Imagen que contiene señal, firmar, parada, tráfico

Descripción generada automáticamente

**INSTITUTO POLITÉCTICO NACIONAL**

**ESCUELA SUPERIOR DE CÓMPUTO**

**ESCOM**

TRABAJO TERMINAL

**“Aplicación Web para la gestión de almacén e inventario con análisis y visualización de datos para tiendas de conveniencia.”**

**2024 – A133**

Presentan:

**Garzón Domínguez Gerardo Ismael**

**Muñoz Ramírez Gabriel de Jesús**

**Sánchez Becerra Ernesto Daniel**

Directores:

**M. en C. Alejandro Botello Castillo M. en C. Nancy Ocotitla Rojas**



Imagen que contiene señal, firmar, parada, tráfico

Descripción generada automáticamente

**INSTITUTO POLITÉCTICO NACIONAL**

**ESCUELA SUPERIOR DE CÓMPUTO**

**SUBDIRECCIÓN ACADÉMICA**

No. de TT: 2024-A13 Fecha de presentación: 13/diciembre/2023

*Documento Técnico*

**“Aplicación Web para la gestión de almacén e inventario con análisis y visualización de datos para tiendas de conveniencia.”**

*Presentan:*

**Gerardo Ismael Garzón Domínguez[[1]](#footnote-2)**

**Gabriel de Jesús Muñoz Ramírez[[2]](#footnote-3)**

**Ernesto Daniel Sánchez Becerra[[3]](#footnote-4)**

*Directores:*

**M. en C. Alejandro Botello Castillo M. en C. Nancy Ocotitla Rojas**

*Resumen:*

En este documento, se lleva a cabo un análisis exhaustivo de una aplicación web diseñada para unir las características de los sistemas de WMS e IMS en un solo entorno. Esta aplicación proporciona a las tiendas de conveniencia una herramienta alternativa que facilita una mejor distribución de los recursos de inventario y almacén disponibles. Además, se emplea el análisis de datos para crear un espacio de consulta y visualización de estos, adaptado a las necesidades específicas identificadas en las tiendas de conveniencia.

El documento detalla las etapas fundamentales que componen el diseño, la implementación y la viabilidad del sistema mencionado. Asimismo, se explican las ventajas que ofrece en comparación con otras alternativas disponibles en el mercado.

**Palabras clave:** Aplicación web, Análisis de datos, Tienda De Conveniencia, WMS, IMS, distribución de productos

**Índice**

[**CAPÍTULO 1: INTRODUCCIÓN** 12](#_Toc151981436)

[**1.1 Planteamiento del problema** 12](#_Toc151981437)

[**1.2 Propuesta de solución** 13](#_Toc151981438)

[**1.3 Objetivos** 13](#_Toc151981439)

[**1.3.1 Objetivo general** 13](#_Toc151981440)

[**1.3.2 Objetivos específicos** 13](#_Toc151981441)

[**1.4 Justificación** 14](#_Toc151981442)

[**CAPÍTULO 2: ESTADO DEL ARTE** 15](#_Toc151981443)

[**2.1 Software de paga** 15](#_Toc151981444)

[**2.1.1 Körber (antes HighJump WMS)** 15](#_Toc151981445)

[**2.1.2 Vend: Sistema de Punto de Venta** 16](#_Toc151981446)

[**2.1.3 Manhattan SCALE** 17](#_Toc151981447)

[**2.2 Aplicaciones gratuitas** 18](#_Toc151981448)

[**2.2.1 Square para Tiendas** 18](#_Toc151981449)

[**2.2.2 Odoo (Inventory)** 20](#_Toc151981450)

[**2.2.3 Aplicación web para la gestión de almacén e inventario con análisis y visualización de datos para tiendas de conveniencia (Kv-wims)** 21](#_Toc151981451)

[**2.3 Estudio comparativo** 23](#_Toc151981452)

[**2.3.1 Tabla comparativa** 23](#_Toc151981453)

[**2.3.2 ¿Por qué elegir Kv-wims?** 25](#_Toc151981454)

[**CAPÍTULO 3: MARCO TEÓRICO** 26](#_Toc151981455)

[**3.1. Introducción a la metodología XP** 26](#_Toc151981456)

[**3.1.1 Definición y características:** 26](#_Toc151981457)

[**3.1.2 Principios, roles y responsabilidades:** 27](#_Toc151981458)

[**3.1.3 Tarjetas CRC (Clase-Responsabilidad-Colaboración).** 29](#_Toc151981459)

[**3.1.4 Historias de Usuario** 30](#_Toc151981460)

[**3.1.5 ¿Cómo utilizar esta metodología en este trabajo?** 31](#_Toc151981461)

[**3.1.6 Productos esperados** 32](#_Toc151981462)

[**3.4 Patrones de arquitectura aplicados** 33](#_Toc151981463)

[**3.4.1 Patrón Modelo Vista Template** 33](#_Toc151981464)

[**3.4.2 Patrón Cliente - Servidor** 35](#_Toc151981465)

[**3.5 Python** 35](#_Toc151981466)

[**3.5.1 Django** 36](#_Toc151981467)

[**3.5.2 Type hints** 37](#_Toc151981468)

[**3.5.3 Otras bibliotecas** 38](#_Toc151981469)

[**3.6 JavaScript** 38](#_Toc151981470)

[**3.7 HTML** 39](#_Toc151981471)

[**3.8 CSS** 39](#_Toc151981472)

[**3.9 Oracle-MySQL** 40](#_Toc151981473)

[**3.10 Minería de datos y proceso de descubrimiento de conocimiento en bases de datos (KDD)** 41](#_Toc151981474)

[**3.10.1 La minería de datos y otras disciplinas** 42](#_Toc151981475)

[**3.10.2 Tipos de modelos de minería de datos** 43](#_Toc151981476)

[**3.10.3 Fases del proceso de KDD** 43](#_Toc151981477)

[**3.10.4 Conjunto de datos (Dataset)** 47](#_Toc151981478)

[**3.10.5 Instrumento de investigación** 47](#_Toc151981479)

[**3.10.6 Otras fuentes de datos** 48](#_Toc151981480)

[**3.10.7 Aumento de datos** 48](#_Toc151981481)

[**3.10.8 Algoritmos utilizados** 48](#_Toc151981482)

[**3.10.9 Reglas de asociación para generar ofertas** 49](#_Toc151981483)

[**3.10.10 Árbol de decisión para clasificar nuevos productos** 49](#_Toc151981484)

[**3.10.11 Pronósticos con series de tiempo ganancias y ciclos de restock** 50](#_Toc151981485)

[**3.10.12 Clústeres; más vendidos menos vendidos** 51](#_Toc151981486)

[**3.11 Logística y roles en tiendas de conveniencia.** 52](#_Toc151981487)

[**3.11.1 Roles y responsabilidades.** 53](#_Toc151981488)

[**3.11.2 Logística.** 54](#_Toc151981489)

[**3.12 Desafíos específicos en tiendas de conveniencia.** 54](#_Toc151981490)

[**3.12.1 Recepción de mercancías.** 55](#_Toc151981491)

[**3.12.2 Ubicación y almacenamiento.** 55](#_Toc151981492)

[**3.12.3 Productos perecederos.** 56](#_Toc151981493)

[**3.13 La tecnología en los sistemas de almacenes.** 57](#_Toc151981494)

[**3.13.1 Warehouse Management System (WMS).** 57](#_Toc151981495)

[**3.13.2 Inventory Management System (IMS).** 58](#_Toc151981496)

[**3.13.3 La unión de WMS e IMS: Optimizando la gestión de almacenes e inventario.** 59](#_Toc151981497)

[**3.14 Métricas y Términos de Control.** 59](#_Toc151981498)

[**3.14.1 Control de Calidad.** 60](#_Toc151981499)

[**3.14.2 Stock-Out.** 60](#_Toc151981500)

[**3.14.3 Punto de Reorden (Reorder Point).** 61](#_Toc151981501)

[**3.14.4 Rotación de Inventario (Turnover).** 61](#_Toc151981502)

[**3.14.5 Cross-Docking.** 62](#_Toc151981503)

[**3.14.6 Lead Time de Suministro.** 63](#_Toc151981504)

[**3.14.7 KPI - Indicadores Clave de Rendimiento.** 63](#_Toc151981505)

[**3.14.8 SKU (Stock Keeping Unit).** 65](#_Toc151981506)

[**3.14.9 Análisis ABC.** 65](#_Toc151981507)

[**3.15 Autenticación de usuarios** 66](#_Toc151981508)

[**3.15.1 Función Hash** 66](#_Toc151981509)

[**3.15.2 Salt** 66](#_Toc151981510)

[**3.15.3 Nonce** 67](#_Toc151981511)

[**CAPÍTULO 4: ANÁLISIS DEL SISTEMA** 68](#_Toc151981512)

[**4.1 Estudio de factibilidad** 68](#_Toc151981513)

[**4.1.1 Estudio Técnico** 68](#_Toc151981514)

[**4.1.2 Estudio Económico** 72](#_Toc151981515)

[**4.1.3 Estudio Operacional** 74](#_Toc151981516)

[**4.1.4 Costo del proyecto** 75](#_Toc151981517)

[**4.2 Análisis de riesgos** 78](#_Toc151981518)

[**4.2.1 Identificación** 78](#_Toc151981519)

[**4.2.2 Valoración** 81](#_Toc151981520)

[**4.2.3 Plan de contingencia** 82](#_Toc151981521)

[**4.3 Historias de usuario** 86](#_Toc151981522)

[**4.4 Casos de uso.** 94](#_Toc151981523)

[**4.5 Requerimientos de la base de datos** 100](#_Toc151981524)

[**4.5.1 Requerimientos de datos** 100](#_Toc151981525)

[**4.5.1 Volumen de datos** 100](#_Toc151981526)

[**4.5.2 Diagrama entidad-relación** 103](#_Toc151981527)

[**CAPÍTULO 5 DISEÑO DEL SISTEMA** 108](#_Toc151981528)

[**5.1 Diagrama de clases** 108](#_Toc151981529)

[**5.2 Diagramas UML** 108](#_Toc151981530)

[**5.3 Tarjetas CRC** 108](#_Toc151981531)

[**5.4 Diagrama relacional de la base de datos** 108](#_Toc151981532)

[**5.5 Arquitectura lógica del sistema** 108](#_Toc151981533)

[**5.6 “Wireframes”** 108](#_Toc151981534)

[**Anexos** 116](#_Toc151981535)

[**BIBLIOGRAFÍA** 117](#_Toc151981536)

**Índice de tablas**

[Tabla 1. Resumen fortalezas herramientas investigadas y Kv-wims 24](#_Toc151934585)

[Tabla 2. Componentes del patrón de arquitectura: modelo-vista-controlador 32](#_Toc151934586)

[Tabla 3. Resumen equipo de hardware destinado al proyecto 68](#_Toc151934587)

[Tabla 4. Resumen componentes de software destinados para el proyecto 70](#_Toc151934588)

[Tabla 5. Desglose costos servicios de Azure 70](#_Toc151934589)

[Tabla 6. Costos destinados a gastos generales de desarrollo. 71](#_Toc151934590)

[Tabla 7. Costos individuales destinados al equipo de desarrollo 72](#_Toc151934591)

[Tabla 8. Costos totales destinados al equipo de desarrollo. 72](#_Toc151934592)

[Tabla 9. Costos destinados a los gastos operativos de desarrollo 73](#_Toc151934593)

[Tabla 10. Costo bono de capacitación 73](#_Toc151934594)

[Tabla 11. Resumen de costos por tipo de estudio 74](#_Toc151934595)

[Tabla 12. Ponderación de complejidad para tipos de objeto 74](#_Toc151934596)

[Tabla 13. Análisis ponderación de puntos de objeto 75](#_Toc151934597)

[Tabla 14. Conteo parcial del total de líneas de código por lenguaje de programación 75](#_Toc151934598)

[Tabla 15. Resumen total de líneas de código reutilizadas 75](#_Toc151934599)

[Tabla 16. Tasa de productividad para puntos de objeto 76](#_Toc151934600)

[Tabla 17. Riesgos identificados para el desarrollo del proyecto 80](#_Toc151934601)

[Tabla 18. Criterios categorización de los riesgos 80](#_Toc151934602)

[Tabla 19. Resumen categorización de los riesgos 80](#_Toc151934603)

[Tabla 20. Plan de contingencia riesgo 1 81](#_Toc151934604)

[Tabla 21. Plan de contingencia riesgo 2 81](#_Toc151934605)

[Tabla 22. Plan de contingencia riesgo 3 82](#_Toc151934606)

[Tabla 23. Plan de contingencia riesgo 4 82](#_Toc151934607)

[Tabla 24. Plan de contingencia riesgo 5 82](#_Toc151934608)

[Tabla 25. Plan de contingencia riesgo 6 82](#_Toc151934609)

[Tabla 26. Plan de contingencia riesgo 7 83](#_Toc151934610)

[Tabla 27. Plan de contingencia riesgo 8 83](#_Toc151934611)

[Tabla 28. Plan de contingencia riesgo 9 83](#_Toc151934612)

[Tabla 29. Plan de contingencia riesgo 10 84](#_Toc151934613)

[Tabla 30. Plan de contingencia riesgo 11 84](#_Toc151934614)

[Tabla 31. Plan de contingencia riesgo 12 84](#_Toc151934615)

[Tabla 32. Historia de usuario 1 85](#_Toc151934616)

[Tabla 33. Historia de usuario 2 85](#_Toc151934617)

[Tabla 34. Historia de usuario 3 85](#_Toc151934618)

[Tabla 35. Historia de usuario 4 86](#_Toc151934619)

[Tabla 36. Historia de usuario 5 86](#_Toc151934620)

[Tabla 37. Historia de usuario 6 86](#_Toc151934621)

[Tabla 38. Historia de usuario 7 87](#_Toc151934622)

[Tabla 39. Historia de usuario 8 87](#_Toc151934623)

[Tabla 40. Historia de usuario 9 87](#_Toc151934624)

[Tabla 41. Historia de usuario 10 87](#_Toc151934625)

[Tabla 42. Historia de usuario 11 88](#_Toc151934626)

[Tabla 43. Historia de usuario 12 88](#_Toc151934627)

[Tabla 44. Historia de usuario 13 88](#_Toc151934628)

[Tabla 45. Historia de usuario 14 88](#_Toc151934629)

[Tabla 46. Historia de usuario 15 88](#_Toc151934630)

[Tabla 47. Historia de usuario 16 89](#_Toc151934631)

[Tabla 48. Historia de usuario 17 89](#_Toc151934632)

[Tabla 49. Historia de usuario 18 89](#_Toc151934633)

[Tabla 50. Historia de usuario 19 89](#_Toc151934634)

[Tabla 51. Historia de usuario 20 90](#_Toc151934635)

[Tabla 52. Historia de usuario 21 90](#_Toc151934636)

[Tabla 53. Historia de usuario 22 90](#_Toc151934637)

[Tabla 54. Historia de usuario 23 90](#_Toc151934638)

[Tabla 55. Historia de usuario 24 91](#_Toc151934639)

[Tabla 56. Historia de usuario 25 91](#_Toc151934640)

[Tabla 57. Historia de usuario 26 91](#_Toc151934641)

[Tabla 58. Historia de usuario 27 91](#_Toc151934642)

[Tabla 59. Historia de usuario 28 92](#_Toc151934643)

[Tabla 60. Historia de usuario 29 92](#_Toc151934644)

[Tabla 61. Historia de usuario 31 92](#_Toc151934645)

[Tabla 62. Historia de usuario 32 92](#_Toc151934646)

[Tabla 63. Historia de usuario 33 92](#_Toc151934647)

[Tabla 64. Historia de usuario 34 93](#_Toc151934648)

[Tabla 65. Descripción del diagrama de caso de uso 1 93](#_Toc151934649)

[Tabla 66. Descripción del diagrama de casos de uso 2 94](#_Toc151934650)

[Tabla 67. Descripción del diagrama de casos de uso 3 95](#_Toc151934651)

[Tabla 68. Descripción del diagrama de casos de uso 4 96](#_Toc151934652)

[Tabla 69. Descripción del diagrama de casos de uso 5 97](#_Toc151934653)

[Tabla 70. Descripción del diagrama de casos de uso 6 99](#_Toc151934654)

[Tabla 71. Descripción del diagrama de casos de uso 7 100](#_Toc151934655)

[Tabla 72. 102](#_Toc151934656)

**Índice de figuras**

[Figura 1. Patrón de arquitectura: modelo, vista, controlador 34](#_Toc151981155)

[Figura 2. Patrón de arquitectura: modelo-vista-template 34](#_Toc151981156)

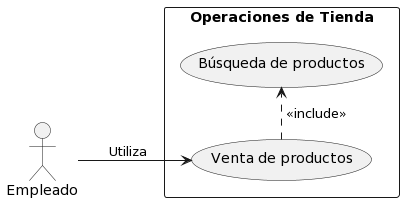
[Figura 3. Patrón de arquitectura: cliente servidor 35](#_Toc151981157)

[Figura 4. 42](#_Toc151981158)

[Figura 5. 43](#_Toc151981159)

[Figura 6. 44](#_Toc151981160)

[Figura 7. Diagrama de casos de uso 1 94](#_Toc151981161)

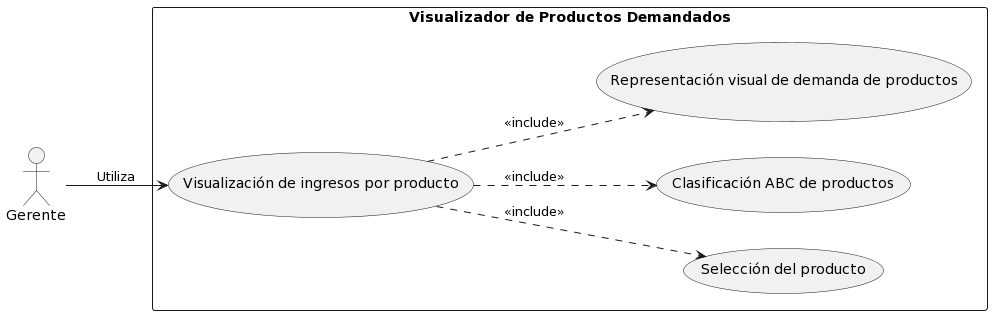
[ 95](#_Toc151981162)

[Figura 8. Diagrama de casos de uso 2 95](#_Toc151981163)

[Diagrama

Descripción generada automáticamente 95](#_Toc151981164)

[Figura 9. Diagrama de casos de uso 3 95](#_Toc151981165)

[ 96](#_Toc151981166)

[Figura 10. Diagrama de casos de uso 4 96](#_Toc151981167)

[Diagrama

Descripción generada automáticamente 97](#_Toc151981168)

[Figura 11. Diagrama de casos de uso 5 97](#_Toc151981169)

[Figura 12. Diagrama de casos de uso 6 98](#_Toc151981170)

[Diagrama

Descripción generada automáticamente 100](#_Toc151981171)

[Figura 13. Diagrama de casos de uso 7 100](#_Toc151981172)

[109](#_Toc151981173)

# **CAPÍTULO 1: INTRODUCCIÓN**

El presente capítulo detalla un análisis de la problemática que se obtuvo como resultado de una investigación orientada a contribuir en el contexto específico de operaciones en las tiendas de conveniencia y los desafíos presentes en la gestión de almacén e inventario de estos lugares.

Además, se presenta una propuesta de desarrollo de una aplicación web, diseñada para satisfacer las necesidades identificadas en relación con la problemática.

## 

## **1.1 Planteamiento del problema**

En una era donde la inmediatez y la conveniencia son esenciales, los clientes esperan encontrar exactamente lo que buscan en el momento exacto en que lo desean. Esta demanda constante ejerce una presión adicional sobre estas tiendas para garantizar que los productos populares y primordiales estén siempre disponibles.

Las tiendas de conveniencia representan un pilar importante en la industria, ofreciendo a los consumidores acceso rápido a una variedad de productos de primera necesidad. Sin embargo, a pesar de su importancia, muchas de estas tiendas enfrentan desafíos significativos en la gestión de sus inventarios y almacenes [1].

En este contexto, surge la pregunta clave: ¿cómo pueden las tiendas de conveniencia la gestionar su inventario y almacén de manera efectiva y asequible buscando satisfacer las necesidades del cliente? Esta cuestión es de vital importancia, ya que una gestión eficiente del inventario no solo impacta en la rentabilidad y competitividad de estas tiendas, sino que también influye en la satisfacción del cliente y en la reducción de desperdicios.

En muchas ocasiones, estos negocios minoristas tales como un Oxxo, 7eleven, CircleK, carecen de las herramientas necesarias para una gestión eficiente del inventario, lo que puede dar como resultado problemas como la pérdida de productos debido a la caducidad, la falta de control sobre los niveles de inventario, la mala organización del almacén y la imposibilidad de realizar un seguimiento preciso de las ventas y los ingresos. Estos problemas no solo generan pérdidas económicas, sino que también pueden afectar la reputación de la tienda y la experiencia del cliente [2].

Entrando en detalle, una mala organización del almacén puede ralentizar las operaciones diarias, lo que impacta negativamente en la eficiencia del negocio. Dada la relevancia de este sector y la creciente competencia en el mercado, es importante abordar estas deficiencias para garantizar la sostenibilidad y éxito de estas tiendas. Por ello, surge la necesidad de investigar y proponer soluciones que optimicen la gestión del inventario y almacén en tiendas de conveniencia, considerando tanto las necesidades actuales del negocio como las expectativas del consumidor.

En este escenario, la gestión de almacén e inventario puede ser especialmente desafiante debido a la falta de espacio de almacenamiento pues generalmente cuentan con un espacio aproximado de 105m2. A su vez, la amplia variedad de productos perecederos añade una complejidad adicional, ya que se debe tener cuidado de evitar la exposición de productos vencidos en el mostrador. Además, los propietarios pueden tener recursos y presupuestos limitados para implementar sistemas de gestión de inventario sofisticados [3].

Dentro de esta perspectiva, es de vital importancia que las tiendas de conveniencia tengan acceso a sistemas de gestión de inventario que sean efectivos, intuitivos y accesibles (que el personal de la tienda pueda utilizarlos eficazmente sin requerir una formación extensa o habilidades técnicas avanzadas.), además de que estén diseñados específicamente para abordar las demandas y necesidades de estos negocios minoristas.

## 

## **1.2 Propuesta de solución**

El desafío abordado por esta propuesta se refiere a la gestión problemática de almacenes e inventarios en tiendas de conveniencia, lo cual conlleva con frecuencia pérdidas económicas debido a factores como la expiración de productos, falta de control de inventario y una mala organización [3].

Implementar este sistema ayudará a las tiendas de conveniencia a disminuir las pérdidas económicas por la mala gestión del inventario, y como consecuencia, mejorar la satisfacción del cliente pues estos también pueden tener una experiencia de compra más agradable y eficiente si los productos están fácilmente accesibles y en stock.

Es por eso que se desarrollará una aplicación web que unifique las funcionalidades de WMS, IMS e integre un módulo de análisis y visualización de datos en un solo sistema en función de permitir la mejora de un 30% de productividad de la tienda de conveniencia ayudando a automatizar el recibimiento, almacenaje, organización y selección de los productos dentro del almacén, al tener noción sobre el espacio que se tiene en el almacén, cuantas existencias hay actualmente en el inventario, y el conocimiento generado mediante los módulos de análisis de datos, se pueden tomar mejores decisiones y disminuir el tiempo de cada proceso.

Un sistema de gestión de almacenes (Warehouse Management System, WMS por sus siglas en inglés) es una solución de software que ofrece visibilidad de todo el inventario en este caso a nivel de una tienda de conveniencia. En términos generales, un sistema de gestión de inventario (Inventory Management System, IMS por sus siglas en inglés) es un sistema de software para el seguimiento de niveles de inventario [4]. Las funcionalidades de ambos tipos de sistemas serán integradas en nuestra propuesta.

## 

## **1.3 Objetivos**

### **1.3.1 Objetivo general**

Diseñar un prototipo de sistema integrado de gestión de almacén, inventario, análisis y visualización de datos, para tiendas de conveniencia pequeñas, que facilite la rotación de productos, mejore la eficiencia en las operaciones del almacén y proporcione una interfaz gráfica de usuario que integre las funcionalidades esenciales del sistema para el personal y la administración.

### **1.3.2 Objetivos específicos**

* Diseñar aplicación Web que combine WMS e IMS para tiendas de conveniencia pequeñas.
* Integrar análisis de datos para ofrecer información valiosa para la toma de decisiones dentro de la tienda de conveniencia.
* Diseñar una interfaz de usuario que integre las funcionalidades esenciales del sistema para el personal y la administración.

