esto significa que, por cada registro en la tabla de usuarios, pueden existir varios registros en la tabla de ventas vinculados a él, lo que permite identificar claramente quién realizó cada transacción.

Por otro lado, la relación entre venta y producto corresponde a una relación de muchos a muchos, ya que una venta puede incluir varios productos distintos, y a su vez, un mismo producto puede estar presente en distintas ventas.

*Figura 5.2: Modelo de datos*

**Diagrama

El contenido generado por IA puede ser incorrecto.**

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto.

## Vista proceso

La vista de procesos, según Kruchten (1995), muestra cómo funciona el sistema mientras está en uso, enfocándose en cómo se comunican sus partes y qué pasa con cada acción que realiza el usuario. Esta vista considera aspectos como la rapidez del sistema, la forma en que se ejecutan varias tareas a la vez y cómo se conectan entre sí los distintos componentes.

En este proyecto, esta vista se representa mediante diagramas de secuencia, los cuales permiten visualizar, de forma ordenada y cronológica, cómo el usuario interactúa con la interfaz, cómo se procesan internamente esas acciones y cómo fluye la información entre las vistas, los controladores y la base de datos.

Para facilitar la lectura e interpretación de los diagramas, se utiliza un sistema de colores: verde para representar las vistas del sistema, azul para los controladores y naranjo para las tablas de la base de datos.

En la *Figura 5.3* se muestra el diagrama de secuencia del “Caso de Uso N°1: Registrando una venta”, uno de los flujos más importantes del sistema. El proceso comienza cuando el usuario se encuentra en la vista de inicio y selecciona la opción *“Crear Venta”* desde el menú lateral. Esta acción activa la vista V\_CrearVenta, la cual solicita al sistema el listado de productos disponibles, información que se obtiene directamente desde la base de datos.

Luego, el usuario va agregando productos al carrito, y el sistema actualiza automáticamente el total de la venta con cada nuevo producto añadido. A continuación, el usuario elige un método de pago, ya sea en efectivo o con tarjeta, y esta información se guarda de forma temporal mientras se completa la venta.

Finalmente, al presionar el botón *“Guardar Venta”*, el sistema reúne todos los datos (productos seleccionados, cantidades y método de pago), los registra en la base de datos y muestra un mensaje de confirmación indicando que la venta fue guardada exitosamente

*Figura 5.3: “Diagrama de secuencia del Caso de Uso N°1, Registrando una venta (Parte 1)”*

Interfaz de usuario gráfica

El contenido generado por IA puede ser incorrecto.

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

En la *Figura 5.4* se complementa el flujo del “Caso de Uso N°1: Registrando una venta”, mostrando el proceso que ocurre cuando el usuario accede al módulo *"Listado de Ventas"* para visualizar la información registrada. El flujo comienza en la vista de inicio, donde el usuario selecciona la opción correspondiente desde el menú lateral para ver las ventas realizadas.

En ese momento, el sistema carga la vista V\_ListaVentas y se inicia una serie de solicitudes internas: desde la vista se solicita al controlador de ventas los datos necesarios; este, a su vez, consulta la base de datos a través del componente C\_MySql, el cual accede a la tabla de ventas y devuelve la información solicitada. Una vez que los datos son recuperados, se muestran en pantalla para que el usuario pueda ver la venta recién ingresada junto con otras anteriores.

*Figura 5.4: “Diagrama de secuencia del Caso de Uso N°1, Registrando una venta (Parte 2)”*

*Diagrama

El contenido generado por IA puede ser incorrecto.*

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

En la *Figura 5.5* se muestra el diagrama de secuencia del “Caso de Uso N°2: Calculando el Total de la Venta”, enfocada en el cálculo del total de la venta a medida que se agregan productos.

El flujo comienza en la vista de inicio, donde el usuario selecciona la opción *“Crear Venta”*. El sistema carga la vista correspondiente y solicita al backend los datos de los productos. Esta información se obtiene desde la base de datos a través de una cadena de controladores, y se muestra al usuario en pantalla.

Cuando el usuario presiona el botón *“Agregar”* sobre un producto, el sistema solicita que se añada dicho producto a la lista de venta. Una vez agregado, el sistema recalcula automáticamente el total y lo muestra actualizado, permitiendo al usuario llevar el control del monto en tiempo real.

*Figura 5.5: “Diagrama de secuencia del Caso de Uso N°2, Calculando el total de la venta”.*

Interfaz de usuario gráfica, Diagrama

El contenido generado por IA puede ser incorrecto.Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

En la *Figura 5.6* se representa el flujo del “Caso de Uso N°4: Modificando cantidad de productos”, el cual ocurre durante la creación de una venta, cuando el administrador necesita ajustar manualmente la cantidad de un producto agregado al carrito.

El proceso comienza cuando el administrador accede a la vista *Crear Venta* desde el menú lateral. Una vez cargada la vista y mostrados los productos disponibles, el administrador selecciona un producto y lo agrega a la venta. Luego, decide modificar la cantidad asignada a ese producto.

El sistema recibe la nueva cantidad solicitada y verifica si existe stock suficiente para realizar el ajuste. Esta verificación se realiza consultando los datos de la tabla Productos a través del controlador C\_MySql. Si hay disponibilidad, se actualiza la cantidad del producto en la venta y se recalcula el total.

*Figura 5.6: “Diagrama de secuencia del Caso de Uso N°4, Modificando cantidad de productos”.*

Imagen que contiene Diagrama

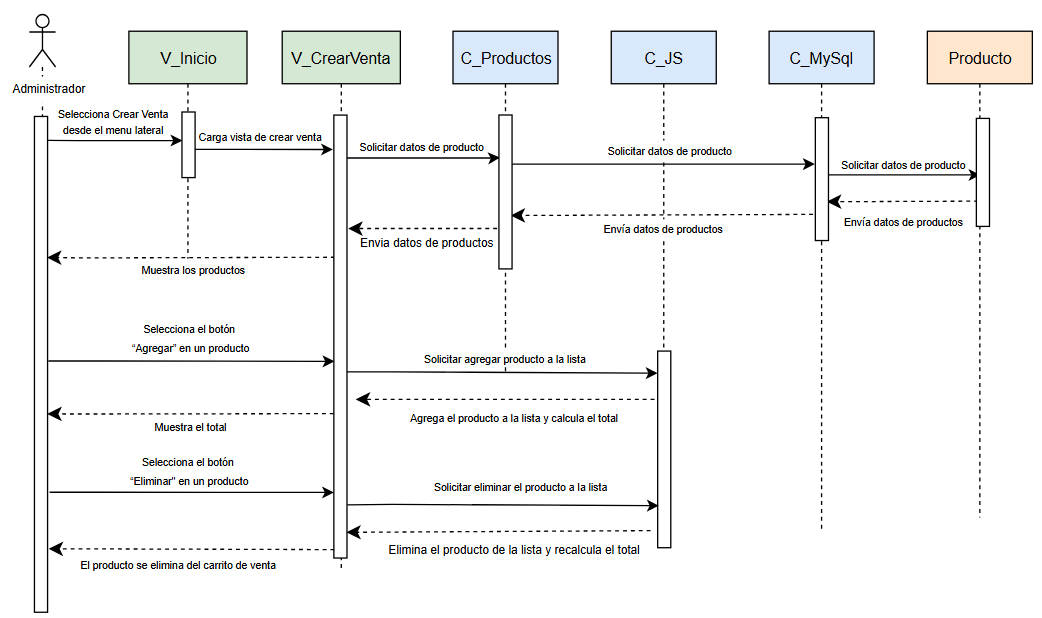
El contenido generado por IA puede ser incorrecto. Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

En la *Figura 5.7* se presenta el diagrama de secuencia del “Caso de Uso N°5: Eliminando productos de una venta”, que describe el flujo cuando un administrador elimina un producto del carrito durante la creación de una venta.

El proceso comienza cuando el administrador accede desde la vista de inicio a la opción *“Crear Venta”*. Una vez cargada la vista, el sistema solicita los datos de los productos y los muestra en pantalla. El administrador puede agregar un producto al carrito, tras lo cual el sistema actualiza y muestra el total de la venta.

Si decide eliminar el producto, selecciona la opción *“Eliminar”* y el sistema solicita al controlador de JavaScript que quite ese producto de la lista, lo elimina y recalcula el total actualizado, que luego se muestra en pantalla.

*Figura 5.7: “Diagrama de secuencia del Caso de Uso N°5, Eliminando productos de una venta”.*

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

La *Figura 5.8* representa el flujo del “Caso de Uso N°7: Descontando automáticamente el inventario”, que muestra cómo el sistema descuenta de forma automática las cantidades vendidas desde el inventario.

El proceso comienza cuando el usuario presiona el botón *“Guardar Venta”*. El sistema toma los productos vendidos y sus cantidades, y los envía a través del controlador de ventas hasta el componente de base de datos, que actualiza la tabla Producto descontando las unidades correspondientes. Una vez completado el proceso, se muestra un mensaje de confirmación en pantalla.

Luego, si el usuario selecciona la opción *“Inventario”*, el sistema realiza una nueva consulta para obtener los datos actualizados de los productos. Esta solicitud recorre los controladores y consulta nuevamente la base de datos, la cual devuelve las cantidades actuales. Finalmente, esta información se muestra en pantalla.

*Figura 5.8: “Diagrama de secuencia del Caso de Uso N°7, Descontando automáticamente el inventario”*

Interfaz de usuario gráfica, Diagrama

El contenido generado por IA puede ser incorrecto.Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

En la *Figura 5.9* se presenta el diagrama de secuencia del “Caso de Uso N°10: Seleccionando método de pago”, que muestra el flujo del sistema cuando el usuario selecciona un método de pago al momento de realizar una venta.

El proceso comienza cuando el usuario elige el método de pago desde la vista *Crear Venta*. Esta información es enviada al sistema, donde se guarda temporalmente. Luego, al presionar el botón *“Guardar Venta”*, el sistema reúne todos los datos necesarios, incluidos los productos y el método de pago, y los envía hacia la base de datos.

El componente encargado de la persistencia guarda los datos en la tabla Venta y confirma que la transacción fue registrada correctamente. Finalmente, se muestra un mensaje en pantalla confirmando que la venta fue guardada con éxito.

*Figura 5.9: “Diagrama de secuencia del Caso de Uso N°10, Seleccionando método de pago”*

Diagrama

El contenido generado por IA puede ser incorrecto.Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

En la *Figura 5.10* se muestra el diagrama de secuencia del “Caso de Uso N°11: Ingresando monto recibido y calcular cambio”, que representa el flujo cuando el usuario paga en efectivo e ingresa el monto recibido para que el sistema calcule el cambio.

El proceso comienza cuando el usuario selecciona el método de pago *“Efectivo”*. El sistema muestra un formulario específico para este tipo de pago, en el que el usuario ingresa el monto recibido. Esa información es enviada al sistema, el cual calcula automáticamente el monto a devolver y lo muestra en pantalla.

Luego, al presionar el botón *“Guardar Venta”*, el sistema envía todos los datos de la transacción, incluyendo el método de pago y el monto recibido, hasta la base de datos. Finalmente, se muestra un mensaje confirmando que la venta fue registrada correctamente.

*Figura 5.10: “Diagrama de secuencia del Caso de Uso N°11, Ingresando monto recibido y calcular cambio”*

Diagrama

El contenido generado por IA puede ser incorrecto.Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

En la *Figura 5.11* se presenta el diagrama de secuencia del “Caso de Uso N°20: Visualizando stock disponible”, que representa el flujo de acciones que realiza el sistema cuando un usuario consulta el stock disponible de productos.

El proceso inicia cuando el usuario selecciona la opción *“Inventario”* desde el menú lateral. Esto activa la vista V\_Inventario, que solicita al sistema el listado completo de productos. Esta solicitud es procesada por el controlador de productos, que consulta a la base de datos a través del componente C\_MySql. Luego, se accede a la tabla Producto, desde donde se obtiene el listado actualizado de productos junto con sus cantidades.