## **1.4 Justificación**

Se propone un sistema computacional cliente-servidor, vista-controlador cuyas funcionalidades principales son aquellas de los sistemas gestores de inventario (Inventory Management System) IMS, sistemas gestores de almacenes (Warehouse Management System) WMS, así como la integración de un módulo de analítica de datos que proporcionará resultados descriptivos y predictivos en base a datos generados por la misma instancia del sistema.

Entre las características principales de la propuesta se encuentran: Una unificación entre las funcionalidades de sistemas que por lo general se conciben como diferentes (IMS, WMS), entre estas funcionalidades generales tenemos; registro de bajas y altas por ventas, reposiciones y caducidades, usuarios con privilegios administrativos como edición de productos de venta y ofertas. Determinación de la ubicación de los productos dentro de los almacenes y seguimiento de su lote, alertas según los niveles de stock.

Al mantener un inventario bien organizado y en constante rotación, las tiendas pueden asegurarse de que los productos estén frescos y en buen estado para su venta.

Otra característica importante del sistema es la visualización de datos, mediante la cual se intentará expresar la mayor cantidad de información útil (como las dimensiones del almacén, disposición de anaqueles y productos dentro de los almacenes, y conclusiones generadas por el análisis de datos) para un entendimiento más accesible a los usuarios y así mismo como para facilitar posibles procesos de capacitación.

El uso WMS e IMS ayudará a las tiendas de conveniencia en facilitar una rotación adecuada del inventario, organización de productos y proporcionar información valiosa para la toma de decisiones. En comparación con las alternativas en el mercado, nuestro sistema, diseñado específicamente para las tiendas de conveniencia, integra las funcionalidades de gestión de almacén, inventario y la aplicación del análisis de datos, que, mediante la visualización de datos, hace más accesible el uso y capacitación del sistema para potenciales clientes.

Un factor relevante para considerar es que las alternativas en el mercado pueden resultar en sistemas complejos y costosos que no están diseñados específicamente para las necesidades de las tiendas de conveniencia. Estos sistemas pueden ser difíciles de usar y pueden requerir una capacitación adicional para el personal e incluso los costos asociados con la adquisición, implementación y mantenimiento de estos sistemas pueden ser prohibitivos para los propietarios de tiendas de conveniencia con recursos limitados.

# **CAPÍTULO 2: ESTADO DEL ARTE**

A lo largo de este capítulo, se presenta el resultado de la investigación enfocada en productos de software gratuito o de paga que han sido elegidos para abordar las necesidades específicas de la gestión de almacenes e inventarios en el contexto de las tiendas de conveniencia.

Para complementar este análisis, se detallan las características, funcionalidades y costos de estas soluciones, con el propósito de establecer una comparación con nuestra propuesta de solución al problema.

## 

## **2.1 Software de paga**

De acuerdo resultados de la investigación llevada a cabo, se han identificado diversos recursos disponibles en el mercado para la gestión del almacén. Estos recursos pueden ser software de pago o servicios que requieren una cotización de las empresas proveedoras. En esta sección del documento se muestra el análisis de los recursos antes mencionados de pago disponibles.

### **2.1.1 Körber (antes HighJump WMS)**

Körber es una solución de software diseñada para optimizar y gestionar eficientemente las operaciones de un almacén o centro de distribución. Esta herramienta se utiliza en una variedad de industrias, incluyendo logística, fabricación y comercio minorista, para mejorar la visibilidad, la precisión y la productividad en el manejo de inventario y la ejecución de pedidos [5]. Como parte de los servicios que ofrece la herramienta se presentan las siguientes funcionalidades teniendo en cuenta que van dirigidas a las pequeñas y medianas empresas que manejan WMS:

* **Gestión de inventario:** 
  + Control preciso del inventario en tiempo real.
  + Seguimiento de la ubicación, cantidad y estado de los productos en el almacén.
* **Optimización de rutas:** 
  + Ayuda a optimizar la selección de rutas.
  + Asignación de tareas para minimizar los tiempos de recogida y entrega.
* **Recepción y despacho de mercancías:** 
  + Facilita la recepción eficiente de productos.
  + Verificación de calidad.
  + Preparación de pedidos para su envío, asegurando la exactitud y la rapidez en el proceso.
* **Administración de pedidos:** 
  + Administrar pedidos entrantes y salientes, gestionando prioridades y asignando recursos de manera eficiente.
* **Cumplimiento normativo:** 
  + Cumplir con regulaciones y estándares de la industria, como las normas de seguridad y trazabilidad.
* **Integración con otros sistemas:** 
  + Integración con sistemas ERP (Enterprise Resource Planning) y otros sistemas empresariales para un flujo de datos sin problemas.
* **Seguimiento y reportes:** 
  + Seguimiento en tiempo real.
  + Generación de informes personalizables para el análisis de datos y la toma de decisiones informadas.
* **Automatización:** 
  + Automatización a través de tecnologías como RFID (Identificación por Radiofrecuencia).
  + Lectura de códigos de barras para agilizar las operaciones y reducir errores.
* **Adaptabilidad:** 
  + Capacidad de adaptarse a las necesidades específicas de un almacén o centro de distribución.

Finalmente, es importante señalar que existen otros servicios dentro del paquete de soluciones que se ofrecen, algunos de ellos son sistema de gestión de transporte, control por voz y robots móviles autónomos, sin embargo, adquirir estos servicios generan un costo mayor y el uso de estos servicios están sujetos a diferentes variables como el espacio del almacén, el total de inventario y el total de personal que cuenta la tienda [5].

### **2.1.2 Vend: Sistema de Punto de Venta**

Vend es un sistema de punto de venta moderno y versátil, diseñado para adecuarse a una variedad de tipos de negocios. Proporciona una plataforma digital intuitiva para gestionar ventas, inventarios, clientes y más [6]. A continuación, se describen sus características y funciones principales:

* **Selección de Categorías de Negocio:** Al iniciar, Vend permite seleccionar la categoría específica de tu negocio, tales como:
  + Joyería
  + Deportes
  + Regalos
  + Para mascotas
  + Servicios
  + Vinos y licores
  + Comida y bebida
  + Libros, juguetes, entre otros.
* **Clasificación de Negocio:** La herramienta categoriza el negocio según lo que vendes, su clasificación general y tus ganancias anuales. Las principales clasificaciones incluyen:
  + Retail
  + E-commerce
  + Restaurante
  + Golf
* **Inicio para Nuevos Usuarios:**
  + Para los nuevos usuarios, se presenta una tienda demo (a modo de tutorial) o la opción de crear una tienda desde cero. Esta fase inicial recopila información importante sobre la tienda, como cuándo planeas vender con Vend y qué sistema utilizas actualmente.
* **Interfaz de Usuario:** La pantalla de inicio es simple y permite:
  + Agregar productos.
  + Practicar la realización de ventas.
  + Acceder a la creación de reportes.
* **Menú Principal:** Los accesos principales del sistema incluyen:
  + Reportes
  + Catálogo
  + Servicios
  + Inventario
  + Clientes
  + Configuración
* **Gestión de Productos:** La sección de productos ofrece:
  + CRUD (Crear, Leer, Actualizar, Eliminar) de productos.
  + Exportación de la tabla de productos como CSV.
  + Muestra de datos relevantes como precio de venta, proveedor, marca y estado del producto (activo/inactivo).
* **Promociones:** Incluye un submenú que permite crear promociones y establecer sus fechas de duración.
* **Gestión de Marcas y Proveedores:** Herramientas integradas para gestionar marcas y proveedores.
* **Inventario:** Ofrece una gama de herramientas como:
  + Listar productos por categoría.
  + Control de stock.
  + Pedir, recibir, regresar, transferir y rastrear stock.
  + Conteo de inventario manual o mediante aplicación.
* **Gestión de Clientes:**
  + Permite registrar, agrupar y gestionar clientes según características demográficas.

### **2.1.3 Manhattan SCALE**

SCALE es un sistema de gestión de almacenes asequible diseñado por Manhattan Associates. El WMS SCALE de Manhattan ayuda a los negocios a optimizar sus operaciones de almacén a través de herramientas de gestión de inventario, surtimiento de pedidos y administración de transporte.

Manhattan SCALE es una plataforma diseñada en la nube que puede ser consultado desde cualquier dispositivo que cuente con un navegador de internet. También es compatible con varios idiomas, monedas y unidades de medida [7]. A continuación, se describen sus características y funcionalidades principales:

* **Amplia gama de soluciones:** 
  + Ofrece una suite de soluciones para la gestión de almacenes.
  + Brinda una suite para la planificación de la cadena de suministro.
  + Permite una integración fluida con otros sistemas.
* **Escalabilidad:** 
  + Brinda una plataforma escalable.
  + Puede adaptarse a las necesidades de empresas de diversos tamaños y sectores.
* **Funcionalidades avanzadas:** 
  + Ofrece funcionalidades avanzadas para la gestión de pedidos
  + Ofrece el seguimiento de inventario en tiempo real.
  + Abarca el cumplimiento normativo.
  + Permite la automatización de tareas y optimización de rutas.
* **Integración de sistemas:** 
  + Integración con sistemas ERP.
  + Integración con aplicaciones empresariales para una visión completa de la cadena de suministro.
* **Tecnología avanzada:** 
  + Funcionalidades mediante IoT (Internet de las Cosas).
  + Análisis de datos y aprendizaje automático para mejorar las operaciones y tomar decisiones más informadas.
* **Automatización y movilidad:** 
  + Automatización de procesos en el almacén
  + Brinda compatibilidad con dispositivos móviles.
  + Recopilación de datos y la ejecución de tareas.
* **Capacidades analíticas y de informes:** 
  + Ofrece análisis y generación de informes avanzados para el seguimiento y la mejora del rendimiento del almacén.
* **Apoyo a la sostenibilidad:** 
  + Proporciona funcionalidades para ayudar a reducir su huella de carbono
  + Cumple con estándares ambientales [7].

## **2.2 Aplicaciones gratuitas**

Por otra parte, se han identificado software de gestión del almacén que son de acceso gratuito. En esta sección del documento, se detalla la información correspondiente a dichos recursos gratuitos, brindando así un panorama general de los recursos antes mencionados que se encuentran disponibles en el mercado.

### **2.2.1 Square para Tiendas**

Es una solución de punto de venta integral que ofrece una amplia variedad de funciones para ayudarte a gestionar y hacer crecer tu negocio de venta minorista. A continuación, se proporciona un resumen detallado de las características clave de Square para Tiendas:

* **Punto de Venta Eficiente:**
  + Square para Tiendas permite a los minoristas procesar pagos de manera rápida y sencilla en la caja registradora. Puedes aceptar tarjetas de crédito y débito, así como pagos en efectivo.
  + La aplicación de punto de venta se ejecuta en dispositivos móviles, como teléfonos inteligentes y tabletas, lo que brinda flexibilidad para atender a los clientes en el lugar y mejorar la experiencia de compra.
* **Administración de Inventarios:**
  + Ofrece un sistema de control de inventario avanzado que permite dar seguimiento a tus productos y variantes.
  + Permite realizar conteos cíclicos de inventario para mantener registros precisos de las existencias.
  + Facilita la gestión de surtidos y la creación de nuevas variantes de productos.
* **Órdenes de Compra:**
  + Permite crear y administrar órdenes de compra para reponer existencias, realizar un seguimiento de proveedores y recibir inventario.
  + La función de órdenes de compra automáticas sugiere automáticamente reponer productos en función de datos como el inventario actual, las predicciones de ventas y los plazos de entrega de proveedores.
  + Puedes importar órdenes de compra desde archivos CSV o Excel para ahorrar tiempo y mejorar la precisión.
* **Gestión de Proveedores:**
  + Square para Tiendas te permite administrar tu lista de proveedores, agregar nuevos proveedores y asociarlos con productos y variantes.
  + Puedes importar en grupo información de proveedores desde archivos Excel o CSV.
  + Ofrece la posibilidad de editar surtidos de productos específicos de proveedores y filtrar el inventario por proveedor.
* **Etiquetas de Código de Barras:**
  + Facilita la impresión de etiquetas de código de barras para tus productos y órdenes de compra.
  + Esto ayuda en la gestión del inventario y en la identificación rápida de productos.
* **Contabilidad y Análisis:**
  + Square para Tiendas proporciona herramientas para llevar un registro de tus ventas y gastos.
  + Ofrece informes detallados y análisis de ventas que te permiten tomar decisiones comerciales informadas.
* **Aplicaciones Móviles y Terminales de Punto de Venta:**
  + Square es compatible con una variedad de dispositivos, incluyendo iPhones, iPads y terminales de punto de venta Square Terminal.
  + Esto permite llevar a cabo transacciones en diferentes entornos comerciales.
* **Aplicaciones para Restaurantes:**
  + Square para Tiendas incluye aplicaciones específicas para restaurantes que permiten administrar menús, pedidos y reservas.
* **Gestión de Empleados:**
  + Permite asignar roles a los empleados y dar acceso a funciones específicas de la aplicación.
  + Facilita la gestión del tiempo y la nómina de los empleados.
* **Servicios de Comercio Electrónico:** 
  + Square ofrece servicios de comercio electrónico que te permiten vender en línea y administrar tus ventas en una plataforma unificada.
* **Herramientas de Marketing y Lealtad:**
  + Ofrece herramientas de marketing y programas de lealtad que te permiten atraer y retener clientes.
* **Integraciones y API:** 
  + Square para Tiendas es compatible con una variedad de integraciones y una API que te permite conectar con otras aplicaciones y servicios.
* **Soporte al Cliente:** 
  + Square proporciona asistencia al cliente y recursos de aprendizaje en línea para ayudarte a aprovechar al máximo la plataforma.
* **Planes de Precios:** 
  + Square para Tiendas ofrece versiones gratuitas y suscripciones premium con características adicionales.
* **Seguridad:** 
  + Square prioriza la seguridad de los pagos y los datos de los clientes, cumpliendo con las normativas de seguridad de la industria.

### **2.2.2 Odoo (Inventory)**

Odoo es una plataforma de gestión de inventario que permite monitorizar el ciclo de vida de los productos, definir reglas de reabastecimiento y generar informes básicos. A pesar de sus características limitadas en la versión estándar, es ampliable mediante complementos de pago.

* No ofrece una variedad de tipos de usuarios.
* Proporciona diagramas que ilustran el ciclo de vida de los productos.
* No incluye diagramas específicos para WMS.
* Establece reglas para el reabastecimiento de stock, permitiendo al usuario definir un número mínimo de existencias.
* La edición relativa a los almacenes se limita únicamente a la dirección del mismo.
* Genera informes en formato de lista, siendo estos relativamente básicos.

**Rendimiento**

* Durante sus operaciones, permite registrar desperdicios o "scrap".
* Es posible ampliar sus funcionalidades, pero hay que considerar costos adicionales. La versión básica tiene características limitadas.
* Cuenta con un tablero de control (dashboard) y una visualización de datos básica. Las visualizaciones parecen ser simplemente consultas ordenadas de mayor a menor.
* Ofrece la opción de visualizar periodos que varían desde 7 días hasta 3 años.
* Permite seleccionar un almacén en específico.
* Facilita la visualización de estadísticas por producto.
* Habilita la creación de entornos colaborativos mediante la invitación de otros usuarios.
* Ofrece opciones para agregar múltiples idiomas.
* Permite alternar unidades de medida entre:
* Kilogramos y libras.
* Metros y pies cúbicos.
* Genera automáticamente un subdominio basado en cómo registres tu empresa durante el proceso de inscripción en la aplicación.
* Cuenta con complementos (plugins) para correo electrónico, Google Calendar, Outlook Calendar, entre otros. Estos complementos son gratuitos por un período de 15 días.
* Ofrece la funcionalidad de enviar mensajes de texto (SMS) utilizando créditos previamente adquiridos.

### **2.2.3 Aplicación web para la gestión de almacén e inventario con análisis y visualización de datos para tiendas de conveniencia (****Kv-wims)**

De acuerdo con los resultados de una investigación previa de herramientas que ofrecen servicios para la gestión de almacén e inventario, surge la idea de crear y proporcionar una alternativa que pretende ser competitiva y está dirigida para aquellas tiendas de conveniencia cuyos recursos económicos podrían no ser suficientes para adquirir alguna de esas opciones debido a que algunas de ellas son de índole gratuita por pequeños periodos de prueba no mayor a 7 días además de que cada nueva funcionalidad tiene un costo extra. Por otra parte, están las opciones que son completamente de paga y que se deben cotizar con las empresas distribuidoras de acuerdo con las características de cada tienda de conveniencia.

Knowledge & visualization for warehouse inventory management system (Kv-wims, por sus siglas en inglés) es la alternativa que integra las principales funcionalidades de WMS e IMS en un solo sitio y que permitirá a los usuarios realizar una administración adecuada y sencilla de los almacenes. Kv-wims se destaca en comparación a las demás opciones por las siguientes razones:

* **Unificación de Funcionalidades:**
  + Kv-wims integra tanto las características de un sistema de gestión de inventario (IMS) como de un sistema de gestión de almacenes (WMS). Esto significa que los usuarios pueden administrar su inventario y la disposición de productos en los almacenes de manera eficiente en un solo sistema.
* **Análisis de Datos Avanzado:**
  + Kv-wims incluye un módulo de análisis de datos que proporciona resultados descriptivos y predictivos basados en datos generados por el sistema. Esto brinda información valiosa para la toma de decisiones comerciales y estratégicas.
* **Facilidad de Uso:**
  + A diferencia de algunas de las alternativas en el mercado, Kv-wims está diseñado específicamente para las necesidades de las tiendas de conveniencia. Ofrece una interfaz de usuario integrada y accesible, lo que significa que los propietarios y empleados pueden utilizarlo con facilidad sin la necesidad de una capacitación extensa.
* **Costos Asequibles:**
  + Kv-wims se presenta como una inversión accesible para las tiendas de conveniencia con recursos limitados. Esto es importante, ya que algunos sistemas en el mercado pueden resultar en costos prohibitivos debido a la adquisición, implementación y mantenimiento.
* **Mejora en la Satisfacción del Cliente:** 
  + Al mantener un inventario bien organizado y en constante rotación, Kv-wims puede mejorar significativamente la satisfacción del cliente. Los productos frescos y disponibles en stock brindan a los clientes una experiencia de compra más agradable y eficiente, lo que puede aumentar la fidelidad del cliente y las ventas.
* **Aplicación web responsiva:**
  + Kv-wims permite la ejecución en dispositivos móviles, como teléfonos inteligentes y tabletas, lo que brinda flexibilidad para atender a los clientes en el lugar y mejorar la experiencia de compra.
* **Control de Inventarios Avanzado:**
  + Kv-wims brinda un control de inventario avanzado que permite el seguimiento de productos y variantes, conteos cíclicos y gestión de surtidos, lo que ayuda a mantener registros precisos de las existencias.
* **Gestión de Proveedores y Órdenes de Compra:**
  + Kv-wims permite administrar la lista de proveedores, asociarlos con productos y variantes, y crear órdenes de compra para reponer existencias, lo que simplifica la gestión de proveedores y la recepción de inventario.
* **Contabilidad y Análisis Integrados:** 
  + Kv-wims ofrece herramientas para llevar un registro de ventas y gastos, así como informes detallados y análisis de ventas que permiten tomar decisiones comerciales informadas.
* **Visualización Gráfica de Almacenes:**
  + A diferencia de las competencias, Kv-wims ofrece una visualización en 2D de los almacenes que permite a los usuarios ver de manera gráfica la disposición de los productos en sus almacenes. Esta característica mejora significativamente la capacidad de los usuarios para comprender y optimizar la organización de su espacio de almacenamiento.
* **Modelado Visual de Anaqueles y Contenedores:**
  + Kv-wims permite a los usuarios modelar visualmente los tipos de anaqueles y contenedores físicos presentes en sus almacenes. Esto garantiza que el sistema se asemeje a la configuración física de su tienda, lo que maximiza la funcionalidad de la aplicación y reduce la necesidad de costosas adaptaciones en el mundo real.
* **Simplificación de la Capacitación:**
  + La visualización en 2D hace que la capacitación del personal sea más eficiente. Los empleados pueden entender rápidamente la disposición de los productos en el almacén y cómo interactuar con el sistema, lo que acelera la implementación y reduce los costos asociados con la capacitación.
* **Identificación Rápida de Productos:**
  + La visualización gráfica facilita la identificación rápida de productos y la ubicación de productos en el almacén. Esto contribuye a una experiencia de compra más eficiente y reduce el tiempo que los empleados deben dedicar a buscar productos.
* **Optimización de Almacenes:** 
  + Al proporcionar una vista en 2D de los almacenes, este sistema ayuda a los usuarios a optimizar la distribución de productos y a tomar decisiones informadas sobre la disposición de sus almacenes. Esto conduce a un uso más eficiente del espacio y ahorro de tiempo en la gestión de inventario.
* **Menor Costo General:**
  + A diferencia de otras soluciones que pueden involucrar costosos planes de suscripción o servicios complementarios de pago, Kv-wims se enfoca en proporcionar un paquete integral a un costo más asequible. Esto garantiza que los propietarios de tiendas de conveniencia con recursos limitados puedan acceder a una solución de calidad sin incurrir en gastos excesivos.
* **Mejor Funcionalidad a Precio Accesible:**
  + A pesar de ofrecer precios competitivos, Kv-wims no sacrifica la funcionalidad. Proporciona una amplia gama de características que incluyen la gestión de inventario, gestión de almacenes y análisis de datos, lo que supera a las herramientas gratuitas y algunas de las soluciones de pago en términos de capacidades.
* **Herramientas de Calidad asequibles:**
  + En comparación con las aplicaciones gratuitas y algunas de las opciones de pago, Kv-wims se basa en la integración de sistemas de gestión de inventario (IMS) y gestión de almacenes (WMS). Esto permite una mayor flexibilidad y eficiencia en la administración del inventario y los almacenes, lo que no siempre se encuentra en aplicaciones gratuitas.
* **Reducción de Costos en Capacitación:**
  + Kv-wims se ha diseñado pensando en la facilidad de uso, lo que disminuye la necesidad de costosas sesiones de capacitación. El personal de la tienda podrá utilizar Kv-wims de manera efectiva sin requerir una inversión significativa en tiempo y recursos de capacitación

## **2.3 Estudio comparativo**

En esta sección del trabajo, se ha elaborado una tabla comparativa de diversas funcionalidades y características de los sistemas previamente investigados, los cuales están dirigidos hacia la gestión de almacén e inventario en el contexto de tiendas de conveniencia.

Además, se proporciona una explicación detallada que destaca las fortalezas que Kv-wims posee, junto con las razones relevantes por las cuales las tiendas de conveniencia deben considerar la adquisición del sistema.

### **2.3.1 Tabla comparativa**

A continuación, se muestra un resumen de las fortalezas y funcionalidades de los sistemas analizados, incluyendo Kv-wims que es la propuesta de solución.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Fortalezas y funcionalidades** | **Herramientas** | | | | | |
| Körber (antes HighJump WMS) | Vend: Sistema de Punto de Venta | Manhattan SCALE | Square para Tiendas | Odoo (inventory) | Kv-wims |
| Visualización gráfica de almacenes |  |  |  |  | Insignia de cruz contorno | Marca de insignia1 contorno |
| Modelado visual de almacenes y contenedores | Marca de insignia1 contorno | Insignia de cruz contorno | Marca de insignia1 contorno | Insignia de cruz contorno | Insignia de cruz contorno | Marca de insignia1 contorno |
| Análisis de datos | Marca de insignia1 contorno  (con pago) | Marca de insignia1 contorno  (con pago) | Marca de insignia1 contorno  (con pago) | Marca de insignia1 contorno | Insignia de cruz contorno | Marca de insignia1 contorno |
| Manejo de inventario | Marca de insignia1 contorno | Marca de insignia1 contorno | Marca de insignia1 contorno | Marca de insignia1 contorno | Marca de insignia1 contorno | Marca de insignia1 contorno |
| Manejo de pedidos | Marca de insignia1 contorno  (con pago) | Insignia de cruz contorno | Marca de insignia1 contorno  (con pago) | Marca de insignia1 contorno | Insignia de cruz contorno | Insignia de cruz contorno |
| Costo (USD) | Se cotiza con la empresa | $119  $169  $249 | Se cotiza con la empresa | Gratis (versión base) $60 (versión pagada) | Gratis (versión base)  $15(Versión standard)  $24(versión personalizada) | Vea capitulo 4 |
| Generación de reportes | Marca de insignia1 contorno | Marca de insignia1 contorno  (con pago) | Marca de insignia1 contorno | Marca de insignia1 contorno | Insignia de cruz contorno | Marca de insignia1 contorno |
| Permisos de usuario | Insignia de cruz contorno | Insignia de cruz contorno | Insignia de cruz contorno | Insignia de cruz contorno | Insignia de cruz contorno | Marca de insignia1 contorno |
| Planes de precio | Marca de insignia1 contorno | Marca de insignia1 contorno | Marca de insignia1 contorno | Marca de insignia1 contorno | Marca de insignia1 contorno | Insignia de cruz contorno |

1. Resumen fortalezas herramientas investigadas y Kv-wims

### **2.3.2 ¿Por qué elegir Kv-wims?**

La integración de distintas funcionalidades en una sola herramienta resulta ser bastante útil pues no solo ayuda a cumplir con las tareas por las que fue creada, sino que ofrece una alternativa completa con la que podrás llevar a cabo una eficaz gestión de tu almacén.