Una vez recuperada la información, el sistema la muestra al usuario en la interfaz de inventario, permitiéndole visualizar fácilmente el stock disponible.

*Figura 5.11: “Diagrama de secuencia del Caso de Uso N°20, Visualizando stock disponible”*

Diagrama

El contenido generado por IA puede ser incorrecto.Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

En la *Figura 5.12* se muestra el diagrama de secuencia del “Caso de Uso N°23: Registrando nuevos productos”, correspondiente al proceso de registro de nuevos productos por parte del administrador del sistema.

El flujo comienza cuando el administrador selecciona la opción *“Agregar producto”*. El sistema muestra un formulario, donde se ingresan los datos del nuevo producto (como nombre, precio y cantidad). Una vez completado el formulario, se envía la información al controlador correspondiente, que procesa los datos y los envía al componente que maneja la conexión con la base de datos.

Finalmente, los datos son almacenados en la tabla Producto, y el sistema devuelve una confirmación indicando que el nuevo producto fue registrado correctamente. Este mensaje se muestra en pantalla para informar al administrador del éxito del proceso.

*Figura 5.12: “Diagrama de secuencia del Caso de Uso N°23, Registrando nuevos productos”*

Diagrama

El contenido generado por IA puede ser incorrecto.

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

La *Figura 5.13* presenta el diagrama de secuencia del “Caso de Uso N°24: Registrando ingreso de stock de productos existentes”, que representa el flujo cuando el administrador registra el ingreso de nuevas unidades a un producto ya existente en el sistema.

El proceso comienza cuando el administrador selecciona la opción *“Agregar Stock”* desde el menú lateral. El sistema carga la vista correspondiente y solicita al backend el listado de productos. Una vez mostrados, el administrador selecciona el producto al que desea agregar stock, y el sistema muestra un formulario donde se ingresa la cantidad deseada.

Esta información es enviada al controlador, que comunica los datos al componente de base de datos. El sistema actualiza la tabla Producto, sumando la cantidad nueva al stock actual. Finalmente, se devuelve una confirmación y se muestra un mensaje indicando que el stock fue actualizado correctamente.

*Figura 5.13: “Diagrama de secuencia del Caso de Uso N°24, Registrando ingreso de stock de productos existentes”*

Imagen que contiene Diagrama

El contenido generado por IA puede ser incorrecto.Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

La *Figura 5.14* muestra el diagrama de secuencia del “Caso de Uso N°40: Creando usuarios”, correspondiente al flujo que se ejecuta cuando el administrador registra un nuevo usuario en el sistema.

El proceso comienza cuando el administrador selecciona la opción *“Agregar Usuario”* en el módulo de gestión de usuarios. El sistema despliega un formulario en el que se ingresan los datos del nuevo usuario. Al enviar el formulario, la información es enviada al backend para validar si el usuario ya existe.

Si no hay duplicados, el sistema guarda los datos en la base de datos y responde con una confirmación de que el registro fue exitoso. Finalmente, se muestra una notificación al administrador confirmando que el nuevo usuario fue creado correctamente.

*Figura 5.14: “Diagrama de secuencia del Caso de Uso N°40, Creando usuarios”*

*Diagrama

El contenido generado por IA puede ser incorrecto.*Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

En la *Figura 5.15* se muestra el diagrama de secuencia del “Caso de Uso N°41: Modificando usuarios”, que describe el flujo que se ejecuta cuando un administrador edita los datos de un usuario registrado en el sistema.

El proceso comienza cuando el administrador selecciona la opción *“Editar usuario”*. El sistema solicita los datos actuales del usuario y despliega el formulario con la información precargada. El administrador modifica los campos necesarios y envía el formulario actualizado.

El controlador de usuarios envía los nuevos datos al componente que gestiona la base de datos. El cual valida y guarda la información actualizada en la tabla Usuario, y luego devuelve una confirmación. Finalmente, el sistema muestra un mensaje indicando que los cambios fueron aplicados con éxito.

Este flujo permite mantener actualizada la información de los usuarios registrados, garantizando que cualquier ajuste sea correctamente validado y almacenado.

*Figura 5.15: “Diagrama de secuencia del Caso de Uso N°41, Modificando usuarios”*

Diagrama

El contenido generado por IA puede ser incorrecto.

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

En la *Figura 5.16* se presenta el diagrama de secuencia del “Caso de Uso N°42: Eliminando usuario”, que describe el flujo cuando un administrador elimina un usuario del sistema.

El proceso se inicia cuando el administrador selecciona la opción *“Eliminar usuario”* en la vista de gestión. El sistema solicita confirmación antes de continuar. Una vez confirmada la acción, se envía la solicitud al backend con el ID del usuario a eliminar.

El controlador de usuarios gestiona la operación y comunica la solicitud al componente encargado de la base de datos. Allí se ejecuta la eliminación en la tabla Usuario, y se retorna una confirmación de que el proceso fue exitoso. Finalmente, se muestra un mensaje notificando al administrador que el usuario ha sido eliminado correctamente.

*Figura 5.16: “Diagrama de secuencia del Caso de Uso N°42, Eliminando usuario”*

Diagrama

El contenido generado por IA puede ser incorrecto.

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

En la *Figura 5.17* se muestra el diagrama de secuencia del “Caso de Uso N°43: Cambiando rol de usuario”, que describe el proceso que se ejecuta cuando un administrador modifica el rol de un usuario en el sistema.

El flujo comienza cuando el administrador selecciona la opción *“Editar usuario”* desde la vista de gestión. El sistema solicita los datos actuales del usuario, los recupera desde la base de datos y despliega un formulario con la información correspondiente. El administrador modifica únicamente el campo relacionado al rol del usuario (por ejemplo, cambiar de “Vendedor” a “Administrador”) y envía el formulario actualizado.

El sistema valida y guarda los nuevos datos en la base de datos, actualizando los permisos del usuario según su nuevo rol. Finalmente, se muestra un mensaje confirmando que la modificación fue realizada con éxito.

*Figura 5.17: “Diagrama de secuencia del Caso de Uso N°43, Cambiando rol de usuario”*

Imagen que contiene Diagrama

El contenido generado por IA puede ser incorrecto.

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

La *Figura 5.18* muestra el diagrama de secuencia del “Caso de Uso N°45: Iniciando sesión en el sistema”, que representa el proceso de inicio de sesión de un usuario en el sistema.

El flujo comienza cuando el usuario ingresa sus credenciales (nombre de usuario y contraseña) en la vista de login. Esta información se envía al controlador de usuarios, que a su vez la remite al componente de base de datos para validar los datos.

El sistema verifica la autenticidad de las credenciales consultando la tabla Usuario. Si la información es correcta, se devuelve una respuesta de autorización exitosa, y el sistema permite el acceso al usuario.

*Figura 5.18: "Diagrama de secuencia del Caso de Uso N°45, Iniciando sesión en el sistema”*

Diagrama

El contenido generado por IA puede ser incorrecto.Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

## Vista de desarrollo

La vista de desarrollo, según Kruchten (1995), muestra el sistema desde la perspectiva del programador y se centra en su organización interna durante el proceso de construcción. Esta vista describe la estructura estática del software dentro del entorno de desarrollo, incluyendo la distribución del código en carpetas, módulos, bibliotecas y componentes reutilizables.

En este proyecto, esta vista se representa mediante un diagrama de componentes, el cual muestra cómo un sistema se divide en componentes funcionales y cómo estos interactúan a través de interfaces y dependencias. Este tipo de diagrama ayuda a comprender la arquitectura general del sistema, mostrando de forma clara qué partes lo componen y cómo están conectadas entre sí, facilitando tanto el desarrollo como el mantenimiento del software (Lucidchart, s.f.).

La *Figura 5.19* muestra el diagrama de despliegue del sistema, que representa cómo están distribuidos físicamente los distintos componentes del proyecto en el entorno donde funciona.

En el diagrama, se puede ver que el usuario accede al sistema desde su navegador, ingresando a la página web a través de una conexión HTTP. Esta parte del sistema corresponde a la vista del cliente, que está construida con tecnologías como HTML, JavaScript y CSS, encargadas de mostrar la interfaz visual con la que se interactúa.

Todas las solicitudes que hace el usuario se envían a un servidor web, que en este caso utiliza Apache. Dentro de este servidor se encuentra la lógica del sistema, organizada en distintos controladores, como ventas, productos, usuarios, y principal. Estos controladores procesan las acciones del usuario, como registrar una venta o consultar el stock.

El servidor también tiene una conexión a la base de datos, que permite comunicarse con MySQL, el sistema donde se almacena toda la información importante, como productos, usuarios y ventas. Además, el servidor puede ser administrado usando herramientas como Cpanel, que sirven para tareas de configuración y mantenimiento.

*Figura 5.19: Diagrama de componentes*

Diagrama

El contenido generado por IA puede ser incorrecto.

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

## Vista física

La vista física según Kruchten (1995), describe cómo se organizan físicamente los componentes del software en el hardware donde se ejecutan. Esta vista es clave para entender cómo se distribuye la aplicación en distintos dispositivos o servidores, mostrando la infraestructura que soporta el sistema en tiempo real, así como las conexiones entre sus partes físicas.

En este proyecto, esta vista se representa mediante un diagrama de despliegue, el cual permite visualizar cómo los componentes del sistema están distribuidos físicamente en distintos entornos, como el cliente, el servidor y la base de datos. Este tipo de diagrama. según Lucidchart, ayuda a los equipos a visualizar el hardware físico de un sistema, los artefactos de software que se ejecutan en él y cómo se comunican esos elementos entre sí ([Lucidchart, s.f.](https://www.lucidchart.com/pages/es/tutorial-de-diagramas-de-despliegue)).

En la *Figura 5.20* se muestra el diagrama de despliegue del sistema de ventas e inventario desarrollado. Este diagrama representa de manera clara cómo están distribuidos físicamente los diferentes componentes de la aplicación, especificando tanto los requisitos técnicos como las funcionalidades que se ejecutan en cada parte del sistema. Se distinguen tres nodos principales: PC Usuario, Servidor Web y Base de Datos.

* PC Usuario: corresponde al computador desde el cual un vendedor o administrador accede al sistema. Este equipo solo requiere tener acceso a Internet y un navegador web compatible. Desde aquí se ejecutan funcionalidades visibles para el usuario, como iniciar sesión, crear una venta, agregar productos al carrito, registrar usuarios, gestionar el inventario, entre otras. Estas acciones se reflejan como comandos que luego son enviados al servidor para ser procesados.
* Servidor Web: es donde está alojado el sistema principal. Este servidor debe contar con características mínimas como un procesador Intel Core i5 (o superior), al menos 2GB de RAM, almacenamiento SSD y conexión a Internet. En él se ejecuta el sistema operativo, el servidor Apache, MySQL, PHP y herramientas de administración como cPanel. Aquí también residen los controladores del sistema (por ejemplo, ventascontrolador, usuarioscontrolador), que se encargan de procesar las peticiones del usuario y ejecutar la lógica necesaria, como registrar ventas, calcular cambio, editar productos, o eliminar usuarios.
* Base de Datos: almacena toda la información estructurada del sistema (productos, ventas, usuarios, etc.). Se conecta directamente con el servidor a través de MySQL. Su función es entregar y guardar los datos requeridos por el sistema, y su único requisito técnico es tener conectividad de red e integración con el servidor web.

Las líneas de conexión indican cómo se comunican entre sí estos tres elementos: el usuario accede al sistema vía HTTP desde su navegador al servidor web, y este último se comunica con la base de datos mediante una conexión MySQL, lo que permite recuperar o guardar la información necesaria.