Kv-wims sobresale por algunas razones clave que lo distinguen en comparación de otras alternativas de gestión almacén en el mercado. En primer lugar, se ofrece una interfaz gráfica y personalizable que simplifica la navegación, adaptación, visualización y gestión de las necesidades específicas de cada tienda. Esto se traduce en una curva de aprendizaje más corta y un aumento significativo en la eficiencia operativa para los clientes.

Además, Kv-wims ha sido diseñado para llevar a cabo un manejo sencillo de los datos del inventario, pues el usuario tendrá la posibilidad de consultar dicha información de los productos que conforman el inventario de manera escrita por medio de la generación de reportes o por medio del módulo visualizador del almacén que brinda una vista 2D del mismo.

En adición, el uso de algoritmos de minería de datos ofrece una gama de posibilidades para el manejo de datos históricos pues se implementan módulos para generar conocimiento en función de datos generados por la instancia del sistema, por ejemplo, mediante reglas de asociación se identifica que productos se suelen vender en conjunto, de igual forma haciendo un análisis de las características de productos existentes, nuevos lanzamientos de productos dentro de la tienda, se podrán categorizar adecuadamente de manera automática, por otra parte, se proveerá un módulo para realizar clustering, el cual agrupará a los productos en un rango de más y menos vendidos, esto, con el fin de exponer esta información al usuario, y pueda tomar decisiones para disminuir la pérdida de inventario por caducidad.

Finalmente, el costo de adquisición para un comercio que carezca de los medios financieros necesarios para acceder a un servicio de gran envergadura disminuye sustancialmente, dado que se contempla que el sistema sea asequible para aquellos establecimientos minoristas que exhiban características específicas, tales como un presupuesto limitado, un espacio de almacenamiento promedio y un reducido número de empleados.

# **CAPÍTULO 3: MARCO TEÓRICO**

En el proceso de desarrollo de este capítulo, se exponen los temas trascendentales, las tecnologías pertinentes y los términos específicos que componen el fundamento teórico de la propuesta, con el propósito de proporcionar una visión minuciosa en cuanto a la extensión y las capacidades del sistema.

## 

## **3.1. Introducción a la metodología XP**

### **3.1.1 Definición y características:**

La Programación Extrema (XP) es una metodología de desarrollo de software que se enmarca en las metodologías ágiles. Fue introducida a mediados de la década de 1990 por Kent Beck y ha ganado una amplia aceptación en la industria del desarrollo de software debido a su enfoque centrado en el cliente y su capacidad para adaptarse a los cambios en los requisitos del proyecto.

* **Antecedentes**: Los antecedentes de XP se encuentran en varias disciplinas, incluyendo la ingeniería de software, la teoría de sistemas y la psicología. Beck y su equipo de desarrollo tomaron inspiración de estas disciplinas para crear un enfoque que abordara los desafíos comunes en el desarrollo de software, como la falta de comunicación efectiva entre equipos, la dificultad en la gestión de requisitos cambiantes y la necesidad de entregar software de alta calidad.
* **Origen y evolución en la Gestión de Proyectos de Desarrollo de Software**: XP surgió como una respuesta a las limitaciones de las metodologías tradicionales de desarrollo de software, como el modelo en cascada. A medida que los proyectos de software se volvieron más complejos y dinámicos, se reconoció la necesidad de enfoques más flexibles y orientados al cliente. XP ha evolucionado a lo largo de los años a medida que la comunidad de desarrollo de software ha refinado y adaptado sus prácticas.

**Características:**

* **Comunicación Constante**: enfatiza la comunicación constante entre desarrolladores, clientes y otros interesados. Se fomenta la colaboración cara a cara y la retroalimentación continua para garantizar que los requisitos sean comprendidos y satisfechos de manera efectiva.
* **Desarrollo Iterativo e Incremental**: divide el proyecto en pequeñas iteraciones. En cada iteración, se desarrolla, prueba y entrega una parte funcional del software. Esto permite una entrega temprana y frecuente de valor al cliente.
* **Pruebas Unitarias**: promueve la escritura de pruebas unitarias antes de escribir el código de producción. Las pruebas unitarias garantizan que el código funcione correctamente y se puede refactorizar de manera segura.
* **Simplicidad**: aboga por mantener el diseño y la implementación del software lo más simple posible. Se evitan características innecesarias y se prioriza la funcionalidad vital.

### **3.1.2 Principios, roles y responsabilidades:**

**1. Principios**

* **Retroalimentación Continua**

El principio de retroalimentación continua en la Programación Extrema (XP) destaca la importancia de obtener comentarios regulares del cliente a lo largo del proceso de desarrollo. Esta retroalimentación constante es sustancial para garantizar que el software evolucione de manera acorde a las necesidades del cliente. Para lograr esto, XP emplea prácticas como la inspección y la revisión del código. La inspección consiste en examinar el código para identificar errores y defectos, mientras que la revisión implica presentar el código a un grupo de personas para obtener sus comentarios.

La retroalimentación continua permite al equipo de desarrollo detectar y corregir problemas de manera temprana en el proceso, lo que resulta en software de alta calidad que cumple con las expectativas del cliente.

* **Simplicidad**

El principio de simplicidad en XP enfatiza que el software debe ser lo más simple posible en términos de su diseño, estructura y funcionamiento. Esto implica que el software debe ser fácil de entender, utilizar y mantener. XP utiliza prácticas como el Desarrollo Dirigido por Pruebas (TDD) y la refactorización para lograr la simplicidad. En TDD, se escriben pruebas antes de desarrollar el código, lo que ayuda a diseñar el software de manera clara y enfocada. La refactorización implica mejorar la estructura del código sin cambiar su funcionalidad.

La simplicidad es primordial para asegurar que el software sea fácil de mantener y que su evolución futura sea manejable.

* **Comunicación**

El principio de comunicación en XP establece que la comunicación efectiva es fundamental para el éxito del desarrollo de software. Esto implica que los miembros del equipo de desarrollo deben mantener una comunicación constante y eficiente entre ellos y con el cliente. Para lograr esto, XP utiliza prácticas como la programación en pareja (pair programming) y las reuniones diarias conocidas como "stand-ups." La programación en pareja implica que dos desarrolladores trabajen en el mismo código, lo que fomenta la colaboración y la revisión constante del trabajo. Los "stand-ups" diarios son reuniones cortas donde el equipo discute el progreso del proyecto y los obstáculos que puedan surgir.

Una comunicación efectiva es vital para asegurarse de que el equipo comprenda los requisitos del cliente y para garantizar que el software se desarrolle de acuerdo con esas necesidades.

* **Flexibilidad**

El principio de flexibilidad en XP subraya la importancia de que el software sea adaptable a los cambios. Esto significa que el software debe ser fácil de modificar y actualizar a medida que evolucionan los requisitos del cliente o cambian las condiciones del entorno tecnológico. XP emplea prácticas como el cambio incremental y la capacidad de adaptación al cambio. El cambio incremental implica que el software se entrega en pequeños incrementos funcionales, lo que permite responder ágilmente a las necesidades cambiantes del cliente. La adaptación al cambio se refiere a la capacidad de ajustar el software en respuesta a cambios en los requisitos del cliente o en la tecnología utilizada.

La flexibilidad es fundamental para garantizar que el software pueda satisfacer de manera continua las necesidades en constante evolución del cliente.

**2. Roles y responsabilidades:**

Se designan roles y responsabilidades para los miembros del equipo con el objetivo de optimizar el proceso de desarrollo de software y garantizar la entrega de productos de alta calidad. A continuación, se detallan los roles típicos en un equipo XP y las funciones y responsabilidades asociadas a cada uno.

* **Cliente**

El cliente en XP es el principal interesado o la entidad que requiere el desarrollo de un software específico. Sus responsabilidades incluyen:

Definir los requisitos del software: El cliente es responsable de especificar y detallar los requisitos del software, tanto funcionales como no funcionales. Esto implica definir qué deberá hacer el software y qué cualidades debe poseer.

Proporcionar retroalimentación al equipo de desarrollo: Es fundamental que el cliente mantenga una comunicación constante con el equipo de desarrollo. Debe ofrecer retroalimentación regular sobre el software en desarrollo, lo que incluye validar que cumple con sus necesidades y realizar ajustes cuando sea necesario.

* **Programador**

El programador es la figura central en el proceso de desarrollo del software y asume las siguientes responsabilidades:

Escribir el código del software: La tarea principal del programador es implementar el código del software de acuerdo con los requisitos establecidos por el cliente. Debe escribir código limpio y funcional.

Escribir pruebas para el código: El programador también es responsable de desarrollar pruebas para el código que ha escrito. Estas pruebas ayudan a verificar que el software funciona correctamente y cumple con los requisitos.

Refactorizar el código: La refactorización es una práctica clave en XP. El programador debe mejorar la estructura y calidad del código sin cambiar su funcionalidad. Esto garantiza que el código sea mantenible y escalable.

* **Tester**

El tester desempeña un papel crítico en la garantía de calidad del software y está a cargo de las siguientes tareas:

Probar el software para garantizar que cumple con los requisitos: El tester es responsable de realizar pruebas exhaustivas en el software para asegurarse de que cumple con los requisitos definidos por el cliente. Esto implica probar diversas funcionalidades y escenarios.

Proporcionar retroalimentación al equipo de desarrollo: Después de completar las pruebas, el tester debe proporcionar retroalimentación al equipo de desarrollo. Esto incluye informar sobre problemas y ayudar a identificar soluciones para mejorar el software.

* **Entrenador**

El entrenador en XP es un experto en la metodología y asume un rol orientador y de apoyo. Sus responsabilidades abarcan:

Ayudar al equipo a aplicar los principios y prácticas de XP: El entrenador juega un papel crucial al ayudar al equipo a implementar de manera efectiva los principios y prácticas de XP. Brinda orientación sobre cómo aplicar estas técnicas en el desarrollo diario.

Resolver problemas y conflictos: Cuando surgen problemas o conflictos en el equipo, el entrenador interviene para ayudar a encontrar soluciones y restablecer la armonía. Su experiencia es fundamental para superar obstáculos.

### **3.1.3 Tarjetas CRC (Clase-Responsabilidad-Colaboración).**

Las tarjetas CRC, que representan Clase-Responsabilidad-Colaboración, son una técnica utilizada en la metodología de Programación Extrema (XP) para modelar las interacciones entre las clases en un sistema de software. Estas tarjetas son una herramienta efectiva para el diseño y análisis de clases en un proyecto de desarrollo ágil. A continuación, se describe detalladamente en qué consisten y cómo se emplean en el desarrollo de software.

**Descripción de las tarjetas CRC:**

1. Clase: En una tarjeta CRC, la "Clase" se refiere a una entidad que agrupa datos y funciones relacionadas. Representa una abstracción de un objeto o entidad del mundo real en el sistema. Las clases son elementos fundamentales de la programación orientada a objetos y se utilizan para modelar las estructuras del sistema.

2. Responsabilidad: La sección de "Responsabilidad" en la tarjeta indica qué acciones o funcionalidades está encargada de realizar la clase. Cada clase tiene responsabilidades específicas que definen sus tareas y funciones en el sistema. Esto puede incluir la manipulación de datos, la ejecución de cálculos o la interacción con otras clases.

3. Colaboración: La sección de "Colaboración" identifica con qué otras clases interactúa la clase en cuestión. Esta colaboración refleja las conexiones y dependencias entre las clases. Las colaboraciones ayudan a comprender cómo se comunican las clases para lograr objetivos específicos del sistema.

**Utilización de las tarjetas CRC en el desarrollo de software:**

El proceso de creación y uso de tarjetas CRC en el desarrollo de software involucra los siguientes pasos:

1. Identificación de clases: El primer paso es identificar las clases necesarias para el sistema. Esto se hace a través del análisis de requisitos y la comprensión de las entidades y objetos involucrados en el dominio del problema.

2. Llenado de tarjetas: Para cada clase identificada, se crea una tarjeta CRC. En la tarjeta, se detalla el nombre de la clase en la sección "Clase". Luego, se enumeran las responsabilidades de la clase en la sección correspondiente. Esto implica especificar las funciones que realiza la clase, como cálculos, validaciones o manipulación de datos. Finalmente, en la sección de "Colaboración", se enumeran las clases con las que interactúa la clase en cuestión.

3. Sesiones de diseño: En las sesiones de diseño de XP, los miembros del equipo se reúnen y utilizan las tarjetas CRC para discutir y analizar la estructura del sistema. Estas tarjetas se utilizan para comprender cómo interactúan las clases y cómo se distribuyen las responsabilidades.

4. Iteración y mejora: A medida que el proyecto avanza, las tarjetas CRC pueden someterse a iteraciones y mejoras. Las tarjetas se actualizan para reflejar cambios en las clases, sus responsabilidades o colaboraciones a medida que se desarrolla el sistema.

5. Documentación y referencia: Las tarjetas CRC sirven como una valiosa fuente de documentación y referencia para el equipo de desarrollo. Facilitan la comunicación y el entendimiento de la estructura del sistema en todo el ciclo de vida del proyecto.

### **3.1.4 Historias de Usuario**

**Definición y Función de las Historias de Usuario en XP**

Las historias de usuario en el contexto de la Programación Extrema (XP) desempeñan un papel crucial al definir los requisitos del software desde la perspectiva del usuario. Estas historias representan características o funcionalidades específicas que un usuario final desea en el software. A través de su lenguaje claro y accesible, cada historia describe un solo elemento funcional, lo que facilita la comprensión tanto para los desarrolladores como para los usuarios.

**Funciones Clave:**

**Definición de Requisitos desde la Perspectiva del Usuario:**

Las historias de usuario ponen en el centro de atención las necesidades de los usuarios finales, garantizando que el software sea relevante y útil. Los requisitos se describen en términos de lo que un usuario necesita lograr, mejorando la comunicación entre el equipo de desarrollo y los stakeholders.

**Priorización y Planificación:**

Las historias de usuario permiten priorizar las características del software. Los usuarios o representantes del negocio pueden asignar prioridades a cada historia, lo que ayuda al equipo de desarrollo a determinar qué funcionalidades implementar primero.

**Estimación de Esfuerzo:**

Cada historia de usuario puede estimarse en términos de tiempo y recursos necesarios para su implementación. Esta estimación es necesaria para la planificación y asignación de tareas en el equipo.

**Seguimiento del Progreso:**

Las historias de usuario permiten un seguimiento detallado del progreso del proyecto. El equipo de desarrollo puede marcar cada historia a medida que se completa, lo que proporciona una vista clara de lo logrado y lo pendiente.

**Flexibilidad y Adaptación:**

Las historias de usuario fomentan la flexibilidad. Los requisitos pueden cambiar a medida que el proyecto avanza, y las historias de usuario permiten una adaptación ágil a estas modificaciones.

**Utilización en XP:**

El proceso de trabajo con historias de usuario generalmente involucra los siguientes pasos:

* **Captura de Historias de Usuario:** Los usuarios o representantes del negocio proporcionan historias de usuario al equipo de desarrollo. Estas historias deben ser claras y concisas, centradas en la funcionalidad que se desea.
* **Priorización:** El equipo y los stakeholders priorizan las historias de usuario según su importancia y valor para el usuario.
* **Estimación**: El equipo de desarrollo estima el esfuerzo necesario para implementar cada historia. Esto puede incluir la definición de las tareas específicas requeridas.
* **Implementación:** El equipo trabaja en la implementación de las historias de usuario en ciclos de desarrollo iterativos y cortos.
* **Pruebas y Validación:** Cada historia de usuario se somete a pruebas para asegurarse de que cumple con los requisitos y expectativas del usuario.
* **Entrega Incremental:** Las historias de usuario completadas se entregan al usuario final de forma incremental, lo que permite la obtención de retroalimentación temprana.

### **3.1.5 ¿Cómo utilizar esta metodología en este trabajo?**

El siguiente fragmento del documento describe la manera en la que se implementa la metodología XP con la propuesta planteada Kv-wims y las principales funcionalidades detectadas.

Al tratarse de un sistema que permita resolver una necesidad particular como lo es la gestión de almacén e inventario, es importante obtener exactamente cuáles serían las principales tareas para resolver por parte de los usuarios que van a adquirir el sistema y es entonces cuando la metodología tiene su primera etapa de implementación mediante el uso de historias de usuario y las tarjetas CRC.

Analizando y obteniendo las Historias de Usuario se tendrá el primer acercamiento a las necesidades específicas del cliente pues mediante una descripción detallada de lo que desea implementar se obtendrá la funcionalidad del sistema o, en otras palabras, vamos a obtener los requerimientos que el usuario desea visualizar en la implementación del producto final.

Por otra parte, las tarjetas CRC definen las funcionalidades del sistema desde la perspectiva del desarrollador y sirven como medio de comunicación con el cliente pues al detallarse en ellas cuál será el funcionamiento de algún requisito obtenido de las historias de usuario, el cliente podrá brindar una retroalimentación individual para cada funcionalidad y así los desarrolladores tendrán un control de los cambios que deben realizarse al sistema.

### **3.1.6 Productos esperados**

**¿Qué se hará en TT1?**

Durante el desarrollo de Trabajo Terminal 1, se trabajaron aspectos teóricos y del diseño del sistema, por lo que, al finalizar este periodo, los productos esperados, salvo la presentación para la exposición del trabajo, se encuentran dentro de este mismo documento técnico, el cual, a grandes rasgos está conformado por:

1. Introducción a la problemática: En este capítulo se indaga sobre la problemática principal que se desea abordar en el diseño y futuro desarrollo del sistema.
2. Estado del arte: Se realiza un estudio comparativo sobre las alternativas de sistemas existentes en el mercado los cuales son aplicables al problema descrito.
3. Marco teórico: En este capítulo, se introduce al lector a los conceptos importantes que están directamente relacionados con el diseño del sistema en capítulos subsecuentes.
4. Análisis del sistema: Se realiza una serie de análisis elementales para determinar el camino de la futura implementación del sistema; entre ellos: el estudio de factibilidad, análisis de riesgos, historias de usuario (para entender el funcionamiento del sistema desde el punto de vista del usuario final), y, requerimientos de la base de datos respecto a distintos aspectos.
5. Diseño del sistema: En este capítulo, se dan los primeros pasos respecto al apartado técnico del diseño del sistema para su futura implementación, mediante tarjetas CRC se describen las clases potenciales que existirán en el sistema, y, posteriormente, se muestran varios diagramas UML que aportan a la documentación y futuro desarrollo iterativo del proyecto.
6. Anexos: Por último, se agregan ciertas notas e información extra que es útil, aunque no necesariamente elemental para el entendimiento de este documento.

**¿Qué se hará en TT2?**

En Trabajo Terminal 2, se continua con el proceso de diseño y desarrollo iterativo de la metodología, por ende, se espera que, para el final de este periodo se cuente con la implementación del sistema, cuyas funcionalidades se pueden resumir en:

1. Módulo de autenticación de usuario.
2. Módulo de gestión de usuarios y controles de acceso.
3. Módulo de edición de productos (crear, consultar, editar, eliminar).
4. Módulo editor de anaqueles: permitirá crear clases de anaqueles según sus características (dimensiones, pisos), instanciarlos dentro del almacén y editar manualmente la disposición esperada de los productos.
5. Módulo visualizador de anaqueles: mostrará el orden esperado de los productos en el anaquel, su clase, nombre y características.
6. Editor visual de almacén: permitirá editar la distribución de anaqueles dentro del almacén, así como modificar las propiedades inherentes del almacén, como sus dimensiones y puntos de entrada y salida.
7. Módulo visualizador de almacén: mostrará la disposición de los anaqueles dentro del almacén.
8. Módulo de administración de inventario: permitirá registrar los cambios en existencias de los productos, así como programar y registrar reposiciones de inventario.
9. Módulo de búsqueda de productos: permitirá buscar la ubicación de un producto dentro del almacén y en el orden esperado en el anaquel, así como señalar sus existencias y otra información relevante, como la fecha más próxima a disponibilidad en caso de que no haya existencias.
10. Módulo de analítica y métricas de datos: basándose en la historia de datos generada por los módulos de administración de inventario, búsqueda de productos, editor de almacén y editor de anaqueles, ofrecerá información útil al usuario como clasificación de productos en el rango de más a menos vendidos, posibles ofertas que podrían implementarse, y otros datos estadísticos que se pueden obtener en base al análisis de datos históricos generados por el sistema.

## **3.4 Patrones de arquitectura aplicados**

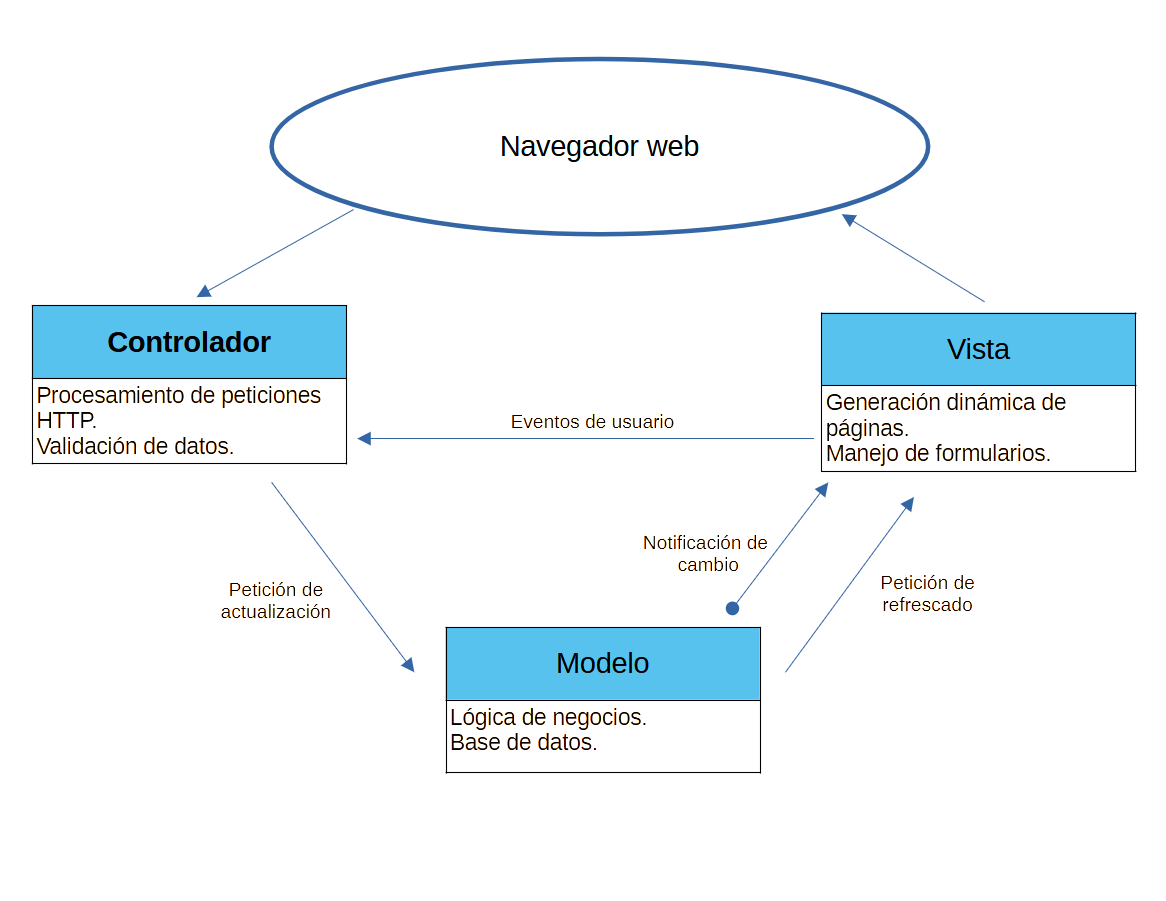
En esta sección del documento se detallan los 2 patrones de arquitectura sobre los cuales está basada la arquitectura general del sistema.

### **3.4.1 Patrón Modelo-Vista-Template**

El patrón de arquitectura Modelo-Vista-Template, es una variación del patrón tradicional conocido como Modelo-Vista-Controlador, el cual es la base de manejo de interacciones de gran parte de sistemas basados en web, tiene soporte en la mayoría de los lenguajes de programación. Esta arquitectura separa el sistema en 3 componentes; datos, interacción y presentación. Cada uno de estos componentes lógicos interactúan entre sí. [8]

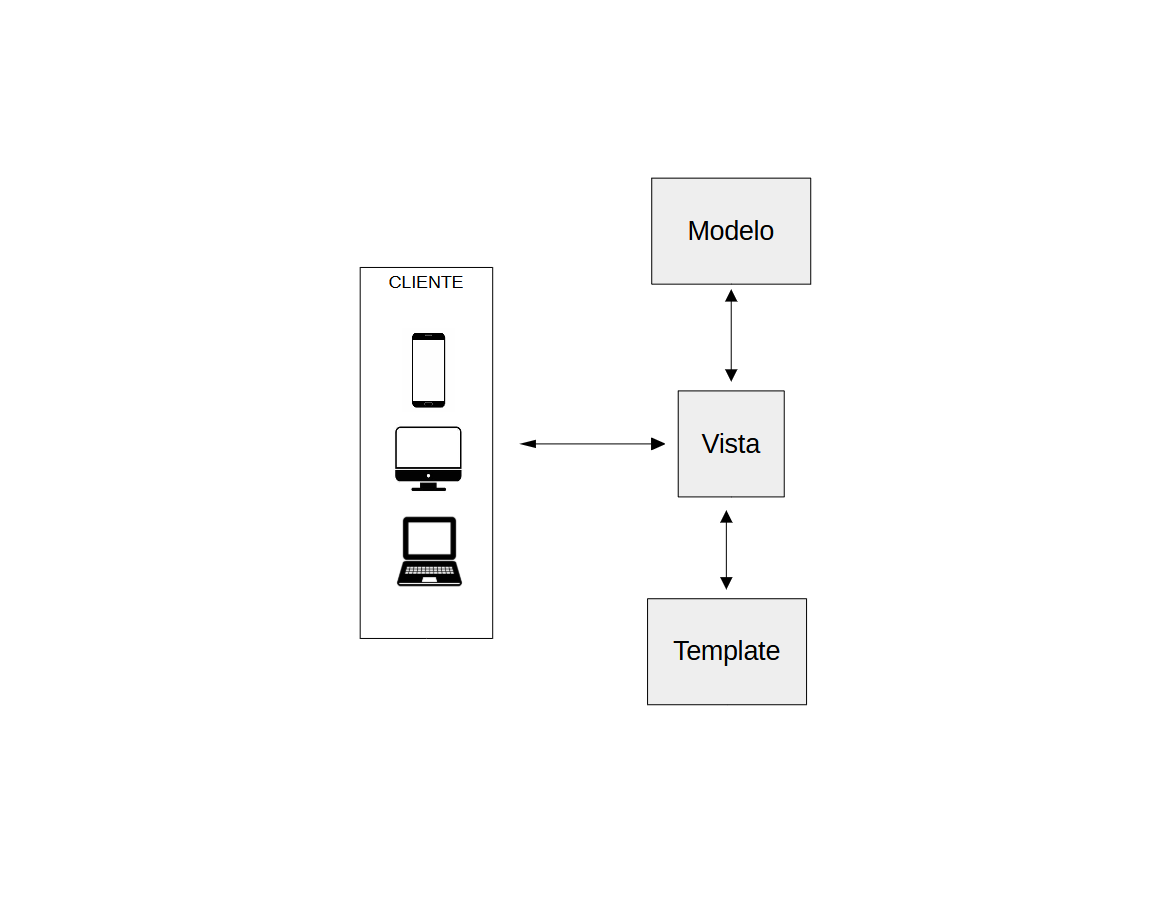
|  |  |
| --- | --- |
| **COMPONENTE** | **FUNCIÓN** |
| Modelo | Maneja los datos del sistema y las operaciones sobre estos datos, es decir, la persistencia de datos y la lógica y reglas del negocio. |
| Vista | Define y maneja como los datos son presentados al usuario, es decir, incluye la generación de páginas dinámicas. |
| Template | Maneja las interacciones y peticiones que el usuario hace mediante la componente vista. Algunas veces actualiza directamente la vista, sin embargo, su función principal es comunicarse con el componente modelo, para que este último actualice la componente vista. |

1. Componentes del patrón de arquitectura: modelo-vista-controlador



1. Patrón de arquitectura: modelo, vista, controlador

Por otro lado, el patrón Modelo-Vista-Template, modifica los roles del patrón MVC, en este caso, la vista tiene la primera interacción con una petición de cliente, esta toma los datos del usuario y se comunica con el modelo el cual se encarga del procesamiento de dichos datos, posteriormente cuando los datos han sido procesados por el modelo, la vista interactúa con una plantilla (template), la cual se encarga de la presentación estos datos mediante HTML y CSS [9]. Este patrón de arquitectura es aplicable en Kv-wims debido a que el servidor web Django, el cual se describe más adelante en este documento (vea capítulo 3.5) lo aplica.



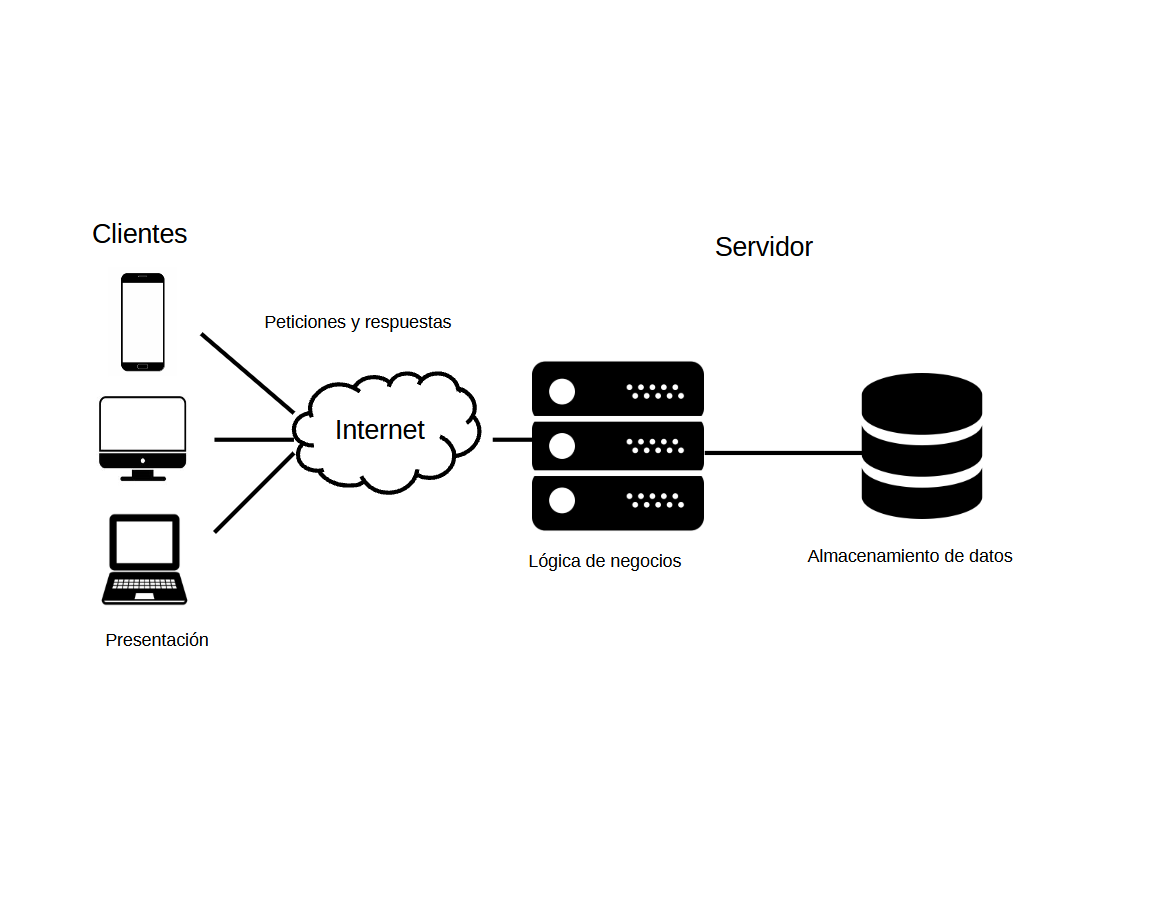
1. Patrón de arquitectura: modelo-vista-template

### **3.4.2 Patrón Cliente - Servidor**

Existen básicamente 2 tipos de arquitecturas cliente-servidor, de 2 niveles y de N-niveles, cada una de estas arquitecturas tiene sus variaciones, por ejemplo, en el caso de Kv-Wims, al contar con almacenamiento de datos, se puede considerar al menos como una arquitectura cliente-servidor de 3 niveles; Presentación, lógica de negocios y almacenamiento de datos.

De manera general, esta arquitectura se encarga de separar el procesamiento del sistema, el cual ocurre por lo menos en 2 procesos distintos y en computadoras distintas. Se habla de que un sistema está basado en esta arquitectura si la manipulación de datos y la presentación de datos están separados. [10]

A continuación, se muestra el diagrama general de arquitecturas cliente-servidor de 3 niveles:



1. Patrón de arquitectura: cliente servidor de 3 niveles

## **3.5 Python**

Python es un lenguaje de programación orientado a objetos, interpretado y de alto nivel de abstracción. Tomando la implementación de referencia de Python; CPython [11] ofrece varias herramientas que lo hacen atractivo para el desarrollo de aplicaciones de manera ágil:

* **Estructuras de datos de alto nivel:** CPython tiene varias estructuras de datos implementadas por defecto; cadenas de texto (*strings*), listas, diccionarios, conjuntos mutables e inmutables (*sets y frozensets respectivamente*), que facilitan la construcción de aplicaciones.
* **Sintaxis sencilla:** La sintaxis de Python es fácil de aprender y enfatiza la legibilidad de los programas mediante el uso de saltos de línea e identación de bloques. [12]
* **Módulos:** Los módulos en Python son simplemente un archivo que contiene definiciones y funciones que pueden ser importadas a otros archivos para reutilizarse, lo cual beneficia la modularidad y separación de tareas de una aplicación, así como la reutilización del código. [13]
* **Paquetes:** Esta funcionalidad de Python ayuda el escalamiento de proyectos permitiendo al desarrollador agrupar módulos mediante el sistema de archivos inherentemente jerárquico del sistema operativo. Usualmente se hace esto para clasificar archivos mediante funcionalidades similares y evitar colisiones en otros nombres de archivo [14].   
  Por último, el administrador de paquetes de Python, “*Python Package Manager*” (PIP), permite a los desarrolladores el acceso e instalación de paquetes externos hechos por otros desarrolladores u organizaciones [15]; Estos paquetes se instalan ya que contienen funcionalidades de las cuales el proyecto en desarrollo depende, de esta manera, se favorece el uso de herramientas existentes y probadas, las cuales aceleran el tiempo de desarrollo de la aplicación.
* **Ambientes virtuales:** Python tiene soporte nativo de ambientes virtuales, estos permiten la instalación de paquetes de manera aislada en algún directorio para una aplicación en particular. Los ambientes virtuales principalmente prevén la colisión de versiones de dependencias entre distintos proyectos mediante instalación de dependencias de manera local al proyecto en vez de instalación de paquetes globales al ecosistema Python [15]. Nuevamente se favorece la modularidad y portabilidad de código mediante el uso de ambientes virtuales (VENV) en Python.
* **Interprete:** Una de las características principales de Python es el hecho de que su implementación por defecto (CPython) ejecuta scripts de Python mediante un intérprete, es decir, en vez de compilar a código máquina, un script es compilado a Bytecode que es directamente ejecutado por el intérprete mediante llamadas a funciones en el lenguaje C. [11] Esta característica del lenguaje hace que los ciclos de edición-prueba-depuración sean más rápidos, ya que cuando el intérprete encuentre algún error al momento de ejecución, se lanza una excepción, que si no es atrapada, muestra un seguimiento de la pila de llamadas a funciones del script, lo cual facilita encontrar la fuente del error e incremente la productividad de los desarrolladores.
* **Tipado dinámico:** Otra característica destacable de Python es su sistema de tipado dinámico. Esto significa que el tipo de dato asociado a una variable puede cambiar durante la ejecución del programa. Si bien este sistema de tipado conlleva su propio conjunto de retos, en general esto favorece el desarrollo eficiente de aplicaciones haciendo el código más sencillo y legible. [12]

### **3.5.1 Django**

Django es un marco de desarrollo web de alto nivel escrito en Python que promueve un diseño limpio y pragmático. Fue desarrollado para manejar las complejidades del desarrollo web, permitiendo a los desarrolladores centrarse en la creación de componentes de su aplicación sin necesidad de reinventar la rueda. Está diseñado para facilitar la creación de aplicaciones web complejas y basadas en bases de datos. [16]

Algunas de las características notables de Django que favorecen el desarrollo de aplicaciones web de manera eficiente son:

* **Orientado a componentes:** Django promueve la reutilización de código a través de lo que denomina "componentes"; unidades autocontenidas que pueden incluirse en múltiples proyectos.
* **Creador de Modelos (Models)**: El creador de modelos de Django es uno de sus componentes más importantes, que proporciona una abstracción de alto nivel para definir la estructura de las bases de datos (esquemas de base de datos) utilizando clases de Python.
* **Sistema de Plantillas (Templates)**: Django ofrece un sistema de plantillas propio, que permite a los desarrolladores definir la presentación de una página web. Las plantillas de Django utilizan su propio lenguaje de marcado, que permite la inserción de código Python para la generación dinámica de HTML. Este sistema permite la creación de filtros y etiquetas personalizadas para ampliar su funcionalidad.
* **Vistas**: En Django, una vista es una función Python que toma una solicitud web y devuelve una respuesta. Este es el lugar donde se coloca la "lógica de negocio" de una aplicación. Las vistas interactúan con el modelo y la plantilla para entregar contenido, redirigir a otra página, manejar sesiones y cookies, y responder a las acciones de los usuarios. Las vistas pueden servir una página web, redirigir, manejar errores, procesar formularios y más.
* **Interfaz de administración generada automáticamente:** Django incluye una interfaz de administración que se genera dinámicamente a partir de la configuración de los modelos.
* **Soporte robusto de URL**: Django permite diseñar URL elegantes y fáciles de recordar, lo que es beneficioso tanto para la usabilidad como para la optimización de motores de búsqueda (SEO).
* **Seguridad**: Incluye medidas de seguridad integradas para ayudar a proteger la aplicación contra problemas como la falsificación de solicitudes entre sitios (CSRF), inyección de SQL.
* **Comunidad y Documentación**: Tiene una comunidad activa y una documentación exhaustiva, lo que hace que sea más fácil para los desarrolladores aprender y resolver problemas. [17]

### **3.5.2 Type hints**

Dada la naturaleza dinámica de Python, un desarrollador puede inferir el tipo de dato de una variable al leer su inicialización, sin embargo, ya que Python es un lenguaje interpretado, muchos errores no aparecerán hasta el momento en que se ejecuta el script. [18]

Las sugerencias de tipado (type hints) en Python proporcionan una manera de anotar código, en especial el tipo de dato asociado a una variable, a los parámetros de una función y también su valor de retorno. Una type hint consiste en una expresión opcional seguida del nombre del parámetro en una función, su lista de parámetros o el nombre de una variable. Estas anotaciones son completamente opcionales y Python no aplica ninguna regla basada en estas anotaciones al momento de compilar y ejecutar un script que contenga dichas anotaciones.

Las anotaciones en un script de Python no interfieren con la ejecución del programa en CPython, sin embargo, el uso asumido de las type hints consiste en que el desarrollador haga analice voluntariamente su script utilizando alguna herramienta de chequeo de tipado estático (static typechecking tool), o linter para encontrar los posibles errores de tipado cometidos. [19]

También se espera que los desarrolladores hagan uso del tipado gradual (gradual typing), es decir, anotar parcialmente un programa de Python y de esta manera aprovechen las ventajas del tipado dinámico y estático que ofrece el lenguaje de manera nativa y con extensiones de parte de terceros respectivamente. [20]

### **3.5.3 Otras bibliotecas**

En Python, varias bibliotecas clave facilitan la implementación de algoritmos para machine learning y análisis de datos, entre ellas: Scikit-learn es ampliamente usada para tareas generales análisis de datos, incluyendo clustering y árboles de decisión. Para series de tiempo, STUMPY y Prophet ofrecen herramientas especializadas, mientras que Statsmodels y SciPy son útiles para análisis estadísticos. Además, Pandas es esencial para la manipulación y preparación de datos. Estas bibliotecas, junto con otras disponibles en Python, ofrecen una amplia gama de opciones para adaptarse a las necesidades específicas de diferentes proyectos​ como lo podría ser en la futura implementación de Kv-wims. [21]

## **3.6 JavaScript**

JavaScript es un lenguaje de programación versátil y poderoso que se utiliza ampliamente en el desarrollo web y en otros contextos. Sus capacidades para crear aplicaciones interactivas, su amplia adopción y su sólido ecosistema de desarrollo son algunas de las principales fortalezas que lo convierten en una opción destacada para los desarrolladores [22].

Sus principales características y fortalezas clave incluyen:

* **Interactividad en el lado del cliente:** JavaScript permite a los desarrolladores crear aplicaciones web interactivas y dinámicas que responden a las acciones del usuario sin necesidad de recargar la página. Esto mejora significativamente la experiencia del usuario.
* **Amplia adopción:** JavaScript es uno de los lenguajes de programación más populares y ampliamente aceptados en el desarrollo web. Esto significa que hay una gran cantidad de recursos, bibliotecas y herramientas disponibles para los desarrolladores.
* **Compatibilidad con múltiples navegadores:** JavaScript es compatible con todos los navegadores web modernos, lo que garantiza que las aplicaciones funcionen de manera consistente en diferentes entornos.
* **Comunidad activa y soporte:** Existe una comunidad de desarrolladores activa que contribuye constantemente a la mejora del lenguaje y proporciona soluciones a los problemas comunes a través de foros, blogs y repositorios en línea.
* **Versatilidad:** JavaScript se utiliza no solo en el desarrollo web, sino también en el desarrollo de aplicaciones móviles, servidores (Node.js), aplicaciones de escritorio y más. Su versatilidad lo convierte en una herramienta poderosa en una variedad de contextos.
* **Librerías y frameworks:** JavaScript cuenta con numerosas librerías y frameworks, como React, Angular y Vue.js, que agilizan el desarrollo y facilitan la creación de aplicaciones web complejas y de alto rendimiento.
* **Ecosistema robusto:** El ecosistema de JavaScript incluye herramientas de prueba, administradores de paquetes (como npm), editores de código especializados y una variedad de recursos de aprendizaje, lo que facilita el desarrollo y la colaboración en proyectos [22].

## **3.7 HTML**

HTML es el lenguaje central para crear la estructura y el contenido de las páginas web. Su facilidad de aprendizaje, compatibilidad universal y capacidad de integración con otras tecnologías lo convierten en un pilar fundamental en el desarrollo web [22].

Las principales características y fortalezas clave de HTML incluyen:

* **Estructura y contenido:** HTML se centra en la estructura y el contenido de una página web, permitiendo a los desarrolladores definir elementos como encabezados, párrafos, listas y enlaces para organizar y presentar información de manera coherente.
* **Facilidad de aprendizaje:** HTML es un lenguaje relativamente sencillo de aprender y comprender, lo que lo convierte en una excelente opción para principiantes en desarrollo web.
* **Compatibilidad universal**: Todos los navegadores web admiten HTML, lo que garantiza que las páginas web creadas con HTML sean accesibles y visualizables en prácticamente cualquier dispositivo o plataforma.
* **Lenguaje de marcado extensible:** HTML es extensible y se puede combinar con otros lenguajes y tecnologías, como CSS (Cascading Style Sheets) y JavaScript, para mejorar la presentación y la interactividad de las páginas web.
* **Estándar de la web:** HTML es un estándar abierto, mantenido por el World Wide Web Consortium (W3C), lo que garantiza la consistencia y la interoperabilidad en la web.
* **Accesibilidad:** HTML proporciona etiquetas y atributos que permiten mejorar la accesibilidad web, lo que es primordial para llegar a un público diverso, incluidas personas con discapacidades.
* **SEO (Optimización de motores de búsqueda**): El uso adecuado de las etiquetas HTML permite a los motores de búsqueda comprender mejor el contenido de una página, lo que contribuye a una mejor clasificación en los resultados de búsqueda.
* **Gran cantidad de recursos y documentación:** Dado que HTML es fundamental en el desarrollo web, hay una abundancia de recursos, tutoriales y documentación disponibles en línea para ayudar a los desarrolladores a aprender y perfeccionar sus habilidades [22].

## **3.8 CSS**

CSS es una herramienta básica para el diseño web, ya que permite controlar la apariencia y el diseño de las páginas HTML. Su capacidad para separar el contenido del estilo, su flexibilidad en el diseño y su compatibilidad con múltiples navegadores lo convierten en una herramienta fundamental en el desarrollo web moderno [22].

Las principales características y fortalezas clave de CSS incluyen:

* **Separación de contenido y presentación:** CSS permite separar el contenido de una página web (definido en HTML) de su estilo y diseño. Esto hace que el mantenimiento y la actualización de un sitio web sean más sencillos y eficientes.
* **Estilo consistente y uniforme:** CSS permite aplicar estilos de manera coherente en un sitio web completo. Esto significa que puedes definir un conjunto de reglas de estilo y aplicarlas a todas las páginas, lo que garantiza una apariencia uniforme.
* **Flexibilidad en el diseño:** CSS ofrece una amplia variedad de propiedades y atributos para controlar el diseño de elementos web, como colores, fuentes, márgenes, tamaños y alineación. Esto permite una personalización completa del aspecto de una página web.
* **Medios de respuesta (responsive design):** CSS facilita la creación de diseños web adaptables, que se ajustan automáticamente a diferentes tamaños de pantalla y dispositivos, lo que mejora la experiencia del usuario en dispositivos móviles y de escritorio.
* **Eficiencia en la carga de páginas:** Al permitir la definición de estilos en un archivo CSS separado, las páginas web se cargan más rápido, ya que se reduce la duplicación de información de estilo en cada página.
* **Mantenimiento sencillo:** La modificación de estilos en un sitio web es más sencilla y rápida gracias a CSS. Cambiar un estilo en una hoja de estilo afecta a todas las páginas que lo utilizan.
* **Compatibilidad con múltiples navegadores:** CSS es ampliamente compatible con todos los navegadores modernos, lo que garantiza que los estilos se apliquen de manera consistente en diferentes entornos.
* **Extensibilidad y preprocesadores:** CSS es extensible a través de preprocesadores como SASS y LESS, que permiten escribir estilos de manera más eficiente y con características adicionales [22].