*Figura 5.20: Diagrama de Despliegue*

Diagrama

El contenido generado por IA puede ser incorrecto.Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

# ÁRBOL DE NAVEGACIÓN

Según InboundCycle (2014), un árbol de navegación se trata de un esquema jerárquico que permite entender de forma rápida y ordenada qué opciones tiene el usuario y desde dónde puede acceder a cada una. Este tipo de árbol ayuda a visualizar la estructura interna del sistema y a mejorar la experiencia de uso, ya que facilita la navegación al mostrar cómo están conectadas las vistas y acciones disponibles.

En la *Figura 6.1* se muestra el árbol de navegación del sistema desarrollado. El recorrido comienza por el acceso al sistema, luego el inicio de sesión y finalmente la carga de la vista llamada Inicio, que es la primera pantalla que ve el usuario. Si bien esta vista no tiene funciones propias, cumple un rol importante como punto de entrada, desde donde el usuario puede acceder a las distintas secciones del sistema utilizando el menú lateral.

En este árbol se muestra que todas las secciones del sistema se abren desde la vista llamada Inicio, para que se entienda mejor cómo funciona la navegación. Cada opción del menú (como “Crear venta”, “Inventario” o “Administrar usuarios”) lleva a una pantalla distinta, donde se realiza una acción específica. Esta forma de mostrarlo se eligió para que combine con los diagramas de secuencia del proyecto, donde se ve que el usuario hace clic en una opción y el sistema carga una nueva pantalla. Aunque en la práctica se puede ir de una sección a otra desde cualquier parte usando el menú lateral, se decidió representarlo así para que sea más fácil entender el recorrido del sistema.

* Crear venta: Agregar/quitar productos, modificar cantidades, seleccionar el método de pago y guardar la venta.
* Lista de ventas: Ver el historial de ventas y sus detalles.
* Inventario: Revisar la lista de productos y su stock.
* Crear productos: Registrar nuevos productos o eliminar los existentes.
* Añadir stock: Aumentar la cantidad de un producto que ya está registrado.
* Administrar usuarios: Agregar, editar, eliminar o ver usuarios registrados en el sistema.

*Figura 6.1: Árbol de Navegación*

Diagrama

El contenido generado por IA puede ser incorrecto.

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

Para una descripción más detallada del funcionamiento de cada vista del sistema, ver el Anexo A: Detalle de vistas del sistema.

# CASOS DE PRUEBA

En esta sección se presentan los casos de prueba aplicados a las distintas funciones del sistema, con el fin de comprobar que cada una responde correctamente ante las acciones del usuario. Para cada caso, se indica qué se va a probar, cuál es el resultado que se espera obtener y cuál fue el resultado real al hacer la prueba, para así, verificar que el sistema funcione bien y detectar posibles errores antes de su uso final.

## Caso de prueba CU01

Este caso de prueba tiene como objetivo verificar que el sistema permita registrar una venta de forma correcta, desde la selección de productos hasta la confirmación final.

**Caso de Uso N°1:** Registrando una venta

**Casos de Prueba:**

1. Verificar que al hacer clic en "Crear Venta" se despliegue correctamente la vista de registro de ventas, mostrando las secciones para agregar productos, visualizar el total y seleccionar el método de pago.
2. Verificar que se puedan agregar productos al carrito de venta.
3. Verificar que el sistema calcule correctamente el subtotal de la venta.
4. Verificar que se habilite el campo de monto recibido al seleccionar el método de pago "Efectivo".
5. Verificar que la venta se registre correctamente al confirmar.
6. Verificar que la venta se muestre correctamente en la vista "Listado de Ventas".

**Resultados Esperados:**

1. Al hacer clic en "Crear Venta", el sistema debe mostrar la vista con las secciones para agregar productos, ver el total de la venta, seleccionar el método de pago y confirmar la venta.
2. El sistema permite agregar los productos al carrito con cantidad ajustable.
3. El sistema actualiza y muestra automáticamente el subtotal al agregar productos.
4. Al seleccionar "Efectivo" como método de pago, se habilita el campo para ingresar el monto recibido.
5. Al confirmar la venta, el sistema la registra con todos los datos ingresados (fecha, productos, cantidades, total, método de pago).
6. La venta registrada aparece en la vista "Listado de Ventas" con todos sus datos visibles.

**Resultados Obtenidos:**

1. La vista de creación de ventas se muestra correctamente al hacer clic en "Crear Venta", permitiendo al usuario realizar todo el flujo de venta.
2. Los productos se pueden agregar sin errores al carrito.
3. El subtotal se calcula automáticamente y refleja el total correcto de los productos agregados.
4. El campo de monto recibido se habilita correctamente al seleccionar el método "Efectivo".
5. La venta se registra correctamente y se guarda en la base de datos.
6. La venta recién realizada aparece correctamente en el "Listado de Ventas" con sus datos completos.

**Evidencias**

A continuación, se presentan las evidencias correspondientes a cada caso de prueba, con el objetivo de confirmar que el sistema responde correctamente frente a las acciones realizadas por el usuario.

1. **Caso de Prueba 1: Visualización de la vista "Crear venta"**

En la *Figura 7.1* se puede ver que al hacer clic en la opción "Crear Venta" del menú lateral, el sistema carga correctamente la vista para registrar una venta. Esta pantalla muestra el espacio para agregar productos, el campo donde se visualiza el total, la opción para elegir el método de pago y también el botón para guardar la venta, todo en un solo lugar. Esto confirma que la vista se carga completa y permite al usuario iniciar el proceso de venta sin problemas, tal como se esperaba en esta prueba.

*Figura 7.1: “Evidencia Caso de Prueba 1, Caso de Uso N°1”*

**Captura de pantalla de computadora

El contenido generado por IA puede ser incorrecto.**

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

1. **Caso de Prueba 2: Agregado de productos**

En la *Figura 7.2* se muestra el momento en que se agregan productos al carrito de venta. Al hacer clic en el botón "Agregar", el sistema añade correctamente el producto seleccionado, mostrando su nombre, precio y una columna para ajustar la cantidad. Esta acción se repite sin errores, lo que confirma que la funcionalidad está operando bien y que el sistema responde como se esperaba cuando el usuario agrega productos a una venta.

*Figura 7.2: “Evidencia Caso de Prueba 2, Caso de Uso N°1”*

Captura de pantalla de un celular

El contenido generado por IA puede ser incorrecto.Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

1. **Caso de Prueba 3: Cálculo automático del total**

En la *Figura 7.3* se puede observar que el sistema actualiza automáticamente el total de la venta a medida que se agregan productos al carrito. No es necesario hacer ningún cálculo manual ni presionar otro botón, ya que el monto total se va actualizando solo, mostrando el valor correcto según la cantidad y precio de cada producto.

*Figura 7.3: “Evidencia Caso de Prueba 3, Caso de Uso N°1”*

Captura de pantalla de un celular

El contenido generado por IA puede ser incorrecto.

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

1. **Caso de Prueba 4: Método de pago y habilitación de campos**

En la *Figura 7.4* se puede comparar cómo cambia la interfaz del sistema según el método de pago que se seleccione. A la izquierda, cuando se elige "Tarjeta", solo se muestra el total de la venta, ya que no se necesita ingresar más información. En cambio, a la derecha, al seleccionar "Efectivo", se habilitan automáticamente los campos "Recibido" y "Vuelto", permitiendo ingresar cuánto dinero entregó el cliente y mostrar el cambio correspondiente. Esto confirma que el sistema se adapta correctamente al tipo de pago seleccionado, mostrando solo los campos necesarios en cada caso.

*Figura 7.4: “Evidencia Caso de Prueba 4, Caso de Uso N°1”*

Captura de pantalla de un celular

El contenido generado por IA puede ser incorrecto.Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

1. **Caso de Prueba 5: Registro de venta**

En la *Figura 7.5* se puede ver que al hacer clic en el botón "Guardar venta", el sistema muestra un mensaje de confirmación que indica que la venta fue registrada correctamente. Esto significa que todos los datos ingresados, como los productos, cantidades, total y método de pago, fueron guardados sin problemas. El mensaje confirma que todo salió bien y le da seguridad al usuario de que la venta fue completada.

*Figura 7.5: “Evidencia Caso de Prueba 5, Caso de Uso N°1”*

Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

1. **Caso de Prueba 6: Visualización en el listado de ventas**

En la *Figura 7.6* se muestra que, una vez registrada la venta, esta aparece de inmediato en el listado de ventas. Se puede ver que el sistema muestra correctamente todos los datos asociados, como la fecha y hora, los productos vendidos, el total y el método de pago. Esto permite al usuario revisar rápidamente que la información fue guardada de forma correcta y que quedó registrada en el historial sin errores.

*Figura 7.6: “Evidencia Caso de Prueba 6, Caso de Uso N°1”*

Tabla, Sitio web

El contenido generado por IA puede ser incorrecto.Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

## Caso de prueba CU02

Este caso de prueba tiene como objetivo validar el comportamiento del sistema al calcular el total de una venta.

**Caso de Uso N°2: Calculando el total de la venta**

**Casos de Prueba:**

1. Verificar que el sistema calcule automáticamente el subtotal al agregar productos al carrito.
2. Verificar que el sistema actualice correctamente el subtotal al modificar la cantidad de productos.
3. Verificar que el sistema recalcule el total al eliminar un producto del carrito.

**Resultados Esperados:**

1. Al agregar uno o más productos al carrito, el sistema calcula y muestra automáticamente el subtotal correspondiente.
2. Al modificar la cantidad de un producto en el carrito, el sistema recalcula el subtotal sin necesidad de acciones adicionales.
3. Al eliminar un producto del carrito, el sistema actualiza el total final de la venta, reflejando correctamente los productos restantes.

**Resultados Obtenidos:**

1. El sistema calcula automáticamente el subtotal al agregar productos al carrito, sin necesidad de refrescar la vista.
2. Al modificar la cantidad de productos, el subtotal se actualiza de forma inmediata y correcta.
3. Al eliminar un producto del carrito, el sistema actualiza de inmediato el total de la venta, mostrando correctamente los montos restantes.

**Evidencias**

A continuación, se presentan las evidencias correspondientes a cada caso de prueba, con el objetivo de confirmar que el sistema responde correctamente frente a las acciones realizadas por el usuario.

1. **Caso de Prueba 1: Cálculo automático al agregar productos**

En la *Figura 7.7* se puede ver cómo el sistema actualiza automáticamente el subtotal de la venta al momento de agregar productos al carrito. En la imagen de la izquierda, el total está en cero porque no hay productos seleccionados. En la imagen de la derecha, luego de agregar dos productos, el sistema muestra el nuevo total sin que el usuario tenga que hacer nada más. Esto confirma que la lógica de cálculo automático funciona correctamente y entrega información clara al vendedor durante el proceso.

*Figura 7.7: “Evidencia Caso de Prueba 1, Caso de Uso N°2”*

Captura de pantalla de un celular

El contenido generado por IA puede ser incorrecto.

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

1. **Caso de Prueba 2: Recálculo al modificar la cantidad**

En la *Figura 7.8* se puede comparar cómo el sistema recalcula automáticamente el subtotal de la venta al modificar la cantidad de productos en el carrito. A la izquierda, se ve la venta con una cantidad inicial, y a la derecha, después de aumentar la cantidad de productos, el total se actualiza de forma inmediata. Esto confirma que el sistema responde correctamente a los cambios y mantiene el valor total actualizado sin que el usuario tenga que hacer nada extra.

*Figura 7.8: " Evidencia Caso de Prueba 2, Caso de Uso N°2”*

Captura de pantalla de un celular

El contenido generado por IA puede ser incorrecto.

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

1. **Caso de Prueba 3: Recalculo al eliminar un producto**

En la *Figura 7.9* se puede observar que el sistema recalcula correctamente el total de la venta al eliminar un producto del carrito. En la primera pantalla, la venta incluye dos productos (Coca-Cola y Vino Blanco), con un total de $14.000. En la segunda pantalla, luego de eliminar el producto “Vino Blanco”, el sistema actualiza automáticamente el total a $6.000, correspondiente únicamente a las dos unidades de Coca-Cola. Este comportamiento confirma que el sistema responde correctamente ante la eliminación de productos, permitiendo al usuario verificar y confirmar el monto final de la venta sin errores ni confusión.

*Figura 7.9: "Evidencia Caso de Prueba 3, Caso de Uso N°2”*

Captura de pantalla de un celular

El contenido generado por IA puede ser incorrecto.

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

## Caso de prueba CU04

Este caso de prueba tiene como objetivo verificar que el sistema permita modificar la cantidad de productos dentro del carrito de ventas, asegurando que el subtotal y el total de la venta se actualicen correctamente en tiempo real.

**Caso de Uso N°4:** Modificando cantidad de productos

**Casos de Prueba:**

1. Verificar que el sistema permita modificar manualmente la cantidad de un producto en el carrito.
2. Verificar que el sistema recalcula automáticamente el subtotal y total al cambiar la cantidad.

**Resultados Esperados:**

1. El campo de cantidad en el carrito permite ser editado directamente por el actor.
2. Al cambiar la cantidad de un producto, el sistema recalcula automáticamente el subtotal y total de la venta.

**Resultados Obtenidos:**

1. El sistema permite modificar manualmente la cantidad de los productos en el carrito.
2. Al cambiar la cantidad, el total se actualiza automáticamente sin errores.

**Evidencias**

A continuación, se presentan las evidencias correspondientes a cada caso de prueba, con el objetivo de confirmar que el sistema responde correctamente frente a las acciones realizadas por el usuario.

1. **Caso de Prueba 1: Modificación manual de la cantidad**

En la *Figura 7.10* se puede observar que el sistema permite modificar directamente la cantidad de cada producto desde el campo numérico del carrito de venta. El usuario puede escribir el número deseado, sin necesidad de usar los botones de “+” o “–”, lo que agiliza el proceso si quiere ingresar cantidades mayores o más precisas. Esto demuestra que el campo de cantidad está habilitado para edición manual y que el sistema responde correctamente al cambio.

*Figura 7.10: "Evidencia Caso de Prueba 1, Caso de Uso N°4”*

Captura de pantalla de un celular

El contenido generado por IA puede ser incorrecto.

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

1. **Caso de Prueba 2: Recalculo automático del total**

En la *Figura 11* se muestra que, al modificar la cantidad de productos en el carrito, el sistema actualiza automáticamente el total de la venta. A la izquierda se ve el total inicial con una unidad del producto, y a la derecha, después de cambiar manualmente la cantidad a cinco, el total se ajusta sin necesidad de presionar otro botón. Esto confirma que el cálculo del total funciona de forma automática y correcta al cambiar cantidades.

*Figura 7.11: "Evidencia Caso de Prueba 2, Caso de Uso N°4”*

Captura de pantalla de un celular

El contenido generado por IA puede ser incorrecto.

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

## Caso de prueba CU05

Este caso de prueba tiene como objetivo verificar que el sistema permita eliminar productos del carrito de ventas, asegurando que el total de la venta se recalcule automáticamente y refleje correctamente los productos restantes.

**Caso de Uso N°5:** Eliminando productos de una venta

**Casos de Prueba:**

1. Verificar que el sistema muestre correctamente los productos agregados al carrito.
2. Verificar que el sistema permita eliminar un producto específico del carrito de venta y que el sistema actualice el total de la venta.

**Resultados Esperados:**

1. El sistema muestra una tabla con los productos agregados y un botón para eliminar cada uno.
2. Al hacer clic en el botón de eliminar, el producto desaparece del carrito y el total de la venta se recalcula automáticamente.

**Resultados Obtenidos:**

1. El sistema despliega correctamente los productos agregados con opción de eliminación.
2. Al eliminar un producto, este desaparece del carrito inmediatamente y se recalcula automáticamente el total.

**Evidencias**

A continuación, se presentan las evidencias correspondientes a cada caso de prueba, con el objetivo de confirmar que el sistema responde correctamente frente a las acciones realizadas por el usuario.

1. **Caso de Prueba 1: Visualización de productos en el carrito**

En la *Figura 7.12* se puede ver que el sistema muestra correctamente todos los productos añadidos al carrito de venta. Cada producto aparece con su nombre, cantidad, precio y la opción de eliminarlo si es necesario. Esta vista permite al usuario revisar lo que ha seleccionado antes de finalizar la venta y tener control total sobre la información mostrada.

*Figura 7.12: "Evidencia Caso de Prueba 1, Caso de Uso N°5”*

Captura de pantalla de un celular

El contenido generado por IA puede ser incorrecto.

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

1. **Caso de Prueba 2: Eliminación de producto desde el carrito**

En la *Figura 7.13* se muestra el funcionamiento correcto del sistema al eliminar un producto específico del carrito de ventas. A la izquierda se observa el carrito con dos productos y un total de $30.880. Luego, al presionar el botón rojo "Eliminar" sobre uno de ellos, dicho producto desaparece de la lista, y el total se actualiza automáticamente a $23.990, correspondiente únicamente al ítem restante.

*Figura 7.13: "Evidencia Caso de Prueba 2, Caso de Uso N°5”*

Captura de pantalla de un celular

El contenido generado por IA puede ser incorrecto.

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

## Caso de prueba CU07

Este caso de prueba tiene como objetivo verificar que el sistema descuente automáticamente las cantidades correspondientes del inventario al momento de completar una venta, asegurando que el stock disponible se actualice correctamente y en tiempo real.

**Caso de Uso N°7:** Descontando automáticamente el inventario

**Casos de Prueba:**

1. Verificar que el sistema identifique correctamente los productos vendidos y sus cantidades.
2. Verificar que el sistema descuente automáticamente del inventario las cantidades vendidas y que el nuevo stock se refleje correctamente en el módulo de inventario.

**Resultados Esperados:**

1. Al finalizar una venta, el sistema detecta automáticamente qué productos se vendieron y cuántas unidades de cada uno.
2. El sistema descuenta automáticamente del stock las cantidades vendidas y el módulo de inventario refleja correctamente el nuevo stock actualizado tras la venta.

**Resultados Obtenidos:**

1. El sistema identifica correctamente los productos involucrados en la venta y sus respectivas cantidades.
2. El sistema actualiza automáticamente el inventario al finalizar la venta, restando del stock la cantidad correspondiente de cada producto, lo que se refleja correctamente al ingresar al módulo de inventario.

**Evidencias**

A continuación, se presentan las evidencias correspondientes a cada caso de prueba, con el objetivo de confirmar que el sistema responde correctamente frente a las acciones realizadas por el usuario.

1. **Caso de Prueba 1: Identificación de productos vendidos.**

Cuando se finaliza una venta, el sistema guarda correctamente qué productos fueron vendidos y en qué cantidad. Esta información queda registrada en la base de datos y luego se puede consultar en el listado de ventas. Esto es clave para llevar un control preciso del inventario y revisar con claridad el detalle de cada transacción.

En la *Figura 7.14* se puede ver cómo el sistema muestra cada producto vendido con su respectiva cantidad, junto con la información del vendedor, el total, el método de pago y la fecha. Esto confirma que el sistema cumple correctamente con el registro detallado de los productos vendidos.

*Figura 7.14: "Evidencia Caso de Prueba 1, Caso de Uso N°7”*

Interfaz de usuario gráfica, Tabla, Sitio web

El contenido generado por IA puede ser incorrecto.

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

1. **Caso de Prueba 2: Descuento automático y actualización del stock en inventario**

Después de registrar una venta, el sistema descuenta automáticamente las unidades correspondientes del inventario y al ingresar al módulo de inventario, se puede ver que el stock actualizado refleja correctamente esa disminución.

En la *Figura 7.15* se muestra el producto Coca-Cola antes de la venta, con un stock de 78 unidades. Luego, en la *Figura 7.16* se observa cómo se realiza una venta por 5 unidades del mismo producto. Finalmente, en la *Figura 7.17* se ingresa al módulo de inventario y se verifica que el stock bajó a 73, lo que confirma que el sistema hizo el descuento de forma automática y reflejó el nuevo stock correctamente.

*Figura 7.15: "Evidencia 1 Caso de Prueba 2, Caso de Uso N°7”*

Tabla

El contenido generado por IA puede ser incorrecto.

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

*Figura 7.16: "Evidencia 2 Caso de Prueba 2, Caso de Uso N°7”*

Captura de pantalla de un celular

El contenido generado por IA puede ser incorrecto.

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

*Figura 7.17: "Evidencia 3 Caso de Prueba 2, Caso de Uso N°7”*

Tabla

El contenido generado por IA puede ser incorrecto.

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

## Caso de prueba CU10

Este caso de prueba tiene como objetivo verificar que el sistema permita seleccionar correctamente un método de pago al momento de registrar una venta, asegurando que esta elección se refleje adecuadamente en el comprobante y en el flujo de la transacción.

**Caso de Uso N°10:** Seleccionando método de pago

**Casos de Prueba:**

1. Verificar que el sistema muestre correctamente las opciones de método de pago.
2. Verificar que se registre correctamente el método de pago seleccionado.

**Resultados Esperados:**

1. El sistema despliega correctamente las opciones "Tarjeta" y "Efectivo" cuando se selecciona método de pago.
2. Al seleccionar una opción, el sistema registra correctamente el tipo de pago en la venta.

**Resultados Obtenidos:**

1. Las opciones de método de pago se despliegan correctamente en la interfaz de venta.
2. El sistema registra correctamente si se paga con tarjeta o efectivo.

**Evidencias**

A continuación, se presentan las evidencias correspondientes a cada caso de prueba, con el objetivo de confirmar que el sistema responde correctamente frente a las acciones realizadas por el usuario.

1. **Caso de Prueba 1: Visualización de métodos de pago**

En la *Figura 7.18* se muestra cómo el sistema despliega correctamente las opciones de método de pago dentro del formulario de ventas. Al seleccionar esta sección, aparecen de forma clara las opciones “Efectivo” y “Tarjeta”, lo que permite al usuario elegir el tipo de pago sin problemas.

*Figura 7.18: "Evidencia Caso de Prueba 1, Caso de Uso N°10”*

Captura de pantalla de un celular

El contenido generado por IA puede ser incorrecto.

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

1. **Caso de Prueba 2: Registro del método de pago seleccionado**

En esta imagen se puede ver que el método de pago seleccionado es “Efectivo”, lo que hace que aparezca el campo “Recibido” para ingresar el dinero entregado por el cliente. En este caso, se ingresó un monto de $10.000 para una venta cuyo total es $3.000. De forma inmediata, el sistema calcula y muestra el vuelto correspondiente ($7.000) en el campo “Vuelto”, sin necesidad de recargar la página ni realizar operaciones externas.

Como se aprecia en la Figura 6.14, una vez realizada la venta, el sistema guarda correctamente el método de pago que se seleccionó. En este caso, la tabla del listado de ventas muestra que se utilizó “Efectivo”, reflejando que la elección fue registrada de forma correcta por el sistema.

*Figura 7.19: "Evidencia Caso de Prueba 2, Caso de Uso N°10”*

Tabla

El contenido generado por IA puede ser incorrecto.

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

## Caso de prueba CU11

Este caso de prueba tiene como objetivo verificar que el sistema permita ingresar el monto recibido por el cliente y calcule automáticamente el cambio a entregar, asegurando que el valor mostrado sea correcto y coherente con el total de la venta.

**Caso de Uso N°11:** Ingresando monto recibido y calculando cambio

**Casos de Prueba:**

1. Verificar que el sistema muestre el campo "Monto recibido" solo si se seleccionó pago en efectivo.
2. Verificar que el actor pueda ingresar el monto entregado por el cliente y que el sistema calcule y muestre correctamente el cambio cuando el monto recibido es mayor al total a pagar.