## **3.9 Oracle-MySQL**

MySQL es un sistema de gestión de base de datos de código abierto, es desarrollado, mantenido y distribuido por Oracle Coorporation. Las características destacables de esta tecnología son:

* **Bases de dato relacionales:** Una base de datos relacional almacena datos en tablas separadas en lugar de poner todos los datos en un único archivo masivo. Las estructuras de la base de datos están organizadas en archivos físicos optimizados para la velocidad de procesamiento. El modelo lógico, que contiene objetos como bases de datos, tablas, vistas, filas y columnas, ofrece un entorno de programación flexible mediante la implementación del estándar SQL. El desarrollador puede establecer reglas que rigen las relaciones entre diferentes campos de datos, como la cardinalidad entre relaciones, campos únicos, requeridos u opcionales. El sistema gestor hace cumplir estas reglas, de modo que, si la base de datos está bien diseñada, la aplicación no verá datos inconsistentes, duplicados, huérfanos, desactualizados o faltantes.
* **El software MySQL es código abierto (Open Source):** El término Open Source significa que es posible que cualquiera utilice y modifique el software. Cualquiera puede descargar MySQL de Internet y usarlo sin pagar nada. El software Open Source ofrece varias ventajas sobre el software propietario:
  + El costo de este suele ser gratuito.
  + Soporte comunitario, estabilidad, longevidad y soporte: Las comunidades de software abierto son muy activas, lo cual propicia los ciclos de actualización continuos y por extensión, la estabilidad y longevidad de estos sistemas, así como el soporte que ofrecen los foros y documentación de este a desarrolladores que desean integrar esta tecnología a sus aplicaciones.
* **Servidor MySQL rápido, fiable, escalable y fácil de usar:** El Servidor MySQL puede funcionar cómodamente en computadoras de escritorio, portátil, o sobre cualquier sistema operativo, como es el caso de las máquinas virtuales que los servicios de hosting o nube ofrecen.
* **MySQL funciona en sistemas cliente-servidor­:** El Software de Base de Datos MySQL es un sistema cliente/servidor que consta de un servidor SQL multihilo que soporta diferentes back ends, varios programas y bibliotecas para el cliente, como los drivers para distintos lenguajes de programación host, herramientas administrativas y una amplia gama de interfaces de programación de aplicaciones (APIs).
* **Disponibilidad de software contribuido a MySQL:** El Servidor MySQL tiene características desarrolladas en estrecha colaboración con la comunidad. Gran cantidad de aplicaciones y lenguajes de programación tienen soporte para MySQL. [23]

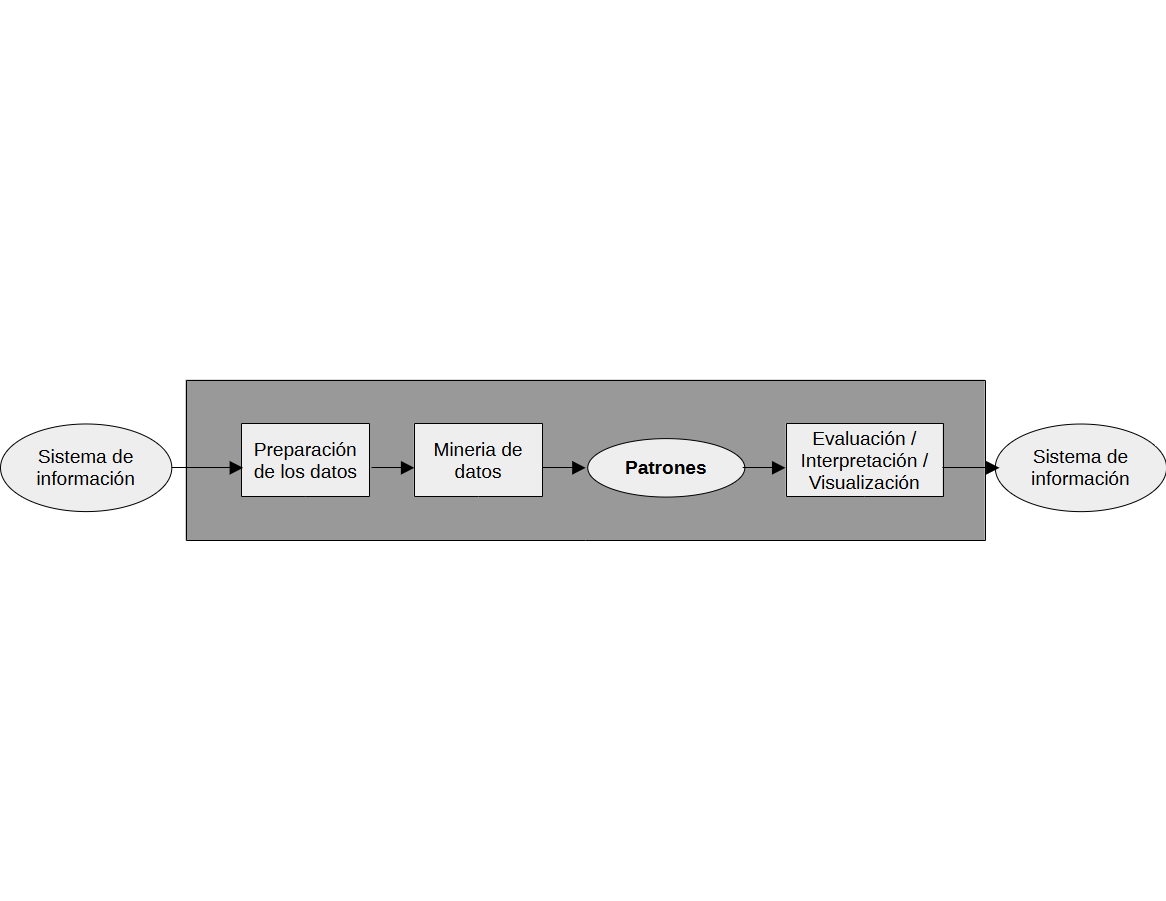
## **3.10 Minería de datos y proceso de descubrimiento de conocimiento en bases de datos (KDD)**

Dentro de la ciencia de datos existen varios conceptos que suelen ser utilizados por la comunidad de manera intercambiable, si bien estos pueden estar estrechamente relacionados, la realidad es que existen claras diferencias entre ambos, como es el caso de los conceptos “data mining” y “proceso de descubrimiento de conocimiento en bases de datos (KDD)”. El termino KDD se utiliza para referirse a un proceso que consta de una serie de fases, donde la minería de datos es solo una de esas fases.

Específicamente, el KDD es el proceso no trivial de identificar patrones válidos, novedosos, potencialmente útiles y ultimadamente comprensibles a partir de los datos. Partiendo de esta definición, podemos destacar las propiedades del conocimiento extraído:

* **Válido:** Se refiere a que los patrones y modelos obtenidos deben tener cierto grado de certidumbre al utilizarse con nuevos datos y no solo con los datos que fueron usados para su obtención.
* **Novedoso:** Se debe obtener información desconocida para el sistema y preferentemente para los usuarios de este.
* **Potencialmente útil:** La información obtenida debe conducir a acciones que reporten beneficios a los usuarios del sistema.
* **Comprensibilidad:** Si la información extraída es incomprensible, se dificulta su interpretación, revisión y validación, por lo cual esta información no proporciona conocimiento desde el punto de vista de su utilidad.

De lo anterior, se puede deducir que el KDD es un proceso complejo que no solo incluye la obtención de modelos y patrones, sino también la evaluación, interpretación e incluso visualización de estos. La siguiente figura ilustra de manera general el proceso de KDD. [24]





### **3.10.1 La minería de datos y otras disciplinas**

El campo de la minería de datos es multidisciplinar y se ha desarrollado en paralelo o como extensión con otras tecnologías. La investigación y avances en la minería de datos se nutren de los avances en estas otras áreas. Además, es pertinente hacer notar que el uso de la minería de datos o el proceso KDD en el sistema Kv-wims, se extiende a algunas de estas otras disciplinas:

* **Bases de datos:** conceptos como los almacenes de datos y el procesamiento analítico en línea (OLAP) tienen una gran relación con la minería de datos, aunque en este último caso no se trata de obtener informes avanzados a base de agregar los datos de cierta manera compleja pero predefinida (como incluyen muchas herramientas de inteligencia de negocios, presentes en sistemas de gestión de bases de datos comerciales).
* **Estadística:** esta disciplina ha proporcionado muchos de los conceptos, algoritmos y técnicas que se utilizan en minería de datos, incluso, algunos paquetes de análisis estadístico se comercializan como herramientas de minería de datos.
* **Aprendizaje automático:** ésta es el área de la inteligencia artificial que se ocupa de desarrollar algoritmos supervisados o no, capaces de aprender, y constituye, junto con la estadística, el corazón del análisis inteligente de los datos. El principio seguido se basa simplemente en que la computadora aprenda un modelo a partir de ejemplos y este modelo se utilice posteriormente para resolver un problema.
* **Sistemas para la toma de decisión:** son herramientas y sistemas informatizados que asisten a los directivos en la resolución de problemas y en la toma de decisiones. El objetivo es proporcionar la información necesaria para realizar decisiones efectivas en el ámbito empresarial o en tareas de diagnóstico.
* **Visualización de datos:** el uso de técnicas de visualización permite al usuario descubrir, intuir o entender patrones que serían más difíciles de “ver” a partir de descripciones matemáticas o textuales de los resultados. Existen técnicas de visualización, como, por ejemplo, las gráficas (diagramas de barras, gráficas de dispersión, histogramas, etc.), las icónicas (basadas en figuras, colores, etc.), las basadas en píxeles (cada dato se representa como un único píxel), las jerárquicas (dividiendo el área de representación en regiones dependiendo de los datos) y muchas otras. [24]

La siguiente figura muestra las disciplinas más relevantes que se integran a la minería de datos, además, en negritas se muestran aquellas que consideramos más relacionadas al desarrollo del sistema Kv-wims.





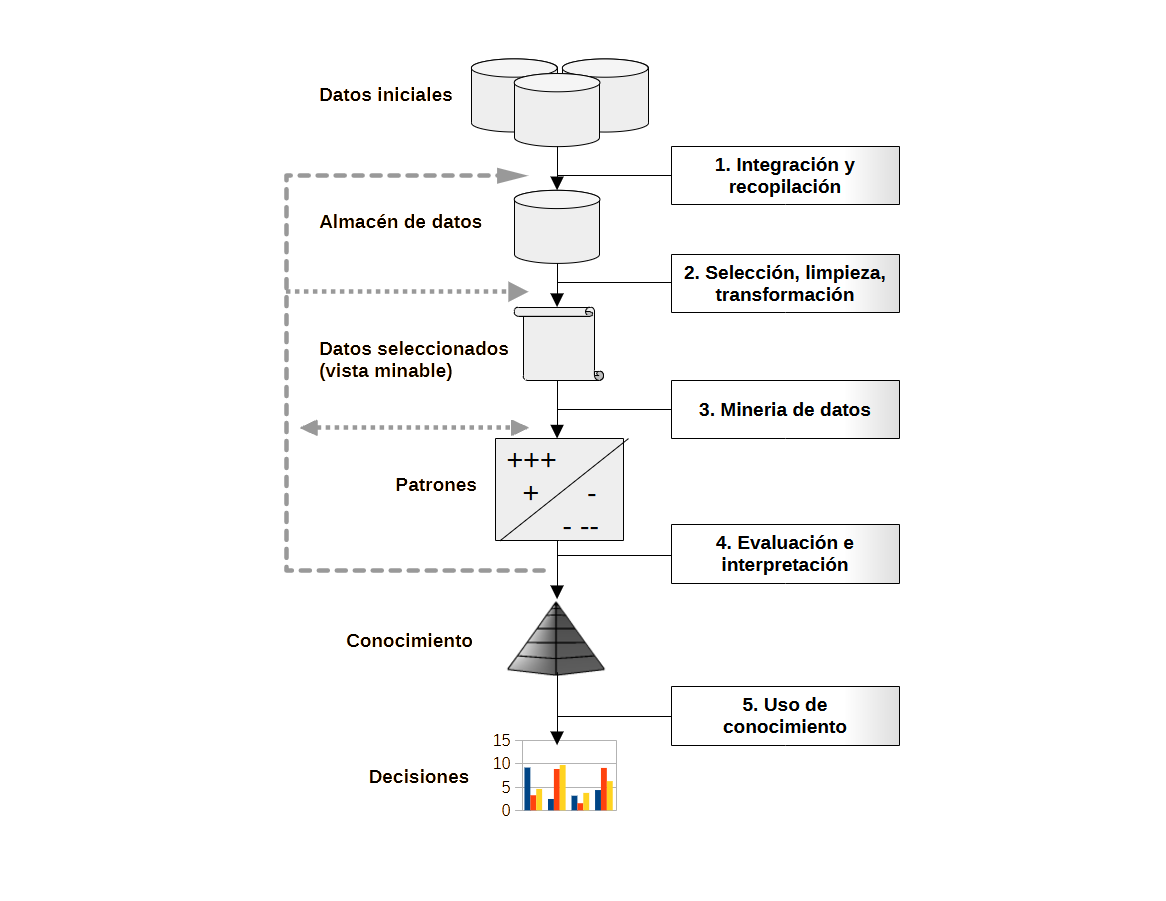
### **3.10.2 Tipos de modelos de minería de datos**

La minería de datos busca escudriñar los datos para descubrir información útil. Esta información puede presentarse en forma de patrones, relaciones o reglas que no se conocían antes, o simplemente como un resumen que da una vista general. Estos patrones o resúmenes son lo que se consideran el modelo de los datos que hemos analizado.

Existen dos tipos principales de modelos en la minería de datos: predictivos y descriptivos. Los modelos predictivos se enfocan en prever valores futuros o no conocidos de ciertas variables de interés, conocidas como variables objetivo o dependientes, mediante el uso de otras variables llamadas independientes o predictivas. En contraste, los modelos descriptivos tienen como finalidad identificar patrones que proporcionan una explicación o un resumen de los datos existentes, y se utilizan para examinar las características de los datos, no para hacer predicciones sobre ellos. [24]

### **3.10.3 Fases del proceso de KDD**

El proceso de KDD se organiza en torno a 5 fases, como se describe en la siguiente figura. El proceso en general se considera iterativo e interactivo. Es iterativo ya que alguna de las fases puede volver a pasos anteriores e interactivo ya que el usuario o expertos en el área pueden ayudar con la validación de resultados o en la toma de decisiones a partir de estos.





Las bases de datos y aplicaciones basadas en el procesamiento tradicional de datos, conocidas como procesamiento transaccional en línea (OLTP, on-line transaction processing) suelen ser suficientes para cubrir las necesidades del día a día de las empresas (tales como la facturación, control de inventario y otras), sin embargo, esa misma herramienta puede resultar ineficiente para otras tareas más complejas como el análisis, la planificación y la predicción. Es decir, decisiones estratégicas a largo plazo, es por eso que la existencia de herramientas que además de proporcionar procesamiento de transacciones en línea, proporcionen KDD, agregan un gran valor gracias a la interactividad que el usuario puede tener con estas para la toma de decisiones de su negocio. A continuación, se describe a grandes rasgos de que trata cada fase del proceso de KDD:

* **Integración y recopilación de datos:** Usualmente esta fase consiste en la unificación de diversas bases de datos existentes para distintos departamentos o lugares de una corporación en un solo almacén de datos con un formato o estándar común para facilitar el procesamiento posterior. En casos donde el volumen de datos no es masivo, como en las tiendas de conveniencia, se puede trabajar con los datos originales e incluso con datos en formatos heterogéneos.
* **Selección, limpieza y transformación:** La calidad del conocimiento obtenido depende de más factores además del algoritmo utilizado, en especial de la calidad de los datos siendo minados. Esta etapa consiste en preparar la vista minable a partir del conjunto de datos original. Este proceso es necesario ya que algunos datos de la etapa anterior pueden ser irrelevantes o innecesarios. Los siguientes puntos son una guía general de que partes de los datos se pueden modificar para que estos puedan ser procesados por algoritmos de minado:
  + **Datos anómalos (outliers):** estos datos son aquellos que no se ajustan, o no son muy parecidos a el resto de datos, es decir son casos límite, y en general depende del algoritmo que se desee utilizar posteriormente con estos datos para decidir qué hacer con estos datos, ya sea omitirlos (varios algoritmos hacen esto) por que no son de mayor interés y podrían afectar el resultado del modelo según su sensibilidad, o por el contrario, este tipo de datos pueden ser de utilidad para otros modelos que buscan extraer otro tipo de información y por lo tanto no es necesario ignorarlos.
  + **Datos faltantes (missing values):** Este tipo de tuplas o datos pueden llevar a resultados pocos precisos por parte del modelo, sin embargo, siempre es útil considerar el significado real de estos datos faltantes, es decir qué tipo de error pudo haberlos producido, ya que, en el aspecto práctico del uso de un sistema, pueden existir diversas causas para el por qué se registraron datos faltantes.
  + **Selección de atributos relevantes:** Este es uno de los preprocesamientos a datos más importantes, ya que es crucial determinar que columnas o atributos de los datos son relevantes para la tarea de minería.
  + **Construcción de atributos:** Esta tarea consiste en construir nuevos atributos a partir de aplicar una función u operación a los atributos ya existentes, este proceso se motiva en el hecho de que algunas veces los atributos originales no tienen suficiente poder predictivo por si solos y de cierta manera hace falta combinarlos para obtener una nueva medida que es más descriptiva por si solo que como dos o más atributos independientes.
  + **Codificación:** Es posible que en algunos casos sea necesario codificar atributos categóricos en otro formato, por ejemplo, una numeración entera. Por otro lado, existe también la discretización, que consiste en tomar una lista de intervalos desde los atributos originales y generar una serie de clases que representen dichos intervalos.
* **Minería de datos:** Esta fase es la más característica del proceso completo de KDD, el objetivo en esta fase es producir nuevo conocimiento para que el usuario pueda hacer uso de este. En esta fase también se realiza una partición en el conjunto de datos, donde una parte se utilizará para entrenar al algoritmo escogido; el conjunto de datos de entrenamiento (training set), y otra parte se utilizará en la fase siguiente para probar determinar la calidad del modelo generado; el conjunto de datos de prueba (test set). Es importante mencionar que también existen varias técnicas para realizar esta partición de manera adecuada. Por último, existen tres pasos generales para seleccionar un algoritmo adecuado a la tarea que se quiere resolver:
  + **Determinar el tipo de tarea de minería:** Se puede partir con una descripción de la tarea a realizar en términos prácticos.
  + **Elegir el tipo de modelo:** A partir de la tarea anterior se intenta dar con el tipo de modelo adecuado para la tarea: predictivo, descriptivo, o algo más específico, como clasificar, identificar patrones, agrupamiento, etc.
  + **Elegir el algoritmo de minería que resuelve la tarea:** A partir de las dos tareas anteriores se puede decidir que algoritmo en específico. Existen muchos algoritmos para una mismo o similar propósito, es por ello por lo que es pertinente elegir según las características de los datos y la tarea que se espera resolver.
* **Evaluación e interpretación:** La separación previa del conjunto de datos en dos; el training set y test set es necesaria para garantizar que la precisión del modelo es una medida independiente y no sufre de alguna sobre estimación (que tienda a ser muy optimista por no haber particionado el conjunto de datos original). La técnica de evaluación utilizada depende generalmente del tipo de modelo, aunque también hay técnicas específicas para algoritmos en particular, a continuación, se mencionan las técnicas de evaluación comunes para algunos tipos de algoritmos de uso frecuente:
  + **Clasificación**, lo normal es evaluar la calidad de los patrones encontrados con respecto a su precisión predictiva, la cual se calcula como el número de instancias del conjunto de prueba clasificadas correctamente dividido por el número de instancias totales en el conjunto de prueba.
  + **Reglas de asociación**, se suele evaluar de forma separada cada una de las reglas con objeto de restringirnos a aquellas que pueden aplicarse a un mayor número de instancias y que tienen una precisión relativamente alta sobre estas instancias. Esto se hace en base a dos conceptos:
    - Cobertura (también referida como soporte): número de instancias a las que la regla se aplica y predice correctamente.
    - Confianza: proporción de instancias que la regla predice correctamente, es decir, la cobertura dividida por el número de instancias a las que se puede aplicar la regla.
  + **Regresión**: la salida de los modelos de regresión es un valor numérico, la manera más habitual de evaluar un modelo es mediante el error cuadrático medio del valor predicho respecto al valor que se utiliza como validación. Esto promedia los errores y tiene más en cuenta aquellos errores que se desvían más del valor predicho.
  + **Agrupamiento**: las medidas de evaluación suelen depender del método utilizado, aunque suelen ser función de la cohesión de cada grupo y de la separación entre grupos. La cohesión y separación entre grupos se puede formalizar, por ejemplo, utilizando la distancia media al centro del grupo de los miembros de un grupo y la distancia media entre grupos, respectivamente. El concepto de distancia y de densidad son dos aspectos cruciales tanto en la construcción de modelos de agrupamiento como en su evaluación.
  + Además de medidas puramente objetivas, como las descritas, también existen medidas subjetivas, como el interés, novedad, simplicidad o comprensibilidad.
* **Uso de datos y monitorización:** Una vez construido y validado el modelo, se puede utilizar con dos finalidades en el contexto organizacional; Primero, que un analista recomiendo acciones basándose en el modelo. La segunda alternativa es integrar el modelo en una aplicación para que este oriente a los usuarios del sistema en la toma de decisiones (esto dependerá de su capacitación y otras medidas subjetivas como las mencionadas en el punto anterior). [24]

### **3.10.4 Conjunto de datos (Dataset)**

A continuación, se presentan algunas estrategias para la recopilación y aumento de datos (en caso de ser necesario posteriormente durante el desarrollo y fase de pruebas del proyecto).

El termino dataset es bien conocido en el área de la computación y este se entiende como una colección de datos, por lo general presentados de manera tabular, donde las columnas son las variables, y cada renglón representa un registro dentro de esta colección.

Gran parte de estas colecciones de datos surgen a partir del registro de transacciones que realizan la mayoría de las empresas y organizaciones mediante las herramientas OLTP que utilizan en su día a día, en este caso en particular, el sistema que la tienda de conveniencia utiliza para su administración de inventario y almacén.

El principal reto que existe en este caso de estudio es el hecho de que la mayoría de las tiendas de conveniencia en México, para las cuales se propone nuestra solución, no solo cuentan con sus propios sistemas OLTP propietarios, donde, es probable que su formato de almacenaje de datos no sea compatible con el nuestro, sino que aun más importante, estas empresas se reservan el derecho a hacer públicos sus datos de ventas, re-stock y logística distribución de productos en sus tiendas. En resumen, como desarrolladores del sistema propuesto en el presente documento, no tenemos acceso a información valiosa y precisa generada por tiendas de conveniencia actuales, la cual es de evidente utilidad para la futura fase de pruebas en el proyecto. Por esta razón, a continuación, se proponen algunas estrategias para generar un conjunto de datos, que si bien, no nos proporcionará la precisión y volumen que un dataset original de alguna de estas tiendas haría, nos ayudará cuando menos a futuro durante la fase de pruebas del proyecto.

### **3.10.5 Instrumento de investigación**

Un instrumento de investigación es una herramienta utilizada para recolectar, medir, y analizar datos relacionados a un interés particular (realizar pruebas de software en este caso). Estos instrumentos pueden incluir entrevistas, pruebas, observaciones o encuestas. A continuación, se mencionan dos instrumentos relevantes para la solución a nuestro problema de dataset.

* **Encuestas:** Engloban cualquier medida tomada que involucre realizar preguntas a los participantes. Las encuestas son de distintos tipos según el periodo de tiempo en que se realicen, para nuestro caso, son relevantes las encuestas de tipo transversal, donde se recolectan datos de distintos individuos en un punto específico o por un periodo no muy largo de tiempo. La utilidad de esto radica en que será posible generar una encuesta mediante la cual podamos recopilar información sobre el tipo de productos que las personas comprar en tiendas de conveniencia como Oxxo, 3B, y otras.
* **Observaciones:** Se refiere al hecho de recopilar datos mediante nuestros sentidos humanos. Los datos recolectados pueden ser cualitativos o cuantitativos. La observación natural es relevante a nuestro caso, ya que, en esta, simplemente se anotan datos sin interferir en el ambiente en el que existen. En nuestro caso, las observaciones nos servirán para obtener una idea de la distribución física usual que tienen los productos en tiendas de conveniencia, esto incluye el numero de anaqueles visibles, como se agrupan los productos, y que lugares tienen mayor y menor visibilidad desde la entrada de la tienda. [25]–[27]

Mediante estos instrumentos de investigación, podremos recolectar datos suficientes, que, al ser complementados, nos podrán ofrecer una colección de información más apta para las futuras pruebas del software.

### **3.10.6 Otras fuentes de datos**

Otro factor que integrar en la colección de datos es, clasificaciones y precio de productos. Existen algunas bases de datos públicas que ofrecen esta información adicional, por ejemplo, el INEGI ofrece mediante su portal web la descarga y consulta de la base de datos de precios promedio [28]; Esta información adicional es útil para algoritmos como clasificadores y el cálculo de valores de ventas para algoritmos como pronósticos con series de tiempo.

### **3.10.7 Aumento de datos**

El aumento de datos es una técnica que consiste en incrementar artificialmente el tamaño de un dataset mediante la creación de nuevas entradas basadas en modificaciones a los registros originales del dataset. Este concepto es comúnmente utilizado en el campo de aprendizaje profundo (Deep learning), en especial para modelos referentes a clasificadores de imágenes, sin embargo, no se limita a ese tipo de datos, sino que también puede ser aplicado a datos de audio, texto, video, o datos tabulares, los cuales son especialmente relevantes para este trabajo.