**Resultados Esperados:**

1. El campo para ingresar el monto recibido aparece solamente si el método de pago seleccionado es “Efectivo”.
2. El sistema permite ingresar valores numéricos válidos en el campo de monto recibido y, si el valor ingresado es mayor al total de la venta, calcula automáticamente la diferencia y muestra el cambio correspondiente en pantalla.

**Resultados Obtenidos:**

1. El campo "Monto recibido" se muestra solo si se elige “Efectivo” como método de pago.
2. El actor puede ingresar el monto sin errores, y al ser mayor al total de la venta, el sistema calcula de inmediato el cambio y lo muestra correctamente en pantalla.

**Evidencia**

A continuación, se presentan las evidencias correspondientes a cada caso de prueba, con el objetivo de confirmar que el sistema responde correctamente frente a las acciones realizadas por el usuario.

1. **Caso de Prueba 1: Visualización del campo "Monto recibido"**

En la *Figura 7.20* se muestra cómo el campo “Monto recibido” aparece únicamente cuando el usuario selecciona “Efectivo” como método de pago. A la izquierda, cuando se elige “Tarjeta”, este campo no se despliega. En cambio, a la derecha, al seleccionar “Efectivo”, el sistema muestra automáticamente tanto el campo “Recibido” como el campo “Vuelto”, lo cual confirma que el comportamiento del sistema es el esperado y funciona correctamente.

*Figura 7.20: "Evidencia Caso de Prueba 1, Caso de Uso N°11”*

Captura de pantalla de un celular

El contenido generado por IA puede ser incorrecto.

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

1. **Caso de Prueba 2: Ingreso del monto entregado**

En la *Figura 7.21* se muestra que el método de pago seleccionado es “Efectivo”, lo que hace que aparezca el campo “Recibido” para ingresar el dinero entregado por el cliente. En este caso, se ingresó un monto de $10.000 para una venta cuyo total es $3.000. De forma inmediata, el sistema calcula y muestra el vuelto correspondiente ($7.000) en el campo “Vuelto”, sin necesidad de recargar la página ni realizar operaciones externas.

*Figura 7.21: "Evidencia Caso de Prueba 2, Caso de Uso N°11”*

Captura de pantalla de un celular

El contenido generado por IA puede ser incorrecto.

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

## Caso de prueba CU20

Este caso de prueba tiene como objetivo verificar que el sistema permita visualizar el stock disponible de cada producto, asegurando que la información mostrada sea clara, precisa y esté actualizada en tiempo real según los movimientos de inventario.

**Caso de Uso N°20:** Visualizando stock disponible

**Casos de Prueba:**

1. Verificar que el actor pueda acceder correctamente al módulo de inventario desde el menú.
2. Verificar que el sistema muestre una tabla con todos los productos y sus cantidades actuales, de forma coherente con la base de datos.

**Resultados Esperados:**

1. El sistema permite ingresar sin errores a la vista de inventario al hacer clic en “Inventario”.
2. Se muestra una tabla con los productos registrados, incluyendo su nombre, código y stock actualizado, el cual debe corresponder con la información registrada en el sistema.

**Resultados Obtenidos:**

1. El actor accede al módulo "Inventario" sin problemas desde el menú.
2. El sistema muestra correctamente la tabla de productos, con los datos actualizados del stock, en coherencia con la información registrada en la base de datos.

**Evidencia**

A continuación, se presentan las evidencias correspondientes a cada caso de prueba, con el objetivo de confirmar que el sistema responde correctamente frente a las acciones realizadas por el usuario.

1. **Caso de Prueba 1: Acceso al módulo de inventario**

En la *Figura 7.22*, se observa que el usuario accede al módulo "Inventario" a través del menú lateral sin que se presenten errores. La interfaz responde correctamente al hacer clic en la opción y muestra la vista correspondiente, lo cual confirma que la navegación funciona como se espera y que no hay problemas en el enlace ni en el despliegue de la vista.

*Figura 7.22: "Evidencia Caso de Prueba 1, Caso de Uso N°20”*

Tabla

El contenido generado por IA puede ser incorrecto.

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

1. **Caso de Prueba 2: Visualización y coherencia del stock**

Tal como se muestra en la *Figura 7.23,* el sistema despliega una tabla con todos los productos disponibles, indicando nombre, código y stock actual. Los valores presentados en pantalla son coherentes con los productos registrados previamente.

*Figura 7.23: “Evidencia Caso de Prueba 2, Caso de Uso N°20”*

Tabla

El contenido generado por IA puede ser incorrecto.

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

## Caso de prueba CU23

Este caso de prueba tiene como objetivo verificar que el sistema permita registrar nuevos productos correctamente, asegurando que toda la información ingresada (como nombre, código, categoría, precio y stock) se guarde y visualice adecuadamente en el sistema.

**Caso de Uso N°23:** Registrando nuevos productos

**Casos de Prueba:**

1. Verificar que el administrador pueda acceder correctamente a la opción "Agregar Producto".
2. Verificar que el sistema despliegue el formulario de registro con todos los campos requeridos.
3. Verificar que se pueda completar y guardar el formulario sin dejar campos vacíos obligatorios.
4. Verificar que el producto se registre correctamente en el sistema y aparezca luego en el inventario.

**Resultados Esperados:**

1. El administrador accede a la vista "Agregar Producto" desde el menú sin errores.
2. El sistema despliega un formulario con campos: nombre, marca, tamaño, precio de compra, precio de venta, código de barras, fecha de vencimiento, proveedor.
3. El sistema valida que todos los campos estén completos antes de guardar.
4. Al guardar, el producto queda registrado y aparece en el módulo de inventario con stock inicial en cero.

**Resultados Obtenidos:**

1. La opción "Agregar Producto" es accesible desde el menú por el usuario administrador.
2. El formulario de registro se muestra completo con todos los campos requeridos.
3. El sistema guarda la información sin errores y valida correctamente los campos obligatorios.
4. El nuevo producto aparece en el inventario tras el registro.

**Evidencia**

A continuación, se presentan las evidencias correspondientes a cada caso de prueba, con el objetivo de confirmar que el sistema responde correctamente frente a las acciones realizadas por el usuario.

1. **Caso de Prueba 1: Acceso al formulario de registro**

En la *Figura 7.24* se puede ver que el administrador accede sin problemas a la opción "Agregar Producto" desde el menú lateral del sistema. Al hacer clic, se despliega correctamente la vista correspondiente, lo que confirma que el acceso al formulario funciona como se espera.

*Figura 7.24: "Evidencia Caso de Prueba 1, Caso de Uso N°23”*

Captura de pantalla de un celular

El contenido generado por IA puede ser incorrecto.

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

1. **Caso de Prueba 2: Visualización del formulario**

En la *Figura 7.25,* se puede ver que el formulario de registro se carga completo, incluyendo todos los campos requeridos como nombre del producto, marca, tamaño, precio de compra y venta, código de barras y proveedor.

*Figura 7.25: "Evidencia Caso de Prueba 2, Caso de Uso N°23”*

Interfaz de usuario gráfica, Aplicación, Correo electrónico

El contenido generado por IA puede ser incorrecto.

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

1. **Caso de Prueba 3: Validación de campos y guardado**

En la *Figura 7.26,* se observa el funcionamiento correcto del formulario para registrar un nuevo producto. Por un lado, se muestra el mensaje de advertencia que aparece cuando un campo requerido está vacío, impidiendo continuar con el registro hasta completar todos los campos obligatorios. Por otro lado, también se visualiza el mensaje de confirmación que entrega el sistema una vez que los datos han sido ingresados correctamente y el producto ha sido guardado sin errores. Esta evidencia confirma que el formulario cuenta con validaciones efectivas que aseguran un ingreso de datos completo y sin omisiones.

*Figura 7.26: "Evidencia Caso de Prueba 3, Caso de Uso N°23”*

Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

1. **Caso de Prueba 4: Visualización en el inventario**

En la *Figura 7.27* se muestra el módulo de inventario luego de registrar un nuevo producto. El nuevo ítem aparece correctamente listado en la tabla, junto con todos los datos que fueron ingresados previamente. Esta evidencia confirma que el producto fue guardado de forma exitosa y que el sistema lo incorpora inmediatamente en la vista del inventario, permitiendo su gestión posterior sin errores.

*Figura 7.27: "Evidencia Caso de Prueba 4, Caso de Uso N°23”*

Tabla

El contenido generado por IA puede ser incorrecto.

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

## Caso de prueba CU24

Este caso de prueba tiene como objetivo verificar que el sistema permita registrar correctamente el ingreso de stock de productos ya existentes, asegurando que la nueva cantidad se sume al stock anterior y se actualice de forma inmediata en el inventario.

**Caso de Uso N°24:** Registrando ingreso de stock de productos existentes

**Casos de Prueba:**

1. Verificar que el administrador pueda acceder a la opción " Añadir Stock”.
2. Verificar que el sistema despliegue el formulario con los campos requeridos: producto, cantidad, fecha de recepción y fecha de vencimiento.
3. Verificar que se pueda completar el formulario correctamente y guardarlo.
4. Verificar que el sistema actualice correctamente el stock del producto correspondiente.

**Resultados Esperados:**

1. El administrador accede correctamente al módulo “Añadir Stock” desde el menú.
2. El sistema despliega un formulario con campos: producto, cantidad, fecha de recepción, fecha de vencimiento.
3. Al completar el formulario con datos válidos, el sistema permite guardar el ingreso.
4. El stock del producto seleccionado se actualiza correctamente, sumando la nueva cantidad ingresada.

**Resultados Obtenidos:**

1. La opción “Añadir Stock” está disponible y accesible desde el menú.
2. El formulario se despliega correctamente con todos los campos necesarios.
3. El sistema valida la información y guarda correctamente el ingreso de stock.
4. El inventario refleja de inmediato la nueva cantidad disponible del producto.

**Evidencia**

A continuación, se presentan las evidencias correspondientes a cada caso de prueba, con el objetivo de confirmar que el sistema responde correctamente frente a las acciones realizadas por el usuario.

1. **Evidencia Caso de Prueba 1: Acceso al módulo “Ingresar Stock”**

En la *Figura 7.28,* se muestra que el administrador accede sin dificultades a la opción “Añadir Stock” desde el menú lateral del sistema. Esta acción carga correctamente la vista correspondiente, confirmando que el acceso funciona como se espera.

*Figura 7.28: "Evidencia Caso de Prueba 1, Caso de Uso N°24”*

Tabla

El contenido generado por IA puede ser incorrecto.

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

1. **Evidencia Caso de Prueba 2: Visualización del formulario**

Como se observa en la *Figura 7.29,* el sistema despliega un formulario que incluye todos los campos requeridos: selección del producto, cantidad a ingresar, fecha de recepción y fecha de vencimiento.

*Figura 7.29: "Evidencia Caso de Prueba 2, Caso de Uso N°24”*

Interfaz de usuario gráfica, Texto, Aplicación

El contenido generado por IA puede ser incorrecto.

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

1. **Evidencia Caso de Prueba 3: Guardado del ingreso de stock**

En la *Figura 7.30,* se muestra el formulario de registro de usuario con todos los campos obligatorios correctamente completados. Una vez ingresados todos los datos válidos, el sistema permite guardar el registro sin mostrar errores. En caso de que falte algún campo obligatorio, la validación impide continuar, asegurando que la información quede registrada de manera completa y correcta.

*Figura 7.30: "Evidencia Caso de Prueba 3, Caso de Uso N°24”*

Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

1. **Evidencia Caso de Prueba 4: Actualización del inventario**

En la *Figura 7.31,* se puede observar cómo el stock del producto “Pisco” se actualiza correctamente tras registrar el ingreso de nuevas unidades. A la izquierda se muestra el valor inicial de 33 unidades, y a la derecha, luego del ingreso, el sistema refleja un total de 93 unidades.

*Figura 7.31: “Evidencia Caso de Prueba 4, Caso de Uso N°24”*

Tabla

El contenido generado por IA puede ser incorrecto.