El aumento de datos es aconsejable en ciertos casos, por ejemplo: para prevenir el sobreajuste de un modelo, mejorar la precisión del modelo o cuando el conjunto de datos iniciales es algo pequeño, (las últimas dos situaciones son aplicables en nuestro caso, ya que no podemos estar seguros de que el volumen de datos obtenidos mediante nuestros instrumentos de investigación sea suficientemente grande, lo cual podría llevar también a un conjunto de datos desbalanceado, es decir, cuando alguna clase dentro del conjunto de datos ocupa un porcentaje muy bajo relativo a otras clases. Un ejemplo de desbalance extremo es cuando una clase ocupa solo el 1% de datos de la colección).

Si bien el aumento de datos ofrece una solución a la problemática de volumen y desbalance de clases, esta también viene con sus propios retos y limitaciones a considerar; El aumento de datos para colecciones tabulares no está del todo bien establecido, debido a la estructura y complejidad únicos a este tipo de datos. [29]–[32]

En resumen, el aumento de datos es un campo relativamente nuevo para la ciencia de datos, el propósito principal de estas técnicas es derivar nuevos datos a partir de datos existentes, su principal utilidad es el generar conjuntos de datos robustos para algoritmos como clasificadores o análisis predictivos. Si bien este campo es relativamente pequeño aún, es posible encontrar ya una variedad de técnicas en diversas fuentes que varían en complejidad e implementación, por ejemplo: muestreos sobre distribuciones de probabilidad conjuntas, modelos de Markov ocultos, Arboles de regresión, redes Bayesianas, y métodos aleatorizados o semi aleatorizados como SMOTE (Synthetic Minority Over-sampling Technique). [33]–[35]

### **3.10.8 Algoritmos utilizados**

A continuación, se describe a grandes rasgos que algoritmos se utilizarán para el módulo de analítica de datos de nuestro software.

### **3.10.9 Reglas de asociación para generar ofertas**

Las reglas de asociación son herramientas cruciales en el análisis de datos y la minería de datos que permiten identificar relaciones interesantes entre variables en grandes bases de datos. Una regla de asociación sugiere que, si un determinado conjunto de elementos aparece con cierta frecuencia, es probable que ocurra otro conjunto de elementos asociado al primero. Este conocimiento es valioso en el ámbito comercial de las tiendas de conveniencia para la generación de ofertas, ya que pueden revelar patrones de compra de los consumidores y a partir de estos, se pueden tomar decisiones de marketing o colocación de productos; Es entonces destacable su uso en tiendas de conveniencia, donde se busca la generación de ingresos mediante distintas estrategias, como lo son las ofertas.

En el contexto de la generación de ofertas, las reglas de asociación se pueden utilizar para crear promociones cruzadas basadas en la probabilidad de que los clientes que compren un producto estén también interesados en la compra de otro producto similar o no.

Una regla de asociación es una proposición probabilística sobre la ocurrencia de ciertos estados en una base de datos, se suele trabajaron con dos medidas para conocer la calidad de una regla:

* **Soporte:** El número de instancias que la regla predice correctamente.
* **Confianza:** Mide el porcentaje de veces que la regla se cumple cuando se puede aplicar.

Existen distintas familias de reglas de asociación clasificadas por ciertos criterios como:

* **Tipos de los valores utilizados en las reglas:** podemos tener reglas que trabajan con atributos binarios que indican la presencia o ausencia de un ítem, este el caso típico de la cesta de la compra.
* **Dimensiones de los datos:** Además de el atributo de existencia, pueden existir otros atributos, como fechas, nombre de clientes, etc.
* **instantáneas y secuenciales:** Depende de si se consideran relaciones en un instante (una compra, por ejemplo), o una secuencia o serie (múltiples compras a lo largo de un periodo de tiempo).

El algoritmo Apriori es uno de los métodos más conocidos para generar reglas de asociación. Funciona identificando primero los conjuntos de elementos frecuentes en la base de datos y luego usando esos conjuntos para construir reglas de asociación que satisfagan un umbral mínimo de confianza. [24]

### 

### **3.10.10 Árbol de decisión para clasificar nuevos productos**

Los árboles de decisión son una técnica de aprendizaje supervisado dentro de la minería de datos. Estos son modelos predictivos que simulan una serie de decisiones tomadas de manera secuencial, que pueden conducir a una conclusión específica. De manera más específica, un árbol de decisión es una estructura de árbol que particiona de manera recursiva el espacio de características en subespacios más homogéneos.

El nodo raíz del árbol representa la totalidad del espacio de muestras. A partir de ahí, se realizan divisiones binarias basadas en reglas de decisión que involucran los atributos de entrada. Estas reglas se generan maximizando una función objetivo que mide la pureza de los nodos resultantes, como la ganancia de información, basada en la entropía, o el índice de Gini, que mide la impureza de una distribución de probabilidad.

Algunos de los algoritmos más conocidos para la creación de árboles de decisión son:

* **ID3:** Utiliza la entropía como medida de impureza y la ganancia de información para seleccionar el atributo que mejor divide el conjunto de datos
* **C4.5:** Extiende ID3 permitiendo atributos con valores continuos. Implementa una técnica de podado de árbol para eliminar ramas que tienen poco poder predictivo, reduciendo así la complejidad del modelo final y mejorando su capacidad de generalización.
* **CART:** Produce árboles binarios utilizando el criterio de impureza del índice de Gini o la reducción de la varianza para la selección de características y la creación de reglas de decisión.
* **Random Forest:** En este caso se tiene un conjunto de árboles de decisión (forest), cada uno construido sobre una muestra aleatoria del conjunto de datos. La decisión final se toma por votación mayoritaria, lo que mejora la precisión y robustez del modelo.

Cada uno de estos algoritmos puede manejar diferentes tipos de problemas de clasificación, así mismo, tienen sus propias ventajas y limitaciones en términos de manejo de datos no lineales, la necesidad de preprocesamiento de datos. [24]

Los árboles de decisión pueden ser altamente útiles para clasificar nuevos productos entrantes en una tienda de conveniencia, por ejemplo, cuando llegan nuevos productos o lanzamientos, el árbol de decisión puede automáticamente asignarlos a la categoría adecuada basándose en varios atributos como tipo de producto, marca, precio, y tamaño. Esto agiliza el proceso de introducción de nuevos inventarios en el sistema.

### 

### **3.10.11 Pronósticos con series de tiempo ganancias y ciclos de restock**

Los pronósticos mediante series de tiempo ocurren cuando se hacen predicciones científicas basadas en datos que contienen información temporal, como fechas o etiquetas de tiempo (timestamps). Involucra la construcción de modelos mediante el análisis histórico de datos para realizar observaciones y orientar estratégicamente las decisiones que se tomarán a futuro. Una parte distintiva del pronóstico mediante series de tiempo es que los datos a futuro al momento de trabajarlas son completamente desconocidos, y serán estimados por los modelos a realizar, es por eso por lo que se considera una técnica predictiva.

Las predicciones realizadas por este tipo de modelos no son siempre exactas y no representan el desenlace real a futuro, además de que la probabilidad de que dichos escenarios ocurran puede variar significativamente, [36] sin embargo, se hace obvio la utilidad que tiene esta información en casos económicos ligados a toma de decisiones, como el caso de tiendas de conveniencia, donde puede ser útil aproximar las ventas que un producto generará en la siguiente quincena, donde por extensión, otra información útil se puede extraer de aquí, como cuando se deberá entonces hacer restock de dicho producto, es decir este modelo no mutuamente excluyente con otras partes del sistema, como reglas estáticas para avisos de restock cuando este cae bajo cierto nivel.

Existen dos tipos generales de métodos de pronóstico, cualitativo y cuantitativo. El pronóstico cualitativo suelen ser decisiones tomadas por humanos según su experiencia, y no es necesaria la existencia de datos minados, por otro lado, el pronóstico cuantitativo se realiza el análisis de datos existentes utilizando algoritmos.

Para hacer los resultados de los modelos más precisos, hay un par de reglas generales:

* **Regla de frecuencia:** Es muy aconsejable que los datos se actualicen constantemente, esto, para capturar cualquier información nueva que refleje cambios en tendencias.
* **Regla de horizonte:** Es recomendable evitar realizar predicciones muy adelante en el futuro, se recomiendan periodos de tiempo más cortos para realizar los pronósticos.

Los datos para series de tiempo pueden ser descompuestos en los siguientes componentes:

* **Nivel:** Cualquier serie temporal tendrá una línea base; el nivel. A esta línea base le añadimos diferentes componentes para formar una serie temporal completa.
* **Tendencia:** Define si, durante un período, una serie temporal aumenta o disminuye. Es decir, tiene una tendencia ascendente (incremento) o descendente (decremento).
* **Estacionalidad:** Define un patrón que se repite a lo largo de un período. Este patrón que se repite periódicamente se denomina estacionalidad.
* **Ciclicidad:** En contraste con la estacionalidad, este dato se repite de forma aperiódica, o sea que no se repite después de intervalos fijos.
* **Ruido:** Después de extraer el nivel, la tendencia y la estacionalidad/ciclicidad, lo que queda es ruido. El ruido es una fluctuación completamente aleatoria en los datos.

Existen dos tipos generales de descomposición en series de tiempo: descomposición estacional aditiva (cuando los componentes se pueden sumar para obtener la serie de tiempo original), y descomposición estacional multiplicativa (los componentes al ser multiplicados dan como resultado la serie original). [37]

### 

### **3.10.12 Clústeres; más vendidos menos vendidos**

El clustering en ciencia de datos se trata de agrupar muestras de un conjunto de datos por propiedades similares. A diferencia de otros algoritmos como los árboles de decisión, el clustering se encuentra dentro de los algoritmos no supervisados. La mayor diferencia entre ambos radica en que los datos de entrada para el cluster, no están previamente etiquetados mediante interacción humana con atributos que simplemente indican la clase a la que pertenecen. El clustering nos ayuda a identificar grupos de datos con características similares previamente desconocidos de manera automática sobre el conjunto de datos en el que se trabaja, además puede ayudar a identificar casos extremos (previamente discutidos, como outliers) que representan comportamientos inusuales en nuestros datos. [38]

La utilidad de este tipo de algoritmo en una tienda de conveniencia radicaría en poder visualizar datos como varios grupos de productos que abarquen desde los más vendidos hasta los menos vendidos, esto sin la necesidad la intervención humana para definir alguna constante arbitraria que dicte a partir de cuantas unidades se considera que un producto es bien vendido y cualquier producto cuyas ventas estén por debajo de esa constante sea por extensión un producto poco vendido. Además, nos permitiría visualizar otro tipo de correlaciones, unidades vendidas y precio, por ejemplo, unidades vendidas y existencias del producto, mostrándonos de manera tentativa la demanda que puede tener algún producto y la oferta que existe del mismo. Como ultima ventaja, y en contraste con la intervención humana para definir los límites entre “muy vendido” y “poco vendido”, donde el agrupamiento de productos es inevitablemente dicotómico, está el hecho de que estos modelos, podrían generar el número necesario de grupos indicando un rango más amplio en la escala de ventas, por ejemplo, productos muy poco vendidos, poco vendidos, medianamente vendidos, muy vendidos, y así sucesivamente, aunque claro, esto dependerá del conjunto de datos, pero la ventaja es que estos algoritmos nos ofrecen una manera automática de generar estas agrupaciones.

Existen al menos dos tipos de algoritmos de clustering según la pertenencia de los registros a grupos; primero, se encuentra el agrupamiento duro (hard clustering), donde cada punto pertenece obligatoriamente a algún cluster. Por otro lado, en el agrupamiento suave (soft clustering), existe una probabilidad de que un punto sea asignado a alguno de los clusters generados.

Cada metodología para clustering sigue una serie de reglas distintas para definir el significado de “similitud” entre distintos puntos de datos; existen más de 100 algoritmos de clustering, sin embargo, son menos los que se utilizan de manera popular. A continuación, se describen estos:

* **Modelos de Conectividad:** Como el nombre sugiere, estos modelos se basan en la noción de que los puntos de datos más cercanos en el espacio de datos muestran más similitud entre sí que los puntos de datos que se encuentran más lejos. Estos modelos pueden seguir dos enfoques. En el primer enfoque, comienzan clasificando todos los puntos de datos en clústeres separados y luego los van agregándolos a medida que la distancia disminuye. En el segundo enfoque, todos los puntos de datos se clasifican como un único clúster y luego se dividen a medida que la distancia aumenta. Además, la elección de la función de distancia es subjetiva. Estos modelos son muy fáciles de interpretar, pero carecen de escalabilidad para manejar grandes conjuntos de datos. Algunos ejemplos de estos modelos son los algoritmos de clustering jerárquico y sus variantes.
* **Modelos Centroides:** Son algoritmos de clustering iterativos en los que la noción de similitud se deriva por la cercanía de un punto de datos al centroide o centro del clúster. El algoritmo de clustering k-Means es un algoritmo popular que cae en esta categoría. En estos modelos, se debe mencionar de antemano el número de parámetros de clúster requeridos. Estos modelos se ejecutan iterativamente para encontrar los óptimos locales.
* **Modelos de Distribución:** Estos modelos de clustering se basan en la noción de cuán probable es que todos los puntos de datos en el clúster pertenezcan a la misma distribución (por ejemplo: Normal, Gaussiana). Estos modelos a menudo sufren de sobreajuste. Un ejemplo popular de estos modelos es el algoritmo de Expectation-maximization.
* **Modelos de Densidad:** Estos modelos buscan en el espacio de datos áreas de variada densidad de puntos de datos. Aíslan diferentes regiones densas y asignan los puntos de datos dentro de estas regiones al mismo clúster. Ejemplos populares de modelos de densidad son DBSCAN y OPTICS. Estos modelos son particularmente útiles para identificar clústeres de forma arbitraria y detectar anomalías o casos outliers. [39]

## 

## **3.11 Logística y roles en tiendas de conveniencia.**

Una tienda de conveniencia se refiere a un establecimiento minorista de tamaño moderado (menos de 500m²) que se especializa en la oferta de una amplia gama de productos de consumo diario, alimentos, bebidas, productos de cuidado personal y otros productos de conveniencia. Estas tiendas son conocidas por su horario de operación extendido, que a menudo abarca las 24 horas del día, los 7 días de la semana, lo que las hace accesibles en cualquier momento.

La principal característica distintiva de una tienda de conveniencia es su énfasis en la accesibilidad y la rapidez. Están estratégicamente ubicadas en áreas de alto tráfico, como intersecciones viales, estaciones de servicio y zonas residenciales densamente pobladas. Su objetivo es brindar a los clientes la comodidad de adquirir productos necesarios sin tener que realizar una gran compra. Por lo tanto, su inventario se selecciona cuidadosamente para incluir elementos que satisfagan las necesidades inmediatas y de última hora de los consumidores. [40]

### 

### **3.11.1 Roles y responsabilidades.**

Dentro de una tienda de conveniencia, tanto el gerente como el empleado desempeñan roles fundamentales en el funcionamiento diario de la tienda. Cada uno tiene sus responsabilidades específicas que contribuyen al éxito del establecimiento:

**Gerente.**

El gerente de la tienda de conveniencia asume una posición de liderazgo y supervisión. Sus responsabilidades clave incluyen:

* **Gestión de personal:** El gerente contrata, capacita y supervisa al personal de la tienda, incluyendo a los empleados de tiempo completo y parcial. También es responsable de la programación de turnos y la gestión de recursos humanos.
* **Gestión del inventario:** Supervisa el inventario de productos, asegurando que se mantenga en niveles adecuados. El gerente es responsable de realizar pedidos de reposición cuando sea necesario y gestionar el inventario para evitar pérdidas debido a productos caducados.
* **Servicio al cliente:** Asegura que se ofrezca un excelente servicio al cliente. Aborda quejas o problemas de los clientes y se asegura de que el personal esté capacitado para brindar una experiencia positiva al cliente.
* **Control de costos:** Administra los costos operativos, incluidos los costos de personal y los costos de mercancía. Debe encontrar formas de optimizar los gastos sin comprometer la calidad del servicio.
* **Gestión financiera:** Es responsable de la contabilidad y la gestión de las finanzas de la tienda, lo que incluye el manejo de los ingresos, gastos y la presentación de informes financieros.
* **Cumplimiento de normativas:** Asegura que la tienda cumpla con todas las regulaciones locales y federales, lo que incluye requisitos de seguridad alimentaria, licencias y otros aspectos legales.

**Empleado.**

El empleado en una tienda de conveniencia es el rostro visible de la tienda y desempeña un papel crucial en la satisfacción del cliente. Sus responsabilidades suelen incluir:

* **Atención al cliente:** Brinda atención al cliente, ayudando a los clientes a encontrar productos y respondiendo a sus preguntas. Debe ser amable y servicial.
* **Operación de caja:** Procesa transacciones de compra utilizando la caja registradora y maneja el dinero con precisión. Esto implica aceptar pagos en efectivo, tarjetas de crédito y débito.
* **Reabastecimiento:** Mantiene las estanterías y vitrinas bien surtidas, reponiendo productos a medida que se agotan y manteniendo el área de ventas ordenada.
* **Limpieza:** Mantiene la tienda limpia y ordenada, incluyendo áreas de ventas y baños.
* **Control de inventario:** Colabora en la revisión de inventario y alerta al gerente sobre productos que necesitan ser reabastecidos o eliminados debido a la caducidad.
* **Seguridad**: Contribuye a la seguridad de la tienda al estar atento a situaciones sospechosas y siguiendo los procedimientos de seguridad establecidos.

### 

### **3.11.2 Logística.**

La logística en tiendas de conveniencia y minoristas desempeña un papel fundamental en la cadena de suministro de productos, asegurando que los productos estén disponibles para satisfacer la demanda de los clientes. La logística se ocupa de la planificación, ejecución y control de la adquisición, transporte y almacenamiento de productos desde el proveedor hasta la tienda de conveniencia y, finalmente, a las manos del consumidor.

La logística en tiendas minoristas, como las tiendas de conveniencia, tiene un enfoque en la eficiencia y la agilidad. Algunos aspectos clave incluyen:

1. **Adquisición de mercancías:** Implica la selección y compra de productos de los proveedores. La logística juega un papel en la elección de proveedores confiables y en la negociación de términos y condiciones que optimicen la cadena de suministro.

2. **Almacenamiento:** Las tiendas de conveniencia a menudo tienen un espacio limitado, lo que requiere una logística eficiente de almacenamiento. La disposición de productos y la rotación adecuada del inventario son cruciales para garantizar que los productos estén disponibles y en buen estado.

3. **Gestión de inventarios:** La gestión de inventarios en una tienda de conveniencia implica mantener un equilibrio entre el exceso y la falta de existencias. La logística ayuda a controlar y optimizar los niveles de inventario.

La logística es esencial para asegurar que los productos estén disponibles para los clientes en el momento adecuado y en la cantidad requerida. Al mismo tiempo, debe minimizar los costos operativos y reducir el desperdicio. Un proceso de logística eficiente en tiendas de conveniencia es fundamental para mantener la competitividad y satisfacer las demandas cambiantes de los consumidores.

## **3.12 Desafíos específicos en tiendas de conveniencia.**

Estos desafíos impactan directamente en la eficiencia operativa, la satisfacción del cliente y la rentabilidad del negocio. A continuación, exploraremos tres de los desafíos más críticos que enfrentan las tiendas de conveniencia.

### 

### **3.12.1 Recepción de mercancías.**

Este proceso involucra la recepción y registro de productos provenientes de múltiples proveedores, lo que puede ser una tarea desafiante debido a la diversidad de productos, las entregas frecuentes y la necesidad de mantener la frescura y la variedad de productos. En este contexto, es crucial entender los desafíos específicos relacionados con la recepción de mercancías en tiendas de conveniencia. A continuación, exploraremos estos desafíos.

**Gestión de entregas múltiples:** Las tiendas de conveniencia suelen recibir entregas de múltiples proveedores a lo largo del día. Esto crea un desafío logístico, ya que es necesario coordinar y gestionar estas entregas de manera eficiente. Hay que asegurar que las entregas lleguen a tiempo y se descarguen de manera ordenada es fundamental para mantener un flujo constante de productos en la tienda.

**Registro preciso en el sistema de inventario:** Cada producto recibido debe ser registrado con precisión en el sistema de inventario de la tienda. Esto incluye la actualización de cantidades disponibles, la asignación de ubicaciones de almacenamiento y la gestión de fechas de vencimiento (particularmente importante para productos perecederos). Un error en el registro puede llevar a problemas de inventario y pérdida de productos, a esto se le llama riesgo operativo.

**Organización eficiente de productos en el almacén**: Después de la recepción, los productos deben ser organizados de manera eficiente en el almacén de la tienda. Esto implica asignar ubicaciones adecuadas, teniendo en cuenta la rotación de inventario y la accesibilidad tanto para los empleados como para los clientes.

**Productos perecederos:** Dado que las tiendas de conveniencia ofrecen una variedad de productos perecederos, como alimentos frescos y bebidas, la gestión de la recepción se vuelve aún más crítica. Se deben seguir rigurosamente las regulaciones de seguridad alimentaria y se deben implementar procedimientos específicos para garantizar la frescura y calidad de estos productos.

**Eficiencia y agilidad:** En un entorno de tienda de conveniencia, la eficiencia y la agilidad en el proceso de recepción son fundamentales. Los empleados deben ser capaces de recibir, verificar y registrar mercancías de manera rápida y precisa para minimizar el impacto en las operaciones diarias de la tienda. [41]

### 

### **3.12.2 Ubicación y almacenamiento.**

En un entorno donde la rapidez y la accesibilidad de los productos son imprescindibles para la satisfacción del cliente, la organización eficiente de productos en el almacén se convierte en un factor determinante en el éxito del negocio. Este punto del marco teórico se centra en la gestión de ubicación y almacenamiento en tiendas de conveniencia, considerando su relevancia y los desafíos que implica:

**Optimización del espacio:** Las tiendas de conveniencia a menudo tienen espacio limitado en comparación con otros tipos de establecimientos. La gestión de ubicación y almacenamiento implica la optimización del espacio disponible para garantizar que la tienda pueda ofrecer una variedad de productos y mantener un flujo de clientes.

**Organización de productos:** Los productos deben estar organizados de manera lógica y eficiente para facilitar la búsqueda y la accesibilidad tanto para los empleados como para los clientes. Esto implica definir sistemas de clasificación y etiquetado que permitan encontrar productos rápidamente.

**Rotación de inventario:** La rotación de inventario es crucial en las tiendas de conveniencia. Los productos deben ubicarse de manera que los más antiguos se utilicen primero. Esto asegura que los productos perecederos se vendan antes de que venzan y minimiza las pérdidas por productos caducados.

**Productos en exhibición y almacenamiento en bodega:** Las tiendas de conveniencia a menudo tienen una sección de exhibición y un área de almacenamiento en bodega. La gestión efectiva implica decidir qué productos se exhiben y cuáles se almacenan en la bodega. Esto se basa en la demanda, la temporada y la rotación de inventario.

**Seguridad:** La organización de productos debe tener en cuenta la seguridad. Los productos deben estar ubicados de manera que se minimicen los riesgos de accidentes o robos. Los productos más costosos o de alto valor pueden requerir una ubicación especial para garantizar su seguridad.

**Categorías y clasificación**: La categorización y clasificación de productos es esencial para la organización. Los productos deben agruparse lógicamente, como alimentos, bebidas, productos para el hogar, entre otros. Esto facilita tanto la gestión como la experiencia del cliente.

**Actualización y mantenimiento:** La gestión de ubicación y almacenamiento implica mantener un proceso constante de revisión y actualización. A medida que cambian las preferencias de los clientes y se introducen nuevos productos, se deben ajustar las ubicaciones y el espacio de almacenamiento. [42]

### **3.12.3 Productos perecederos****.**

Estos productos incluyen alimentos frescos, bebidas refrigeradas, productos lácteos y cualquier otro producto que tenga una vida útil limitada. La gestión de productos perecederos plantea desafíos específicos que deben abordarse para garantizar la calidad de los productos y minimizar las pérdidas. A continuación, exploraremos la importancia de la gestión de productos perecederos y los desafíos que conlleva:

**Importancia de la gestión de productos perecederos.**

**Satisfacción del cliente:** Las tiendas de conveniencia a menudo se promocionan por la disponibilidad de productos frescos y listos para consumir. Los clientes esperan encontrar alimentos y bebidas de alta calidad. La falta de gestión adecuada de productos perecederos puede resultar en productos de mala calidad, lo que afecta negativamente la satisfacción del cliente.