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

## Caso de prueba CU40

Este caso de prueba tiene como objetivo verificar que el sistema permita crear nuevos usuarios correctamente, asegurando que los datos ingresados se almacenen de forma adecuada y que el nuevo usuario pueda acceder al sistema según el rol asignado.

**Caso de Uso N°40:** Creando Usuarios

**Casos de Prueba:**

1. Verificar que el administrador pueda acceder correctamente a la opción “Administrar Usuarios”.
2. Verificar que el sistema despliegue el formulario con los campos requeridos: nombre, nombre de usuario, contraseña y rol.
3. Verificar que el formulario valide correctamente los datos ingresados.
4. Verificar que el sistema registre correctamente al nuevo usuario y muestre un mensaje de éxito.
5. Verificar que el nuevo usuario pueda iniciar sesión con las credenciales registradas.

**Resultados Esperados:**

1. El sistema permite el acceso a la opción “Crear nuevo usuario” desde el menú de administración.
2. Se muestra un formulario con todos los campos requeridos para el registro de un nuevo usuario.
3. El sistema valida que no se dejen campos vacíos, y que el nombre de usuario no esté repetido.
4. Tras guardar, el sistema muestra un mensaje de confirmación y guarda los datos correctamente.
5. El nuevo usuario puede iniciar sesión usando su nombre de usuario y contraseña, accediendo según el rol asignado.

**Resultados Obtenidos:**

1. El administrador accede sin problemas a la vista de creación de usuarios.
2. El formulario se muestra completo con todos los campos necesarios.
3. El sistema detecta datos vacíos o duplicados e impide guardar hasta corregirlos.
4. El nuevo usuario se registra correctamente y aparece en el listado de usuarios.
5. El usuario puede iniciar sesión sin problemas y tiene acceso según su rol.

**Evidencia**

A continuación, se presentan las evidencias correspondientes a cada caso de prueba, con el objetivo de confirmar que el sistema responde correctamente frente a las acciones realizadas por el usuario.

1. **Evidencia Caso de Prueba 1: Acceso a la opción “Administrar Usuarios”**

En la *Figura 7.32,* se muestra que el administrador accede correctamente al menú “Administrar Usuarios”. Al hacer clic en esta opción, el sistema redirige sin errores a la vista de gestión, donde se pueden crear nuevos usuarios.

*Figura 7.32: "Evidencia Caso de Prueba 1, Caso de Uso N°40”*

Interfaz de usuario gráfica

El contenido generado por IA puede ser incorrecto.

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

1. **Evidencia Caso de Prueba 2: Visualización del formulario**

Como se observa en la *Figura 7.33,* el sistema despliega un formulario completo con los campos requeridos: nombre, nombre de usuario, contraseña y rol. Esto permite al administrador ingresar toda la información necesaria para crear una cuenta.

*Figura 7.33: "Evidencia Caso de Prueba 2, Caso de Uso N°40”*

Interfaz de usuario gráfica, Texto, Aplicación

El contenido generado por IA puede ser incorrecto.

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

1. **Evidencia Caso de Prueba 3: Validación de datos**

En la *Figura 7.34* se puede ver que el sistema no permite guardar el formulario si hay campos obligatorios vacíos o si el nombre de usuario ya está siendo utilizado. En ambos casos se despliega un mensaje de advertencia que indica claramente qué dato debe completarse o corregirse. Esta funcionalidad asegura que los datos ingresados sean válidos antes de guardar, evitando errores en el registro de usuarios.

*Figura 7.34: "Evidencia Caso de Prueba 3, Caso de Uso N°40”*

Interfaz de usuario gráfica, Texto, Aplicación

El contenido generado por IA puede ser incorrecto.

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

1. **Evidencia Caso de Prueba 4: Registro exitoso**

Luego de completar el formulario con datos válidos, como se muestra en la *Figura 7.35,* el sistema guarda correctamente al nuevo usuario y despliega un mensaje de éxito confirmando que el registro fue realizado. A continuación, el usuario recién creado aparece en la tabla de usuarios junto con su nombre, nombre de usuario y rol asignado lo que confirma que la información fue almacenada y mostrada sin errores.

*Figura 7.35: "Evidencia Caso de Prueba 4, Caso de Uso N°40”*

Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

1. **Evidencia Caso de Prueba 5: Inicio de sesión del nuevo usuario**

En la *Figura 7.36*, se observa que el nuevo usuario accede al sistema ingresando su nombre de usuario y contraseña. El sistema lo deja ingresar sin errores y lo redirige al panel de inicio, indicándole su rol y lo que puede realizar en el sistema confirmando que el proceso fue exitoso.

*Figura 7.36:"Evidencia Caso de Prueba 5, Caso de Uso N°40”*

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

El contenido generado por IA puede ser incorrecto.

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

## Caso de prueba CU41

Este caso de prueba tiene como objetivo verificar que el sistema permita modificar los datos de un usuario existente, asegurando que los cambios realizados (como nombre, correo o rol) se actualicen correctamente y se reflejen en el sistema sin errores.

**Caso de Uso N°41:** Modificando Usuarios

**Casos de Prueba:**

1. Verificar que al hacer clic en “Editar”, el sistema muestre un formulario con los datos actuales del usuario.
2. Verificar que el administrador pueda modificar el nombre o rol.
3. Verificar que los cambios se guarden correctamente y se muestre un mensaje de confirmación.

**Resultados Esperados:**

1. Al hacer clic en “Editar”, el sistema carga un formulario con los datos actuales del usuario seleccionado.
2. El sistema permite modificar los campos editables (nombre, rol) sin errores de validación.
3. Al guardar los cambios, el sistema actualiza la información del usuario y muestra un mensaje de éxito.

**Resultados Obtenidos:**

1. El formulario de edición se carga correctamente con la información actual del usuario seleccionado.
2. El administrador puede modificar los datos y guardarlos sin errores.
3. El sistema guarda los cambios correctamente y muestra un mensaje confirmando la actualización.

**Evidencia**

A continuación, se presentan las evidencias correspondientes a cada caso de prueba, con el objetivo de confirmar que el sistema responde correctamente frente a las acciones realizadas por el usuario.

1. **Evidencia Caso de Prueba 1: Formulario de edición**

Al hacer clic en el botón “Editar”, el sistema carga un formulario con los datos actuales del usuario, como se ve en la *Figura 7.37*. Es importante destacar que el campo “Nombre de usuario” está bloqueado y no puede ser modificado, lo cual es correcto según las reglas que se decidieron implementar en el sistema.

*Figura 7.37: "Evidencia Caso de Prueba 1, Caso de Uso N°41”*

*Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

El contenido generado por IA puede ser incorrecto.*

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

1. **Evidencia caso de prueba 2: Modificación de nombre o rol**

En la *Figura 7.38*, se muestra cómo el sistema permite editar correctamente los datos de un usuario. En este caso, se modificó el nombre completo y se mantuvo el mismo rol. Los campos del formulario funcionan sin errores, permitiendo aplicar los cambios necesarios. Una vez realizada la edición, el sistema confirma la operación mostrando un mensaje de éxito, lo que indica que los datos fueron actualizados correctamente.

*Figura 7.38: "Evidencia Caso de Prueba 2, Caso de Uso N°41”*

Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

1. **Evidencia Caso de Prueba 3: Guardado y mensaje de confirmación**

Luego de realizar los cambios y presionar el botón “Guardar”, el sistema actualiza los datos del usuario correctamente. En la *Figura 7.39* se observa que el nombre modificado ya aparece reflejado en la tabla de usuarios, junto con el nombre de usuario y el rol asignado.

*Figura 7.39: "Evidencia Caso de Prueba 3, Caso de Uso N°41”*

Tabla

El contenido generado por IA puede ser incorrecto.

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

## Caso de prueba CU42

Este caso de prueba tiene como objetivo verificar que el sistema permita eliminar usuarios de forma segura, asegurando que estos ya no puedan acceder al sistema y que su información se retire o desactive correctamente según las reglas establecidas.

**Caso de Uso N°42:** Eliminando Usuario

**Casos de Prueba:**

1. Verificar que al hacer clic en “Eliminar”, el sistema solicite confirmación antes de proceder.
2. Verificar que, tras la confirmación, el sistema elimine correctamente al usuario.
3. Verificar que el usuario eliminado ya no pueda iniciar sesión.

**Resultados Esperados:**

1. El sistema solicita una confirmación antes de eliminar el usuario seleccionado.
2. Una vez confirmada la eliminación, el sistema borra al usuario y muestra un mensaje de éxito.
3. El usuario eliminado ya no puede acceder al sistema con sus credenciales.

**Resultados Obtenidos:**

1. El sistema despliega una ventana de confirmación al intentar eliminar un usuario.
2. Al confirmar, el usuario se elimina correctamente y desaparece de la lista.
3. Al intentar iniciar sesión con el usuario eliminado, el sistema muestra error de acceso.

**Evidencia**

A continuación, se presentan las evidencias correspondientes a cada caso de prueba, con el objetivo de confirmar que el sistema responde correctamente frente a las acciones realizadas por el usuario.

1. **Caso de Prueba 1: Solicitud de confirmación antes de eliminar**

Como se aprecia en la *Figura 7.40*, al hacer clic en “Eliminar”, el sistema no elimina de inmediato al usuario, sino que muestra una ventana emergente solicitando confirmación.

*Figura 7.40: "Evidencia Caso de Prueba 1, Caso de Uso N°42”*

Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto

El contenido generado por IA puede ser incorrecto.

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

1. **Caso de Prueba 2: Eliminación correcta del usuario**

Tal como se observa en la *Figura 7.41,* luego de confirmar la acción de eliminar, el sistema muestra un mensaje que indica que el usuario ha sido borrado correctamente, donde también el usuario eliminado desaparece de inmediato del listado, lo que confirma que la operación fue exitosa y que la interfaz se actualiza automáticamente para reflejar los cambios.

*Figura 7.41: "Evidencia Caso de Prueba 2, Caso de Uso N°42”*

Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

1. **Caso de Prueba 3: Usuario no puede iniciar sesión**

Finalmente, al intentar ingresar al sistema usando las credenciales del usuario eliminado, el sistema rechaza el acceso y muestra un mensaje de error, como se evidencia en la *Figura 7.42*. Esto confirma que el usuario fue completamente removido del sistema.

*Figura 7.42: "Evidencia Caso de Prueba 3, Caso de Uso N°42”*

Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto

El contenido generado por IA puede ser incorrecto.

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

## Caso de prueba CU43

Este caso de prueba tiene como objetivo verificar que el sistema permita cambiar el rol asignado a un usuario existente, asegurando que el nuevo rol se aplique correctamente y que el usuario acceda únicamente a las funcionalidades correspondientes a dicho rol.

**Caso de Uso N°43:** Cambiando rol de usuario

**Casos de Prueba:**

1. Verificar que el sistema muestre el rol actual y permita cambiarlo por otro.
2. Verificar que el sistema guarde correctamente el nuevo rol y actualice los permisos.
3. Verificar que, al iniciar sesión nuevamente, el usuario vea los módulos correspondientes a su nuevo rol.

**Resultados Esperados:**

1. El administrador puede cambiar el rol a una opción distinta (por ejemplo, de “Vendedor” a “Administrador”).
2. El sistema guarda el nuevo rol y actualiza internamente los permisos asignados al usuario.
3. Al iniciar sesión, el usuario ve su nueva interfaz y funcionalidades según el nuevo rol asignado.

**Resultados Obtenidos:**

1. El administrador modifica el rol sin problemas y el sistema permite guardar los cambios.
2. Se confirma visualmente que el cambio fue exitoso y el nuevo rol está asignado.
3. En la siguiente sesión del usuario, los accesos reflejan el nuevo perfil correctamente.

**Evidencias**

A continuación, se presentan las evidencias correspondientes a cada caso de prueba, con el objetivo de confirmar que el sistema responde correctamente frente a las acciones realizadas por el usuario.

1. **Evidencia Caso de Prueba 1: Cambio del rol actual por otro**

Como se observa en la *Figura 7.43*, el campo de rol es editable, y el administrador puede seleccionar una opción distinta desde el menú desplegable.

*Figura 7.43: "Evidencia Caso de Prueba 1, Caso de Uso N°43”*

Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

1. **Evidencia Caso de Prueba 2: Guardado exitoso del nuevo rol**

Una vez realizado el cambio, el sistema guarda correctamente la nueva configuración. En la *Figura 7.44,* se muestra el mensaje de confirmación que indica que los cambios se han guardado satisfactoriamente.

*Figura 7.44: "Evidencia Caso de Prueba 2, Caso de Uso N°43”*

Tabla

El contenido generado por IA puede ser incorrecto.

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

1. **Evidencia Caso de Prueba 3: Visualización de módulos según el nuevo rol**

Al iniciar sesión nuevamente, el usuario ve una interfaz distinta según su nuevo perfil. En la *Figura 7.45,* se evidencia que los módulos disponibles corresponden al nuevo rol asignado, confirmando que el cambio surtió efecto correctamente.

*Figura 7.45: "Evidencia Caso de Prueba 3, Caso de Uso N°43”*

Interfaz de usuario gráfica, Texto, Aplicación

El contenido generado por IA puede ser incorrecto.

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

## Caso de prueba CU45

Este caso de prueba tiene como objetivo verificar que el sistema permita iniciar sesión correctamente, asegurando que las credenciales sean validadas y que el usuario acceda al panel correspondiente según su rol asignado.

**Caso de Uso N°45:** Iniciar sesión en el sistema

**Casos de Prueba:**

1. Verificar que el formulario de inicio de sesión se muestre correctamente al acceder al sistema.
2. Verificar que el usuario pueda ingresar su nombre de usuario y contraseña.
3. Verificar que el sistema permita el acceso al panel correcto si las credenciales son válidas.
4. Verificar que el sistema muestre un mensaje de error si las credenciales son incorrectas.
5. Verificar que el usuario visualice únicamente los módulos y funcionalidades correspondientes a su rol (Administrador o Vendedor) una vez iniciada la sesión.

**Resultados Esperados:**

1. El sistema carga y muestra correctamente la página de inicio de sesión con su formulario.
2. El formulario acepta el ingreso de nombre de usuario y contraseña.
3. Si las credenciales son correctas, el usuario accede al sistema.
4. Si las credenciales son incorrectas, se muestra un mensaje claro de error sin acceder al sistema.
5. El sistema permite al usuario autenticado visualizar únicamente los módulos y funcionalidades correspondientes a su rol (Administrador o Vendedor).

**Resultados Obtenidos:**

1. El formulario de inicio de sesión se muestra sin errores al abrir la página.
2. El usuario puede ingresar sus datos sin restricciones.
3. El sistema valida correctamente y e inicia al sistema sin problemas.
4. Las credenciales válidas permiten acceso al sistema sin errores.
5. Una vez autenticado, el usuario visualiza correctamente solo los módulos que le corresponden según su rol asignado en el sistema.

**Evidencia**

A continuación, se presentan las evidencias correspondientes a cada caso de prueba, con el objetivo de confirmar que el sistema responde correctamente frente a las acciones realizadas por el usuario.

1. **Caso de Prueba 1: Verificar que el formulario de inicio de sesión se muestre correctamente al acceder al sistema.**

En la *Figura 7.46* , se muestra el formulario de inicio de sesión que aparece al acceder al sistema. Esta vista se carga sin errores, mostrando los campos necesarios para ingresar el nombre de usuario y la contraseña, junto con el botón de acceso.

*Figura 7.46: "Evidencia Caso de Prueba 1, Caso de Uso N°45”*

**Imagen que contiene interior, frente, sostener, tabla

El contenido generado por IA puede ser incorrecto.**

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

1. **Caso de Prueba 2: Verificar que el usuario pueda ingresar su nombre de usuario y contraseña.**

En la *Figura 7.47*, se observa cómo el usuario puede escribir sus datos de acceso sin problemas. Los campos responden bien, permiten completar la información y no presentan validaciones erróneas ni bloqueos inesperados.

*Figura 7.47: "Evidencia Caso de Prueba 2, Caso de Uso N°45”*

Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto

El contenido generado por IA puede ser incorrecto.

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

1. **Caso de Prueba 3: Verificar que el sistema valide correctamente las credenciales ingresadas.**

En la *Figura 7.48*, se muestran dos momentos clave del proceso de autenticación. En la imagen izquierda, el usuario ingresa su nombre de usuario y contraseña en el formulario de inicio de sesión. Una vez que las credenciales son validadas correctamente, el sistema redirige al usuario automáticamente al panel principal. Esto se aprecia en la imagen derecha, donde se despliega la pantalla de bienvenida junto con las funcionalidades habilitadas según el rol.

*Figura 7.48: "Evidencia Caso de Prueba 3, Caso de Uso N°45”*

Interfaz de usuario gráfica, Texto, Aplicación

El contenido generado por IA puede ser incorrecto.

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

1. **Caso de Prueba 4: Verificar que el sistema muestre un mensaje de error si las credenciales son incorrectas.**

En la *Figura 7.49* se observa el comportamiento del sistema cuando se ingresan credenciales erróneas. En este caso, el sistema despliega un mensaje de error indicando que los datos no son válidos y no permite continuar hasta que la información sea corregida.

*Figura 7.49: "Evidencia Caso de Prueba 4, Caso de Uso N°45”*

**Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto

El contenido generado por IA puede ser incorrecto.**

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

# REUNIONES SCRUM Y GESTIÓN DEL AVANCE DEL PRIMER INCREMENTO

En esta sección se describe cómo el equipo organizó su trabajo durante el primer incremento utilizando los principios de Scrum. Se incluyen las reuniones diarias, la revisión del Sprint, la retrospectiva y los ajustes al Product Backlog, todos elementos clave para asegurar un desarrollo ágil, organizado y enfocado.

## Reuniones diarias (Daily Scrum)

Durante el desarrollo del primer incremento, el formato tradicional del Daily Scrum fue adaptado debido a la carga académica y a la disponibilidad horaria. En lugar de realizar reuniones diarias, se estableció una reunión fija cada lunes, posterior al encuentro con el profesor. En dichas instancias se revisaban los avances, se identificaban posibles ajustes y se organizaban las tareas correspondientes a la semana.

De manera complementaria, cuando surgían dudas o se requería revisar el progreso con mayor detalle, se programaban reuniones extraordinarias de acuerdo con la disponibilidad de los integrantes. Esta modalidad garantizó una comunicación constante, facilitó la toma de decisiones y aseguró la continuidad del trabajo durante el Sprint.

## Revisión del Sprint (Sprint Review)

Al finalizar el primer incremento, el equipo sostuvo una reunión en línea con el administrador de la Botillería Joaquín, con el propósito de mostrar el estado del sistema y verificar que las funciones implementadas hasta ese momento respondieran a sus necesidades.

Durante la reunión, se presentó el sistema en funcionamiento en tiempo real, evidenciando paso a paso cómo se ejecutaban acciones clave, tales como registrar una venta, revisar el stock, agregar productos nuevos y generar comprobantes. El cliente tuvo la oportunidad de probar la plataforma directamente y entregar su opinión basada en su experiencia diaria en la botillería. Esta instancia permitió identificar algunos ajustes menores que no habían sido considerados inicialmente.

## Retrospectiva del Sprint (Sprint Retrospective)

Después de la Revisión del Sprint con el cliente, se llevó a cabo una reunión interna con el equipo con el fin de reflexionar sobre el desarrollo del trabajo durante el primer incremento.

Uno de los aspectos positivos más destacados fue la buena comunicación entre los integrantes del equipo. Aunque no fue posible reunirse diariamente, se mantuvo una organización adecuada que permitió avanzar en las tareas, dar seguimiento al progreso general y brindar apoyo mutuo frente a dudas o complicaciones. Esta coordinación favoreció que todas las tareas propuestas fueran completadas antes de la entrega final.

Como aspecto de mejora para los próximos sprints, se definió la realización de reuniones diarias siempre que las condiciones lo permitan, lo que facilitaría un control más riguroso del avance y la resolución oportuna de problemas. Asimismo, se recomendó incrementar la frecuencia de reuniones con el cliente durante el desarrollo, con el propósito de recibir retroalimentación constante y garantizar que el software responda a sus expectativas y necesidades.

## Finalización del Product Backlog

Como resultado de las reuniones realizadas, el Product Backlog del primer incremento fue revisado y actualizado. Se completó el 100% de las actividades planificadas para este Sprint, lo que refleja una correcta organización, compromiso y eficiencia en la gestión del trabajo.

Cada tarea fue asignada con un tiempo estimado y una fecha límite, y la mayoría de ellas se completó dentro del rango de fechas planificadas, sin necesidad de extender el plazo general del Sprint. Esta condición permitió mantener el control del avance y cumplir con los objetivos definidos desde el inicio del incremento.

En relación con el uso del tiempo, las horas hombre disponibles fueron distribuidas de manera adecuada, lo que permitió equilibrar la carga de trabajo entre los distintos integrantes. La asignación de tareas se realizó considerando las fortalezas de cada persona del equipo, lo que favoreció un avance fluido, sin bloqueos ni retrasos significativos.

Como resultado de esta organización, todas las funcionalidades planificadas en este primer ciclo de desarrollo fueron implementadas, probadas y validadas, quedando disponibles como parte de la versión funcional del sistema.

La Figura 8.1 presenta el detalle completo del Product Backlog, incluyendo la prioridad de cada tarea, su caso de uso asociado, responsables, esfuerzo estimado, esfuerzo real, fechas planificadas, fechas reales y su estado actual, el cual corresponde a “finalizado” en todos los casos.

*Figura 8.1: Product backlog con tareas finalizadas*

*Tabla

El contenido generado por IA puede ser incorrecto.*

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

Para una visualización más clara y detallada, se recomienda revisar directamente el archivo “Ingeniería de Software – Primer Incremento.xlsx”, específicamente en la pestaña Planificación 2.

# TABLA DE ESFUERZO PRIMER INCREMENTO

En esta sección se presenta la tabla de esfuerzo, creada a partir del Product Backlog del primer incremento, donde todas las tareas planificadas fueron completadas con éxito. El objetivo de esta tabla es mostrar de forma clara la distribución del tiempo real de trabajo, indicando cuántas horas fueron destinadas a cada tarea y en qué días del Sprint se registró actividad sobre ellas.

Cada fila de la tabla representa una tarea, y cada columna corresponde a un día del Sprint. Las celdas muestran las horas trabajadas en cada tarea en un día específico. Este formato permite identificar qué tareas requirieron mayor dedicación, cuáles se ejecutaron en paralelo y en qué fechas se concentró una mayor carga de trabajo

La *Figura 9.1* muestra de forma clara y ordenada esta distribución diaria de esfuerzo, funcionando como una herramienta clave para:

* Verificar si las tareas se completaron en los plazos esperados.
* Comparar el esfuerzo estimado con el esfuerzo real invertido.
* Detectar periodos de mayor exigencia o acumulación de tareas.
* Identificar oportunidades de mejora para la planificación de futuros sprints.

*Figura 9.1: Tabla de Esfuerzo*

Imagen en blanco y negro

El contenido generado por IA puede ser incorrecto.

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

Para una visualización más clara y detallada, se recomienda revisar directamente el archivo “Ingeniería de Software – Primer Incremento.xlsx”, específicamente en la pestaña Tabla de Esfuerzo.

## Burn up y burn down horas hombre

A partir de los datos obtenidos en la tabla de esfuerzo correspondiente al Primer Incremento, se elaboraron dos gráficos fundamentales para el análisis del rendimiento del equipo: el gráfico Burn Up y el gráfico Burn Down. Ambos permiten visualizar de manera clara el progreso del equipo en relación con las estimaciones iniciales.