**Minimización de desperdicio:** Los productos perecederos tienen una vida útil limitada. Una gestión deficiente puede resultar en la pérdida de inventario debido a productos caducados o en mal estado. Esto se traduce en pérdidas económicas para la tienda.

**Cumplimiento normativo:** La venta de productos perecederos a menudo está sujeta a regulaciones y normativas estrictas en cuanto a la seguridad alimentaria. La gestión adecuada es indispensable para cumplir con estas regulaciones y evitar sanciones legales.

**Desafíos de la gestión de productos perecederos.**

**Control de caducidades:** Es importante llevar un registro preciso de las fechas de vencimiento de los productos y garantizar que los productos más antiguos se utilicen antes. Esto requiere una supervisión constante y sistemas de seguimiento de inventario.

**Almacenamiento adecuado:** Los productos perecederos deben almacenarse a la temperatura y humedad adecuadas para mantener su calidad. Esto puede requerir sistemas de refrigeración o congelación específicos.

**Rotación de inventarios:** La rotación adecuada de inventarios es esencial para garantizar que los productos no permanezcan en el almacén durante mucho tiempo. Los productos más nuevos deben ubicarse detrás de los más antiguos y utilizarse primero.

**Optimización de pedidos:** Las tiendas deben realizar pedidos precisos de productos perecederos para evitar el exceso o la falta de inventario. Esto se basa en datos históricos de ventas y la demanda actual. [43]

## 

## **3.13 La tecnología en los sistemas de almacenes.**

Es fundamental comprender dos conceptos cruciales que desempeñarán un papel importante en la eficiencia operativa y la satisfacción del cliente. Estos dos conceptos son el Warehouse Management System (WMS) y el Inventory Management System (IMS). En esta sección, exploraremos en profundidad lo que representan, su importancia y cómo su unión puede ofrecer beneficios significativos.

### 

### **3.13.1 Warehouse Management System (WMS).**

El Warehouse Management System (WMS), en español Sistema de Gestión de Almacenes, es una herramienta tecnológica fundamental en la gestión eficaz de almacenes y centros de distribución. Su principal propósito es optimizar las operaciones de almacenamiento y distribución, asegurando que los productos se muevan de manera eficiente, desde su llegada al almacén hasta su entrega al cliente final.

**Importancia del WMS.**

**Optimización de espacio:** Un WMS ayuda a utilizar el espacio de almacenamiento de manera eficiente, asegurando que los productos se almacenan en los lugares adecuados y de manera organizada.

**Seguimiento de existencias:** Permite un seguimiento preciso de los niveles de existencias, lo que garantiza que siempre haya productos disponibles y reduce los casos de agotamiento de existencias.

**Mejora de la eficiencia:** Ayuda en la asignación de tareas, lo que ahorra tiempo y mano de obra además de que aumenta la eficiencia operativa.

**Generación de datos y métricas:** Los WMS son capaces de medir la eficiencia de las operaciones, lo que permite la toma de decisiones basada en datos para mejorar los procesos.

**Reducción de errores:** Minimiza errores humanos al guiar a los trabajadores a través de los procesos de manera sistemática.

**Uso del WMS.**

El WMS en una tienda de conveniencia permite un control y seguimiento completo de todos los productos en el almacén. Puede analizar la capacidad y los niveles de existencias y hacer un seguimiento de cuánto tiempo y mano de obra se dedica a las distintas actividades. Esto permite generar datos que miden la eficiencia y sugieren formas de mejorarla. En este caso usando un WMS de complejidad nivel 2. [44]

Un WMS también es esencial para llevar un registro de productos con caducidad, lo que asegura que los productos frescos se muevan y vendan antes de su fecha de vencimiento.

### **3.13.2 Inventory Management System (IMS).**

El Sistema de Gestión de Inventario (IMS) es una parte sustancial en la gestión de almacenes de tiendas de conveniencia. Su enfoque se centra en controlar y gestionar el inventario de productos de la tienda, lo que es fundamental para garantizar que los productos estén disponibles en el momento adecuado, en las cantidades correctas y en óptimas condiciones. A continuación, exploraremos en detalle el significado, la importancia y el uso de un IMS en este contexto.

**Significado de IMS.**

Un Inventory Management System (IMS) es un conjunto de herramientas, tecnologías y procesos que ayudan a supervisar, controlar y optimizar el inventario de productos en una tienda de conveniencia. Su principal objetivo es asegurarse de que los productos estén siempre disponibles para satisfacer la demanda de los clientes, al tiempo que se minimizan los costos asociados con el almacenamiento excesivo o la falta de existencias. El IMS se enfoca en administrar la cantidad correcta de productos para evitar pérdidas y mantener satisfechos a los clientes. [45]

**Importancia del IMS.**

La importancia del IMS en una tienda de conveniencia radica en varios aspectos fundamentales:

**Satisfacción del cliente:** Mantener los productos en mostrador es esencial para satisfacer las necesidades de los clientes. Un IMS eficiente asegura que los clientes encuentren lo que necesitan cuando lo necesitan, lo que mejora la satisfacción del cliente y fomenta la lealtad.

**Reducción de costos:** Controlar el inventario de mostrador evita costos asociados con el almacenamiento excesivo y minimiza la pérdida de productos debido a la caducidad o el daño. También ayuda a optimizar las compras al tener una visión clara de cuándo y cuánto reabastecer.

**Gestión de productos frescos:** En tiendas de conveniencia, donde se venden alimentos y productos frescos, un IMS es crucial para rastrear fechas de caducidad y asegurar que los productos más antiguos se vendan antes de los nuevos, evitando desperdicio.

**Eficiencia operativa:** Un IMS bien implementado permite una administración más eficiente del tiempo y la mano de obra al facilitar las tareas de inventario, reabastecimiento y verificación de productos.

### **3.13.3 La unión de WMS e IMS: Optimizando la gestión de almacenes e inventario.**

La combinación de un Sistema de Gestión de Almacenes (WMS) y un Sistema de Gestión de Inventario (IMS) es un componente crucial para la eficiente operación de una tienda de conveniencia, y se convierte en el núcleo fundamental de este proyecto. Esta unión tiene como objetivo transformar y mejorar la administración del almacén y la gestión de inventario en el contexto específico de las tiendas de conveniencia. En este apartado, examinaremos detenidamente la importancia de esta combinación y cómo puede marcar la diferencia en este entorno minorista único.

**La sinergia entre WMS e IMS.**

La combinación de un WMS y un IMS en tiendas de conveniencia representa una sinergia de capacidades esenciales. El WMS se enfoca en la gestión eficiente de las operaciones de almacén, mientras que el IMS se dedica a garantizar la disponibilidad y el control de los productos en el inventario. Esta unión estratégica combina la capacidad de medir la eficiencia y sugerir mejoras con la gestión precisa del inventario, creando un sistema integral que beneficia significativamente a la operación de la tienda.

**Importancia en tiendas de conveniencia.**

La relevancia de la unión de un WMS e IMS en tiendas de conveniencia es innegable y se manifiesta en múltiples aspectos:

**Optimización de la cadena de suministro**: Las tiendas de conveniencia operan con productos perecederos, lo que requiere una gestión precisa y en tiempo real. La combinación de un WMS y un IMS permite un seguimiento constante del inventario, mejorando la precisión de las predicciones y, en consecuencia, reduciendo costos y desperdicio.

**Satisfacción del cliente:** La disponibilidad constante de productos es crucial en una tienda de conveniencia. Al asegurar que los productos estén siempre en stock, los clientes obtienen lo que necesitan en el momento que lo necesitan, lo que fortalece su satisfacción y fomenta la lealtad.

**Reducción de costos:** La gestión eficiente de inventario y almacén contribuye a reducir costos asociados al almacenamiento excesivo y la pérdida de productos debido a la caducidad o el daño. Además, un sistema que mide y mejora la eficiencia ahorra tiempo y recursos.

**Mejora en la eficiencia operativa:** La combinación de un WMS e IMS automatiza muchas tareas manuales, como la gestión de inventario y la planificación de reabastecimiento, permitiendo a los empleados enfocarse en tareas estratégicas y de atención al cliente.

**Generación de datos accesibles:** La recopilación y el análisis de datos en tiempo real sobre el rendimiento del almacén y el inventario son esenciales para tomar decisiones informadas. Este análisis proporciona información valiosa para la toma de decisiones estratégicas. [46]

## 

## **3.14 Métricas y Términos de Control.**

En este apartado, exploraremos un conjunto de métricas y términos de control que desempeñan un papel importante en la gestión de almacenes de tiendas de conveniencia. Estas métricas y términos proporcionan a los administradores y responsables de la cadena de suministro las herramientas necesarias para evaluar, supervisar y optimizar las operaciones diarias. Este apartado detalla cómo estas herramientas son cruciales para el funcionamiento exitoso de una tienda de conveniencia.

### **3.14.1 Control de Calidad.**

La importancia del control de calidad radica en garantizar que los productos disponibles para los clientes sean de alta calidad, seguros y cumplan con los estándares y requisitos establecidos. En este contexto, el control de calidad implica un conjunto de prácticas y procesos diseñados para evaluar y garantizar que los productos almacenados y vendidos en la tienda cumplan con los estándares predefinidos. A continuación, se exploran algunos de los aspectos clave del control de calidad en tiendas de conveniencia:

1. **Inspección y evaluación de productos:** Esto implica la revisión y verificación de los productos almacenados en el almacén y en exhibición. Se inspeccionan aspectos como la fecha de caducidad, la integridad del embalaje, la calidad del producto y otros criterios relevantes.

2. **Normativas y estándares:** Las tiendas de conveniencia deben cumplir con las normativas y estándares establecidos por las autoridades reguladoras, tanto en términos de seguridad alimentaria como de calidad de los productos. Esto incluye la gestión de productos perecederos y la manipulación segura de productos químicos o materiales peligrosos.

3. **Capacitación del personal:** El personal que trabaja en tiendas de conveniencia debe recibir capacitación adecuada sobre los procedimientos de control de calidad. Esto incluye la manipulación segura de productos, la identificación de problemas potenciales y la toma de decisiones sobre la retirada de productos defectuosos.

4. **Retroalimentación del cliente:** El control de calidad también implica la recopilación de comentarios de los clientes. Las tiendas de conveniencia deben estar abiertas a las quejas o preocupaciones de los clientes y responder de manera oportuna y efectiva.

5. **Auditorías y evaluaciones regulares:** Para garantizar la consistencia y la calidad a largo plazo, se realizan auditorías y evaluaciones regulares del control de calidad en la tienda de conveniencia.

6. **Cumplimiento de políticas de calidad:** La tienda debe adherirse a políticas de calidad internas o externas establecidas por franquicias, proveedores o regulaciones gubernamentales. [47]

### **3.14.2 Stock-Out.**

El "stock-out" es un término que se utiliza para describir la situación en la que un producto específico se agota en el inventario de una tienda de conveniencia y ya no está disponible para los clientes. Este fenómeno puede tener un impacto significativo en las operaciones de la tienda y en la satisfacción del cliente.

**Impacto en las operaciones de la tienda:**

**Pérdida de ventas:** Cuando un producto se agota y no se repone rápidamente, la tienda pierde ventas, ya que los clientes no pueden comprar lo que necesitan.

**Insatisfacción del cliente:** Los clientes que no pueden encontrar un producto que están buscando pueden sentirse frustrados y decepcionados. Esto puede afectar negativamente la percepción de la tienda y reducir la lealtad del cliente.

**Impacto en otros productos:** El stock-out de un producto puede afectar a otros productos relacionados o complementarios. Por ejemplo, si falta el pan, es posible que los clientes no compren mermelada. Esto puede tener un efecto dominó en la falta de ventas.

**Costos de urgencia:** Para remediar un stock-out, la tienda puede incurrir en costos adicionales para obtener productos rápidamente, lo que puede reducir los márgenes de beneficio.

Para evitar el stock-out, la gestión de inventario es esencial. Estrategias como el reabastecimiento dinámico y el seguimiento en tiempo real de las existencias son importantes para garantizar que los productos estén disponibles cuando los clientes los necesiten. [48]

### **3.14.3 Punto de Reorden (Reorder Point).**

El punto de reorden es un valor crítico que ayuda a determinar cuándo se debe realizar un pedido para reponer un producto en la tienda de conveniencia. Este punto se basa en factores como la demanda del producto, el tiempo de espera para la entrega y la cantidad mínima que debe mantenerse en existencia. Su cálculo es esencial para evitar situaciones en las que los productos se agoten antes de que llegue un nuevo pedido.

**Importancia.**

**Evita la escasez de productos:** Al establecer un punto de reorden adecuado, se garantiza que los productos se repongan antes de agotarse, evitando la pérdida de ventas debido a productos no disponibles.

**Optimiza los costos de inventario:** El punto de reorden ayuda a mantener un inventario mínimo necesario para satisfacer la demanda, lo que minimiza los costos de almacenamiento.

**Cálculo:** El punto de reorden se calcula mediante la siguiente fórmula:

*Punto de Reorden = Demanda Diaria Promedio×Tiempo de Espera+Inventario de Seguridad.*

**Demanda Diaria Promedio:** Es el número de unidades promedio que se venden de un producto específico en un día.

**Tiempo de Espera:** Es el tiempo que transcurre desde que se realiza un pedido hasta que se recibe el nuevo inventario.

**Inventario de Seguridad:** Es una cantidad adicional de productos que se mantiene en existencia para lidiar con la variabilidad en la demanda y los tiempos de entrega.

Una vez que el inventario de un producto alcanza el punto de reorden, se debe realizar un nuevo pedido para garantizar que esté disponible antes de que se agote. El punto de reorden se personaliza para cada producto en función de su demanda, tiempo de entrega y variabilidad. [49]

### **3.14.4 Rotación de Inventario (Turnover).**

La rotación de inventario, también conocida como turnover de inventario, es un indicador crítico en la gestión de almacenes e inventario para tiendas de conveniencia. Representa la rapidez con la que un negocio gira o renueva su inventario durante un período de tiempo específico. En otras palabras, mide con qué frecuencia los productos se compran y se reponen en los estantes. La rotación de inventario es fundamental para garantizar que los productos no se queden obsoletos o se acumulen en el almacén.

**Importancia.**

**Eficiencia en la gestión del inventario:** Un alto índice de rotación de inventario indica que los productos se venden rápidamente y se reponen de manera eficiente, evitando la pérdida de ventas debido a productos agotados o exceso de inventario.

**Minimización de obsolescencia:** Una rotación de inventario saludable reduce el riesgo de tener productos obsoletos que no se venden, lo que podría generar pérdidas económicas.

**Optimización de costos:** Una rotación adecuada de inventario ayuda a evitar costos innecesarios relacionados con el almacenamiento y el mantenimiento de productos durante largos períodos.

**Cálculo:** La rotación de inventario se calcula mediante la siguiente fórmula:

*Rotación de Inventario=Ventas Netas Anuales/Promedio de Inventario Total.*

**Ventas Netas Anuales:** Representa la cantidad total de productos vendidos durante un año. Este valor se obtiene sumando todas las ventas de productos de la tienda en ese período.

**Promedio de Inventario Total:** Este valor se refiere al promedio de inventario que tienes disponible durante el mismo período. Para calcularlo, se suma el inventario inicial y el inventario final de un período y se divide entre 2. En esencia, se promedia el inventario al principio y al final del período.

Un alto índice de rotación de inventario (por ejemplo, 4) indica que, en promedio, todo el inventario de la tienda se vende y repone cuatro veces al año, lo que refleja una buena gestión del inventario. Por otro lado, un índice bajo podría indicar problemas en la gestión del inventario y la necesidad de ajustes. [50]

### **3.14.5** **Cross-Docking.**

El Cross-Docking es una técnica logística empleada en la gestión de almacenes y la distribución que implica recibir productos de diferentes proveedores y transferirlos directamente de la recepción a las áreas de exhibición o venta, sin almacenarlos en el almacén interno. Esta práctica se centra en la eficiencia y rapidez en la transferencia de productos, y en tiendas de conveniencia, es fundamental para agilizar el proceso de distribución y mantener productos frescos y disponibles.

**Relevancia.**

**Reducción de tiempos de almacenamiento:** El Cross-Docking permite minimizar los tiempos de almacenamiento en el almacén, lo cual es especialmente esencial en tiendas de conveniencia, ya que muchos de sus productos son perecederos o de alta rotación.

**Optimización de la cadena de suministro:** Facilita la gestión de inventario y la distribución rápida de productos, asegurando que los productos estén disponibles en las tiendas cuando sea necesario.

**Reducción de costos:** Al minimizar el tiempo de almacenamiento y la necesidad de un gran espacio de almacén, el Cross-Docking puede ayudar a reducir costos operativos en la cadena de suministro.

**Mayor eficiencia:** Facilita una gestión más eficiente de la cadena de suministro al evitar demoras innecesarias en el almacén.

El Cross-Docking es especialmente valioso para tiendas de conveniencia, donde la rapidez y la eficiencia son críticas para mantener productos frescos y disponibles para los clientes. Esta técnica permite una distribución más rápida y oportuna de los productos, lo que contribuye a una mejor satisfacción del cliente y a la optimización de las operaciones de la tienda.

### **3.14.6 Lead Time de Suministro.**

El "Lead Time de Suministro" se refiere al tiempo transcurrido desde que se coloca un pedido de productos o mercancías hasta que se entregan en el almacén de una tienda de conveniencia. Este tiempo de espera en la cadena de suministro es un factor crítico en la gestión de almacenes, ya que puede influir significativamente en la disponibilidad de productos y en la satisfacción del cliente.

**Influencia en la gestión de almacenes.**

**Planificación de inventario:** El conocimiento del lead time de suministro permite a las tiendas de conveniencia planificar y ajustar su inventario de acuerdo con los tiempos de entrega. Si el lead time es largo, es necesario mantener un inventario más grande para evitar stock-outs.

**Gestión de pedidos:** Una gestión eficiente del lead time implica hacer pedidos con la antelación adecuada para que los productos estén disponibles en el momento requerido.

**Optimización de costos:** Una gestión adecuada del lead time puede ayudar a reducir costos relacionados con el almacenamiento y la administración de inventario. Un lead time más corto permite reducir la necesidad de mantener grandes cantidades de inventario.

Para gestionar eficazmente el lead time de suministro, es esencial contar con sistemas de seguimiento y pronóstico precisos y mantener una comunicación efectiva con los proveedores. Reducir el lead time puede contribuir a una gestión de almacenes más eficiente y a una mayor satisfacción del cliente. [51]

### **3.14.7 KPI - Indicadores Clave de Rendimiento.**

Los Indicadores Clave de Rendimiento (KPI) desempeñan un papel clave en la gestión de almacenes, particularmente en el contexto de las tiendas de conveniencia. Estos KPI se han convertido en herramientas fundamentales para evaluar y medir el desempeño en diversas áreas de la gestión de inventario y almacenes. Los KPI proporcionan una evaluación tanto cuantitativa como cualitativa del rendimiento, lo que permite a los gestores tomar decisiones informadas y mejorar continuamente los procesos.

**Definición y utilidad.**

Los KPI son métricas cuantitativas que evalúan el rendimiento de un proceso o sistema en relación con los objetivos preestablecidos. En el contexto de la gestión de almacenes en tiendas de conveniencia, los KPI pueden abordar aspectos como el nivel de inventario, la eficiencia en el espacio de almacenamiento y otros aspectos. Los KPI proporcionan información precisa que permite a los gestores:

1. **Medir el rendimiento actual:** Los KPI proporcionan una medida objetiva del rendimiento en tiempo real. Por ejemplo, se puede medir la rotación de inventario, que evalúa cuánto tiempo se tarda en vender el inventario disponible.

2. **Comparar con objetivos y metas:** Los KPI se utilizan para comparar el desempeño real con los objetivos previamente establecidos, lo que facilita la identificación de desviaciones y áreas de mejora.

3. **Tomar decisiones informadas:** Los KPI ofrecen información relevante para la toma de decisiones. Por ejemplo, si el nivel de inventario es inferior al punto de reorden, se puede tomar la decisión de realizar un nuevo pedido de productos.

4. **Identificar áreas de mejora:** Los KPI también ayudan a identificar áreas que requieren mejoras. Por ejemplo, si el KPI muestra una baja eficiencia en el espacio de almacenamiento, esto puede indicar la necesidad de optimizar la distribución de productos.

Además, en una tienda de conveniencia, se pueden emplear diversos indicadores clave de rendimiento (KPI) para evaluar el desempeño en diferentes áreas:

* **Nivel de inventario.**
  + ***Tasa de obsolescencia.***
    - Evalúa la cantidad de productos obsoletos en comparación con el inventario total.
    - *Tasa de Obsolescencia = (Cantidad de Productos Obsoletos / Inventario Total) x 100*
* **Gestión del espacio.**
  + ***Porcentaje de espacio utilizado****.*
    - Evalúa la eficiencia en la utilización del espacio de almacenamiento.
    - *Porcentaje = (Espacio utilizado / Espacio total de almacenamiento) x 100%*
  + ***Densidad de almacenamiento.***
    - Mide cuántos productos se almacenan por metro cuadrado o cúbico.
* **Tiempo de entrega y recepción.**
  + ***Tiempo promedio de entrega de proveedores.*** 
    - Calcula cuánto tiempo lleva recibir mercancía de los proveedores.
    - Promedio entre la realización del pedido a un proveedor y la recepción de la mercancía.
* **Niveles de stock de productos clave.**
  + ***Porcentaje de existencias para productos más vendidos.***
    - Mide la disponibilidad de productos populares.
    - *Existencia = (Existencias de productos más vendidos / Total de existencias) x 100%*
  + ***Tasa de agotamiento de stock.***
    - Evalúa la frecuencia de productos agotados en la tienda.
    - *Stock-Out = (Número de productos agotados / Total de productos) x 100%*
* **Gestión de caducidad.**
  + ***Porcentaje de productos vencidos.***
    - Evalúa cuántos productos han caducado en relación con el inventario total.
    - *Tasa = (Productos vencidos / Total de productos en inventario) x 100%*
  + ***Días de vida promedio de productos perecederos.***
    - Calcula cuántos días en promedio permanecen los productos perecederos en el inventario.
    - *Promedio = (Días de vida Total / Cantidad de productos) x 100%*
* **Satisfacción del cliente.**
  + ***Encuestas de satisfacción del cliente.***
    - Evalúa la satisfacción de los clientes con la tienda y los productos. [52]

### **3.14.8 SKU (Stock Keeping Unit).**

Un SKU, o "Stock Keeping Unit", es un identificador único utilizado para rastrear y gestionar productos individuales dentro del inventario de una tienda de conveniencia. Cada producto o variante de producto tiene su propio SKU, lo que facilita la identificación y el seguimiento de los productos en el almacén.

**Relevancia en el Seguimiento de Productos.**

Precisión en el inventario**: El uso de SKUs permite un seguimiento preciso del inventario, lo que** es esencial para mantener registros exactos de los productos disponibles y agilizar la gestión de almacenes.

**Identificación rápida**: Los SKUs proporcionan una forma rápida y eficiente de identificar productos específicos, lo que es especialmente importante en una tienda de conveniencia donde la rapidez es clave.

**Gestión de variaciones**: En tiendas de conveniencia, es común tener productos en diferentes tamaños, sabores o variantes. Los SKUs ayudan a diferenciar y gestionar estas variaciones de manera efectiva.

**Asignación de SKU.**

**Asignar un SKU único**: Cada producto o variante de producto debe tener un SKU único. Este número o código es específico para cada producto y no se repite en otros productos de tu inventario.

**Estructura significativa:** Diseñar una estructura de SKU que tenga sentido para el negocio. Se puede incluir información relevante en el propio SKU, como el tipo de producto, el proveedor, el tamaño, el color, etc. Por ejemplo, un SKU para una lata de refresco de cola de 500 ml podría ser "COLA-500ML-001".

**Etiquetado en el almacén:** Utilizar etiquetas de SKU en los estantes y áreas de almacenamiento para asegurarse de que cada producto se coloca en el lugar correcto. Esto facilita la localización de productos durante la reposición y reduce errores. [53]

### **3.14.9 Análisis ABC.**

El Análisis ABC es una técnica fundamental en la gestión de inventario que se emplea para clasificar y priorizar productos en función de su importancia relativa en la operación de una tienda de conveniencia. Esta metodología es esencial en la administración de inventario para garantizar una eficiente asignación de recursos y satisfacción de la demanda del cliente.