La *Figura 9.2* presenta el gráfico Burn Up, el cual muestra el avance acumulado del trabajo medido en horas hombre (HH) a lo largo de los días del Sprint.

Este gráfico incorpora dos líneas:

* La línea gris representa el avance estimado acumulado, correspondiente a la cantidad de horas que se proyectaba haber completado en cada día según la planificación inicial.
* La línea azul muestra el avance real acumulado, que refleja las horas efectivamente trabajadas y registradas por el equipo.

La comparación entre ambas líneas permite observar si el equipo avanzó según lo planificado, si existieron días con mayor o menor carga de trabajo y si se alcanzó la meta final. En este caso, ambas líneas se mantuvieron próximas durante todo el Sprint y al final convergieron cerca de las 220 horas acumuladas, lo que evidencia que el equipo completó todo el trabajo planificado, sin desviaciones significativas.

*Figura 9.2: Burn Up Incremento 1 (HH)*

Gráfico, Gráfico de líneas

El contenido generado por IA puede ser incorrecto.

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

La *Figura 9.3* muestra el gráfico Burn Down, que representa cómo fue bajando la cantidad de trabajo pendiente cada día durante el Sprint.

Al igual que en el gráfico Burn Up, se comparan dos líneas:

* La línea gris muestra cuántas horas se esperaba que quedaran pendientes cada día, según la planificación.
* La línea azul muestra cuántas horas realmente quedaban por hacer al final de cada jornada, según lo que el equipo fue trabajando.

Este gráfico permite identificar si el equipo avanzó de forma constante o si en algún momento existieron estancamientos. En este caso, la línea azul descendió de manera uniforme, lo que indica que el equipo mantuvo un ritmo de trabajo estable y no acumuló tareas hacia el final del Sprint.

*Figura 9.3: Burn Down Incremento 1 (HH)*

Gráfico, Gráfico de líneas

El contenido generado por IA puede ser incorrecto.

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

## Burn up y burn down tareas

A partir de los datos registrados en la tabla de esfuerzo, se generaron los gráficos Burn Up y Burn Down correspondientes al primer incremento. A diferencia de los gráficos basados en horas hombre (HH), estos muestran el avance y la reducción de tareas pendientes durante el Sprint, lo que permite evidenciar con mayor claridad el cumplimiento de los entregables desde el punto de vista funcional.

La *Figura 9.4* presenta el gráfico Burn Up, que representa cómo se acumularon las tareas completadas día a día durante el Sprint. En este gráfico:

* El eje horizontal (X) muestra los días, desde que comenzó el Sprint hasta que terminó.
* El eje vertical (Y) indica cuántas tareas se habían completado hasta ese momento.

El gráfico evidencia que al inicio del Sprint el avance fue menor, debido a que los primeros días se destinaron al análisis y la planificación. Posteriormente, a partir del 17 de mayo, comenzó un aumento progresivo en la cantidad de tareas finalizadas. El crecimiento se mantuvo constante y más intenso entre el 22 de mayo y el 4 de junio. Finalmente, el 15 de junio se completaron todas las tareas planificadas.

*Figura 9.4: Progreso del Esfuerzo (Burn Up)*

Gráfico

El contenido generado por IA puede ser incorrecto.

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

La *Figura 9.5* presenta el gráfico Burn Down, que representa la disminución diaria de las tareas pendientes durante el Sprint.

* El eje horizontal (X) muestra los días, desde que comenzó hasta que terminó el incremento.
* El eje vertical (Y) indica cuántas tareas estaban pendientes.

El gráfico inicia con cerca de 50 tareas planificadas y muestra una reducción constante a lo largo del tiempo, con una disminución más notoria entre fines de mayo y principios de junio. Para el 10 de junio, la línea se aproximó a cero, lo que refleja que prácticamente todo el trabajo se había completado antes de la fecha final del Sprint.

*Figura 9.5: Esfuerzo Pendiente (Burn Down)*

Gráfico

El contenido generado por IA puede ser incorrecto.

Fuente: Elaborado por los estudiantes de acuerdo con el proyecto

# BIBLIOGRAFIA

Kruchten, P. (1995). *The 4+1 View Model of Architecture. IEEE Software*, 12(6), 42–50. https://www.cs.ubc.ca/~gregor/teaching/papers/4+1view-architecture.pdf

InboundCycle. (2014). *El árbol web: qué es y para qué sirve*. InboundCycle. https://www.inboundcycle.com/blog-de-inbound-marketing/bid/195257/el-rbol-web-qu-es-y-para-qu-sirve

Lucidchart. (s.f.). *Tutorial de diagrama de clases UML.* Recuperado el 15 de junio de 2025, de <https://www.lucidchart.com/pages/es/tutorial-de-diagrama-de-clases-uml>

Lucidchart. (s.f.). *Diagrama de componentes UML*. Lucidchart. Recuperado el 15 de junio de 2025, de <https://www.lucidchart.com/pages/uml-component-diagram>

Lucidchart. (s.f.). *¿Qué es un modelo de base de datos?* Lucidchart. Recuperado el 15 de junio de 2025, de <https://www.lucidchart.com/pages/es/que-es-un-modelo-de-base-de-datos>

Lucidchart. (s.f.). *Tutorial de diagramas de despliegue*. Lucidchart. Recuperado el 15 de junio de 2025, de <https://www.lucidchart.com/pages/es/tutorial-de-diagramas-de-despliegue>

Lucidchart. (s.f.). *Diagrama de secuencia UML: qué es y cómo hacerlo.* <https://www.lucidchart.com/pages/es/diagrama-de-secuencia>

# ANEXOS

### **Anexo A: Detalle de vistas del sistema**

Este anexo presenta una descripción general del funcionamiento de las vistas principales del sistema de ventas e inventario desarrollado para la Botillería Joaquín, donde cada vista corresponde a una pantalla del sistema desde la cual el usuario puede realizar acciones específicas. Las vistas aquí descritas son las mismas que se representan en el árbol de navegación, lo que permite comprender cómo están conectadas entre sí, y cómo se estructura el recorrido que realiza el usuario dentro del sistema.

**Vista: Login (V\_Login)**

La vista *Login* es la pantalla de acceso al sistema, y está disponible para todos los perfiles de usuario, cumpliendo la función principal de autenticar al usuario antes de permitirle ingresar a las funcionalidades del sistema. En esta vista, se debe ingresar un nombre de usuario y una contraseña válidos, los cuales son verificados en la base de datos.

Si las credenciales ingresadas son correctas, el sistema redirige al usuario a la vista *Inicio*, cargando automáticamente las funciones correspondientes según su perfil, ya sea Administrador o Vendedor, mientras que, si los datos son incorrectos, se muestra un mensaje de error indicando que las credenciales no son válidas.

**Vista: Inicio (V\_Inicio)**

La vista Inicio corresponde a la primera pantalla que se presenta al usuario luego de iniciar sesión en el sistema, y aunque no contiene funcionalidades operativas propias, cumple un rol fundamental como punto de acceso a todas las demás secciones mediante el menú lateral. Desde aquí es posible navegar hacia funcionalidades principales como registrar ventas, revisar el inventario, administrar usuarios o añadir productos, todo dentro de un diseño simple y funcional que permite orientar al usuario rápidamente y facilita la interacción con el resto del sistema.

Además, el contenido de esta vista varía según el perfil del usuario que haya iniciado sesión. En el caso del Administrador, se muestra un mensaje de bienvenida personalizado, acompañado de un resumen de las funciones disponibles y una lista de recomendaciones para mejorar la gestión del sistema, como revisar el stock crítico o supervisar los accesos. Para el perfil de Vendedor, esta vista también incluye un mensaje de bienvenida y un listado adaptado de funcionalidades disponibles, enfocadas en la realización de ventas y la consulta de inventario.

Aunque no se ejecutan acciones directamente desde esta pantalla, su objetivo principal es entregar una visión general del sistema y guiar al usuario según su rol dentro de la plataforma.

**Vista: Crear Venta (V\_CrearVenta)**

La vista *Crear Venta* está disponible tanto para el perfil de Administrador como para el de Vendedor, ya que corresponde a una de las funciones principales del sistema. Desde esta pantalla, el usuario puede registrar una nueva venta de manera completa, permitiendo agregar productos al carrito, modificar cantidades, eliminar productos si es necesario y seleccionar el método de pago, ya sea en efectivo o con tarjeta. A medida que se agregan productos, el sistema actualiza automáticamente el total de la venta, facilitando el seguimiento del monto final.

Una vez finalizado el proceso, el usuario puede guardar la venta, lo que registra la operación en la base de datos y descuenta automáticamente el stock de los productos involucrados.

**Vista: Lista de Ventas (V\_ListaVentas)**

La vista *Lista de Ventas* está disponible tanto para el perfil de Administrador como para el de Vendedor, ya que permite consultar el historial de ventas realizadas dentro del sistema, mostrando una tabla con información clave de cada transacción, como la fecha, el total de la venta y el nombre del usuario que la registró.

Además, permite acceder al detalle de cada venta, donde se visualizan los productos vendidos, las cantidades y el método de pago utilizado, lo que resulta especialmente útil para hacer seguimiento a las operaciones realizadas y verificar que el proceso de ventas se haya ejecutado correctamente.

**Vista: Inventario (V\_Inventario)**

La vista *Inventario* está disponible tanto para el perfil de Administrador como para el de Vendedor, y entrega al usuario una visión clara y actualizada del estado de los productos disponibles en la botillería. A través de una tabla, se puede consultar información relevante como el nombre del producto, su código, el stock disponible, la fecha de vencimiento y el proveedor, lo que facilita una revisión rápida y completa del inventario.

Esta funcionalidad cumple un rol clave en la toma de decisiones relacionadas con la reposición de productos, el control de vencimientos y la gestión general del inventario, ya que permite detectar con facilidad si algún producto está agotado o por debajo del stock mínimo deseado.

**Vista: Crear Productos (V\_CrearProductos)**

La vista *Crear Productos* está disponible únicamente para el perfil de **Administrador**, y está pensada para registrar nuevos productos dentro del sistema, permitiendo ingresar datos esenciales como el nombre del producto, su tamaño, precio de compra, precio de venta, código de barras, proveedor y fecha de vencimiento estimada.

Además, desde esta misma vista es posible eliminar productos que ya no se comercialicen, lo que permite mantener el catálogo actualizado y ordenado, facilitando tanto las tareas de venta como el control del inventario.

**Vista: Añadir Stock (V\_AñadirStock)**

La vista *Añadir Stock* es exclusiva del perfil de Administrador, y permite aumentar la cantidad disponible de un producto que ya se encuentra registrado en el sistema. El proceso consiste en seleccionar el producto desde una lista, ingresar la cantidad recibida y, registrar la nueva fecha de vencimiento, lo que asegura que el ingreso de mercadería sea ordenado y trazable.

Esta funcionalidad es esencial para mantener la información del inventario actualizada, y permite registrar de forma separada cada ingreso, lo que facilita la trazabilidad de los productos y el control por lote o por fecha, mejorando así la gestión general del stock.

**Vista: Administrar Usuarios (V\_AdministrarUsuarios)**

La vista *Administrar Usuarios* está disponible únicamente para el perfil de Administrador, y permite gestionar los perfiles de acceso al sistema de forma centralizada. Desde esta sección es posible crear nuevos usuarios, editar los datos de los existentes (como el nombre, la contraseña o el rol asignado), eliminar usuarios que ya no forman parte del equipo y consultar la lista completa de usuarios registrados.

Esta funcionalidad asegura que cada persona tenga acceso únicamente a las funciones que le corresponden según su rol (por ejemplo, administrador o vendedor), y permite mantener un control claro sobre la seguridad y la trazabilidad de las acciones realizadas dentro del sistema.