**Categorización de Productos:** Los productos se dividen en tres categorías: A, B y C, basadas en su valor en inventario y su rotación. A continuación, se describe la clasificación típica:

* **Categoría A:** Esta categoría incluye productos de alta importancia. Representan una parte significativa del valor del inventario, pero tienen una rotación más baja. Estos productos suelen contribuir de manera sustancial a los ingresos de la tienda, y su gestión es crítica. Generalmente, estos productos representan aproximadamente el 20% del inventario, pero contribuyen al 80% de los ingresos.
* **Categoría B:** Aquí se encuentran los productos de importancia moderada. Aunque tienen un valor de inventario decente, su rotación es intermedia. La gestión de esta categoría requiere atención, pero no tan intensiva como la de la Categoría A. Por lo general, representan alrededor del 30% del inventario y contribuyen al 15-20% de los ingresos.
* **Categoría C:** Esta categoría comprende productos de menor importancia. A pesar de su alta rotación, su valor en inventario es bajo. Estos productos, aunque son numerosos, tienen un impacto limitado en los ingresos.

Una vez categorizados los productos, se aplican estrategias de gestión diferenciadas. Para el marco teórico, es relevante entender cómo estas estrategias impactan la eficiencia de la tienda de conveniencia.

**Productos de Categoría A:** Debido a su alta importancia, estos productos requieren una gestión cercana. Las estrategias deben centrarse en reducir costos, evitar excesos de inventario y garantizar que estén siempre disponibles para satisfacer la demanda.

**Productos de Categoría B:** La gestión de estos productos es cuidadosa, pero no tan intensiva como la de Categoría A. El equilibrio entre satisfacer la demanda y evitar acumulaciones innecesarias es clave.

**Productos de Categoría C:** Dado su bajo valor en inventario y alta rotación, las estrategias aplicadas a estos productos pueden ser más flexibles. Se pueden utilizar sistemas de reposición automática y mantener niveles de inventario más bajos.

**Evolución y revisión continua**: El Análisis ABC no es una tarea única. Para mantener su efectividad, se requiere una revisión y actualización periódica de las clasificaciones. Esto se debe a que la importancia de un producto puede cambiar con el tiempo debido a diversas variables, como fluctuaciones en la demanda y evolución del negocio. [54]

## **3.15 Autenticación de usuarios**

Autenticación es un término que se refiere al proceso de demostrar que algún hecho o algún documento es genuino. En el campo del cómputo, este término está típicamente asociado con probar la identidad de un usuario. Por lo general, un usuario demuestra su identidad proporcionando sus credenciales, es decir, una pieza de información acordada compartida entre el usuario y el sistema. La combinación de usuario y contraseña es el mecanismo de autenticación más común, es también conocido como autenticación por contraseña.[55]

### **3.15.1 Función Hash**

Es una función sobre una cadena de bits cuya longitud de salida es fija, debe cumplir las siguientes propiedades. (Un camino): debe ser computacionalmente inviable encontrar la entrada original de la función a partir del resultado de salida. (Resistente a colisión): debe ser computacionalmente inviable encontrar dos entradas que produzcan exactamente la misma salida.[56]

### **3.15.2 Salt**

Un valor no secreto que se utiliza en un proceso criptográfico, generalmente para asegurar que los resultados de los cálculos de una entrada particular a alguna función (por ejemplo hash), no puedan ser reutilizados por un atacante. [57]

### **3.15.3 Nonce**

Un valor que varía con el tiempo y que tiene, como máximo, una probabilidad despreciable de repetirse, tal como un valor aleatorio que se genera de nuevo para cada uso. Este valor se incluye habitualmente en los datos intercambiados por un protocolo, con el fin de garantizar la transmisión de datos en vivo en lugar de datos retransmitidos, detectando y protegiendo así contra ataques de repetición. [58], [59]

# **CAPÍTULO 4: ANÁLISIS DEL SISTEMA**

En adición al contenido del documento, en este capítulo se presenta un análisis exhaustivo del sistema, focalizando de manera primordial en la evaluación de los costos inherentes al diseño, producción e implementación por parte del equipo de desarrollo, además de la capacitación del personal involucrado; la configuración de la estructura de la base de datos junto con sus respectivos requisitos; la descripción de las historias de usuario que delimitan las funcionalidades principales de Kv-wims y, por último, la presentación de diagramas que detallan el funcionamiento del sistema para cada tipo de usuario de acuerdo con sus permisos.

## 

## **4.1 Estudio de factibilidad**

En las siguientes secciones del documento se realiza una serie de estudios para determinar la factibilidad del proyecto; Entre ellos, se cuenta con el estudio técnico, donde se analizan aspectos como software y hardware del sistema; el estudio económico, donde se estima el precio y viabilidad del proyecto según los costos respecto a inversión en tecnología, cantidad de desarrolladores, tiempo de desarrollo y líneas de código estimadas; y el estudio operacional, se revisan los posibles retos que puede tener el personal de las tiendas de conveniencia y si es posible que necesiten capacitación para utilizar Kv-wims.

### 

### **4.1.1 Estudio Técnico**

La fase inicial del estudio de factibilidad comprende el estudio técnico, cuyo propósito radica en llevar a cabo un análisis integral de los componentes de hardware y software que serán empleados en el proceso de desarrollo de Kv-wims.

Este análisis se enfoca en la integración de los recursos tecnológicos ya disponibles, conjuntamente con las herramientas existentes buscando minimizar los costos iniciales de inversión. En consecuencia, se prioriza la adquisición únicamente de aquellos componentes que no se encuentran actualmente en posesión, lo que se traduce en una reducción significativa de los gastos iniciales asociados al proyecto.

* ***Hardware***

Al tomar en consideración los requisitos mínimos sustanciales para el desarrollo de Kv-wims, y tras una minuciosa evaluación del hardware ya existente, se ha identificado que se cuenta con una serie de equipos de cómputo que no demandarán una inversión adicional ni la adquisición de nuevos recursos en la etapa inicial del proyecto.

A continuación, se muestra una tabla que proporciona una descripción de las especificaciones técnicas de los recursos tecnológicos que se encuentran disponibles y se destinan para llevar a cabo las tareas de diseño, producción y la subsiguiente implementación del sistema Kv-wims.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Equipo** | **Cantidad** | **Fabricante** | **Modelo** | **Procesador** | **Memoria RAM** | **Capacidad del disco duro** | **Precio ($)** |
| Laptop Lenovo IdeaPad 3 15IIL05 15.6", i3-1005G1, 8GB, 128Gb SSD | 1 | Lenovo | IdeaPad 3 | AMD RYZEN 3, 5000 SERIES | 20GB | 500GB | $ 9,000 |
|  | 1 | Dell | Latitude E7250 | Intel(R) Core (TM) I7-5600U | 16GB | 256GB | $5,000 |
| Acer E5-575-33BM Aspire E 15 Laptop de 15.6 Pulgadas, Full HD, procesador  Intel Core i3-7100U 7a generación, 4GB DDR4, 1 TB HDD, Windows 10 Home,  Únicamente Laptop, Negro obsidiana : Acer America | 1 | Acer | Aspire E-15 | Intel(R) Core (TM)  i3-5005U | 8GB | 1TB | $ 3, 500 |
| **Total ($)** | **$ 17,500** | | | | | | |

1. Resumen equipo de hardware destinado al proyecto

* ***Software***

En lo que respecta a la parte del software, los resultados del análisis de las tecnologías requeridas para la implementación de Kv-wims determinan que algunas de estas herramientas se hallan disponibles de forma gratuita, no obstante, existen otras que demandan recursos económicos para una inversión inicial, tal como sucede con el "hosting" que albergará la infraestructura de Kv-wims, permitiendo a las empresas el acceso y uso del sistema.

A continuación, se presenta una tabla que proporciona un desglose detallado de las herramientas de software que serán empleadas durante el proceso de desarrollo de Kv-wims. Esta información resulta fundamental para comprender la disposición de recursos y costos asociados al desarrollo y mantenimiento de Kv-wims, lo cual facilita una planificación eficaz y un manejo óptimo de los aspectos financieros del proyecto.

|  |  |  |  |
| --- | --- | --- | --- |
| **Herramienta** | **Costo** | **Descripción** | **Imagen** |
| **Python** | Gratuito | Es el lenguaje de programación con el que se llevará a cabo el análisis de datos del inventario en el almacén. | Python Software Foundation License - Wikipedia, la enciclopedia libre |
| **Django** | Gratuito | Es el framework de Python para el manejo de componentes web, su función es facilitar el manejo de los componentes y de código para el desarrollo web del sistema. | Django: All about the Python web development framework |
| **JavaScript** | Gratuito | Es el lenguaje de programación con el que se llevará a cabo el frontend y backend de la aplicación web. Además, es el lenguaje seleccionado para implementar la modelación 2D en el sistema. | JavaScript - Wikipédia |
| **HTML** | Gratuito | Es el lenguaje de programación que se utilizará para darle estructura a la aplicación web, crear botones, mostrar funciones y será vital para definir los elementos finales en el sistema. | Html 5 - Iconos gratis de redes sociales |
| **CSS** | Gratuito | Es la herramienta que permitirá controlar, diseñar y personalizar la apariencia de la aplicación web con el propósito final de darle estilo a nuestro sistema. |  |
| **Visual Studio Code** | Gratuito | Esta herramienta será nuestra IDE en la que se realizarán los códigos correspondientes al Frontend y Backend de la aplicación web. Esta es la alternativa más rentable pues tiene la capacidad de compilar todo tipo de códigos lo que facilita las tareas de la fase de desarrollo. |  |
| **Oracle - MySQL** | Gratuito | Este gestor de base de datos será el encargado de almacenar nuestras tablas correspondientes a los usuarios, almacenes, anaqueles, entre otras. Al tratarse de una herramienta Open Source, esto ocasiona que no se generen costos los cual es favorable para los intereses del sistema. | Oracle Publicly Commits to MySQL — SitePoint |
| **Azure** | Plan annual | Microsoft Azure es una plataforma de computación en la nube la cual permitirá construir, probar, desplegar y administrar la aplicación web en función de los servicios en la nube que ofrece como pueden ser almacenamiento de la página web, base de datos, seguridad de los datos, robustes del sistema, entre otros. | Microsoft Azure - Wikipedia, la enciclopedia libre |
| **Sistema Operativo Windows** | Gratuito | Windows es una familia de sistemas operativos desarrollada por Microsoft. Es uno de los sistemas operativos más populares en el mundo, utilizado en una amplia variedad de dispositivos, desde computadoras personales hasta servidores empresariales |  |
| **Sistema Operativo Linux (Ubuntu)** | Gratuito | Ubuntu es conocido por su facilidad de uso y su enfoque en la experiencia del usuario. Ubuntu se utiliza comúnmente en computadoras de escritorio y servidores y es conocido por su estabilidad y seguridad. | Ubuntu Logo PNG Transparent – Brands Logos |

1. Resumen componentes de software destinados para el proyecto

Como se puede observar, en el resumen de herramientas de software que se ocuparán para el desarrollo de Kv-wims únicamente se contempla un costo que es el del servidor de Azure cuya principal función será la de albergar nuestra aplicación.

De acuerdo con la información de los costos que se manejan en la página oficial de Azure, se contempló un plan que conjunta algunos servicios que ofrecen recursos suficientes para soportar el despliegue y almacenamiento del sistema desarrollado, a continuación, se muestra una tabla con el desglose de costos que conllevará adquirir un servidor en Azure.

|  |  |  |
| --- | --- | --- |
| **Servicio** | **Costo anual ($)** | |
| Plan de servicio P0v3 | $ 2308.06 |
| Dominio personalizado | $ 207.42 |
| Certificado SSL estándar | $ 1210.77 |
| **Total** | **$ 3726.42** | |

1. Desglose costos servicios de Azure

### **4.1.2 Estudio Económico**

Considerando los resultados derivados del análisis técnico realizado, se ha llegado a la conclusión de que la organización ya dispone de ciertos elementos de hardware y software que son adecuados y suficientes para afrontar las diversas etapas del desarrollo del proyecto.

En vista de esta premisa, el presente estudio se enfocará en llevar a cabo una evaluación de los costos generales asociados al proyecto, incluyendo los gastos relacionados con el equipo de desarrolladores y los costos operativos inherentes a la implementación de Kv-wims.

* ***Costos Generales***

Los costos generales hacen referencia a los gastos efectuados en consumibles (accesorios y materiales de papelería) de uso diario en las fases de desarrollo. Se consideró la siguiente lista de materiales necesarios y que serán utilizados durante las distintas fases del desarrollo de Kv-wims.

* Hojas blancas
* Lápices
* Plumas
* Gomas
* Sacapuntas
* Libretas
* Cartuchos para impresora
* Notas adhesivas

A continuación, en la siguiente tabla se muestra el costo de estos productos en relación con los meses de trabajo del equipo de desarrollo donde se realizó un ajuste pues existen periodos vacacionales y teniendo en cuenta que la etapa de desarrollo tiene menos meses para realizar operaciones:

|  |  |  |
| --- | --- | --- |
| **Elemento** | **Mes** | **Costo ($)** |
| Material de papelería | 1 | $ 100 |
| Material de papelería | 2 | $ 100 |
| Material de papelería | 3 | $ 150 |
| Material de papelería | 4 | $ 150 |
| Material de papelería | 5 | $ 100 |
| Material de papelería | 6 | $ 50 |
| Material de papelería | 7 | $ 50 |
| Material de papelería | 8 | $ 50 |
| Material de papelería | 9 | $ 50 |
| Material de papelería | 10 | $ 50 |
| **Total** | | **$ 850** |

1. Costos destinados a gastos generales de desarrollo.

* ***Costos Equipo de trabajo***

Los costos generados por el equipo de trabajo se refieren a aquellos gastos de tipo operativo que involucran al recurso humano el cual se encarga de realizar las tareas correspondientes de cada etapa de desarrollo del sistema.

Como parte de este análisis se tomó en cuenta que al ser 3 integrantes del equipo de desarrollo se dividieron las tareas y responsabilidades de acuerdo con las habilidades particulares de cada miembro. Entonces, como resultado de las decisiones del equipo se definieron las siguientes reglas:

1. Para la etapa de diseño se considera el salario de un Desarrollador Web Frontend.
2. Para la etapa de desarrollo se considera el salario de un Desarrollador Web Backend.
3. Para la parte de implementación se considera el salario de un Desarrollador de aplicaciones web.

Los salarios antes mencionados fueron consultados en una de las páginas de búsqueda de empleo más utilizada en México como lo es Indeed. A continuación, se muestra una tabla resumen de los sueldos consultados:

|  |  |  |  |
| --- | --- | --- | --- |
| **Puesto** | **Salario por mes** | **Salario por semana** | **Salario por hora** |
| Desarrollador Web Frontend | $ 16, 268 | $ 3,957 | $ 279 |
| Desarrollador Web Backend | $ 20, 064 | $ 4,815 | $ 100 |
| Desarrollador de aplicaciones web | $ 15,633 | $ 3,752 | $ 78.17 |

1. Costos individuales destinados al equipo de desarrollo

Considerando que se estima que el desarrollo del sistema completo desde el diseño hasta la implementación o despliegue será de 10 meses se debe de realizar ese cálculo para obtener el total de acuerdo con los meses de trabajo. En adición a estos costos, se decidió que se considere cobrar por hora en función de recibir una remuneración económica de acuerdo con las horas de trabajo reales que se invirtieron para el desarrollo del proyecto.

|  |  |
| --- | --- |
| **Puesto** | **Salario (10 meses)** |
| Desarrollador Web Frontend | $ 13,950 |
| Desarrollador Web Backend | $ 5,000 |
| Desarrollador de aplicaciones web | $ 781.7 |
| **Total** | **$ 19,731.7** |

1. Costos totales destinados al equipo de desarrollo.

* ***Costos operativos***

Los costos operativos se refieren a aquellos que surgen de las actividades que se llevan a cabo durante el periodo de desarrollo del sistema. De acuerdo con el análisis que se realizó para definir costos operativos, se consideraron únicamente aquellos elementos sustanciales que constituyen un gasto mientras está en desarrollo el sistema. A continuación, se muestra un resumen detallando la inversión de capital que cubra estos costos.

|  |  |  |  |
| --- | --- | --- | --- |
| **Elemento operativo** | **Descripción** | **Costo mensual** | **Costo tiempo de desarrollo (10 meses)** |
| Luz |  | $ 150 | $ 1,500 |
| Agua |  | $ 205 | $ 2, 050 |
| Internet |  | $ 385 | $ 3, 850 |
| Renta de oficina |  | $ 3,059 | $ 30, 590 |
| **Total** | | | **$ 37, 990** |

1. Costos destinados a los gastos operativos de desarrollo

### **4.1.3 Estudio Operacional**

Finalmente, es importante considerar que el personal de la tienda de conveniencia necesita ser capacitado para hacer un uso correcto de las funcionalidades que ofrecerá Kv-wims de acuerdo con el nivel de acceso al sistema.

Realizando una investigación en los comunicados oficiales emitidos por la Secretaría de Gobernación, la Secretaría del Honorable Consejo Técnico, estipula en un documento emitido 7 de Julio de 2022 que las empresas tienen la obligación de adquirir los servicios de capacitación de sus empleados con el fin aumentar las habilidades y capacidades de su personal en pro de subir el rendimiento de dicha empresa.

Entonces, recopilando la información investigada se planteó el siguiente plan de capacitación dirigido al personal perteneciente a las tiendas de conveniencia buscando dedicar el tiempo suficiente para mostrar las funcionalidades que ofrece Kv-wims.

Haciendo uso de los datos del Diario Oficial de la Federación, se estipula que los profesionales encargados de brindar una capacitación a la plantilla de trabajo de una empresa que adquiera sus servicios tendrán que cubrir el sueldo base que posee el profesional capacitador de acuerdo con el número de horas que se destinen a la capacitación. Entonces, basándonos en una estimación de tiempo aproximada se pretende seguir el siguiente resumen de costos para capacitar al personal de la tienda de conveniencia con el sistema Kv-wims:

|  |  |  |
| --- | --- | --- |
| **Total días** | **Número horas (día)** | **Bono ($ hora)** |
| 3 | 6 | $ 78.17 |
| **Total** | | **$ 1,407.06** |

1. Costo bono de capacitación

Analizando los datos de la tabla, se destinarán 3 días para llevar a cabo la capacitación con una duración de 18 horas, 6 horas por día el cuál se dividirá en 2 tiempos para capacitar al personal que labore en horario matutino y/o vespertino por lo que para cada horario de abordará la capacitación de 3 horas con escenarios reales pues sucederá en horario laboral.

El pago por adquirir la capacitación será de un total de $1,407 pesos que fungirán como el bono de capacitación por parte del profesional capacitador. Se asigno un costo por hora de capacitación pues de esta manera se cumplen las normas que se establecen en el Diario Oficial de la Federación con el propósito respetar en todo momento las reglas con las que se rigen este tipo de responsabilidades en el país.

### **4.1.4 Costo del proyecto**

Recopilando los costos de acuerdo con acá tipo de estudio (económico, operacional y técnico), podremos obtener un estimado de cuál sería la inversión que se tendría que hacer para lograr el desarrollo del proyecto. A continuación, se muestra un resumen de los costos que se obtuvieron en relación con el análisis de cada estudio desarrollado.

|  |  |
| --- | --- |
| **Estudio** | **Costo obtenido** |
| Técnico | $ 21,226.42 |
| Económico | $ 58,570.7 |
| Operacional | $ 1,407.06 |
| **Total** | **$ 81, 204.18** |

1. Resumen de costos por tipo de estudio

Realizando la suma de los costos por tipo de estudio se obtuvo que hasta este punto el proyecto tendría un costo aproximado de $81,204.18 pesos, pero aún se deben de realizar los cálculos correspondientes en función de la complejidad y algunas consideraciones más.

Como parte del análisis realizado para estimar el costo del proyecto en sus etapas de diseño se pueden ocupar distintos modelos que permiten calcular un primer presupuesto de inversión para adquirir los servicios que se ofrecen.

El modelo COCOMO II es una alternativa muy utilizada en los proyectos de software en los que se detalla un coste de los sistemas desarrollados en ingeniería tomando en cuenta los 3 principales hitos de dicho desarrollo como lo son el tiempo, el esfuerzo y el costo de producción.

De acuerdo con la definición de COCOMO II, existen 3 formas de obtener el costo de nuestro sistema en función de la complejidad de la naturaleza de este [60]. Utilizando los resultados del análisis de Kv-wims, se consideró que la propuesta posee una complejidad media por lo que se hará uso de esa alternativa de cálculo para estimar el costo total del software.

La alternativa de COCOMO II de mediana complejidad hace uso de los llamados puntos de objeto que se define como la medida de software indirecta que se calcula usando conteos de estos elementos:

1. Pantallas (interfaz de usuario)
2. Reportes
3. Componentes que constituyen el proyecto

Una vez que se tienen identificados los elementos a tomar en cuenta para realizar los cálculos iniciales, debemos tener en cuenta la siguiente tabla que resume cuáles serían los criterios de ponderación en función de la complejidad del proyecto:

|  |  |  |  |
| --- | --- | --- | --- |
| **Tipo de objeto** | **Peso de complejidad** | | |
| **Simple** | **Medio** | **Difícil** |
| Pantalla | 1 | 2 | 3 |
| Reporte | 2 | 5 | 8 |
| Componentes |  |  | 10 |

1. Ponderación de complejidad para tipos de objeto

El siguiente paso es categorizar las pantallas, reportes y componentes del sistema de acuerdo con la complejidad descrita en la tabla 5 para posteriormente obtener la cifra total de puntos de objeto.

|  |  |  |
| --- | --- | --- |
| **Tipo de objeto** | **Complejidad** | **Peso** |
| Pantalla inicio de sesión | Simple | 1 |
| Pantalla crear cuenta nueva | Simple | 1 |
| Pantalla Inicio empleado | Simple | 1 |
| Pantalla de inicio gerente | Simple | 1 |
| Pantalla de editor visual | Difícil | 3 |
| Pantalla de visualizador de almacén | Difícil | 3 |
| Pantalla de visualizador de anaquel | Difícil | 3 |
| Pantalla búsqueda de productos | Medio | 2 |
| Reporte tasa de daños | Medio | 5 |
| Reporte ventas | Medio | 5 |
| Reporte de ofertas | Medio | 5 |
| Reporte nuevos productos | Medio | 5 |
| Pantalla punto de venta | Medio | 2 |
| **Total de puntos de objeto** | | 37 |

1. Análisis ponderación de puntos de objeto

Dada la naturaleza del proyecto y que algunos de los componentes se van a reutilizar se debe calcular un parámetro llamado porcentaje de reuso que será parte importante para calcular el ajuste de los puntos de objeto (NOP, por sus siglas en inglés). Entonces, se hace el conteo del total de líneas de código a la par del conteo de líneas de código reutilizadas. A continuación, se muestran las tablas que describen un estimado de las líneas de código próximas y las que se van a reutilizar.

|  |  |
| --- | --- |
| **Lenguaje** | **Total de líneas** |
| HTML | 1,000 |
| CSS | 700 |
| Python | 700 |
| SQL | 200 |
| JavaScript | 700 |
| **Total** | **3, 300** |

1. Conteo parcial del total de líneas de código por lenguaje de programación

|  |  |
| --- | --- |
| **Lenguaje** | **Reuso de líneas de código** |
| HTML | 200 |
| CSS | 140 |
| Python | 140 |
| SQL | 40 |
| JavaScript | 140 |
| **Total** | **660** |

1. Resumen total de líneas de código reutilizadas

Ahora se realizar una regla de 3 para determinar el %reuso que quedaría de la siguiente forma:

Una vez calculado el %reuso podremos aplicar la fórmula para obtener el NOP, el modelo matemático que debemos resolver es el siguiente:

Sustituyendo los valores tenemos que:

Posteriormente, se debe seleccionar el valor de la tasa de productividad del equipo desarrollador, para eso nos basaremos en la tabla que detalla las tasas de productividad en los proyectos de software que ocupen puntos de objeto.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Experiencia/ capacidad del desarrollador | Muy baja | Baja | Nominal | Alta | Muy alta |
| Madurez/capacidad del entorno | Muy baja | Baja | Nominal | Alta | Muy alta |
| **PROD** | **4** | **7** | **13** | **25** | **50** |

1. Tasa de productividad para puntos de objeto

Haciendo un análisis de acuerdo con los proyectos que el equipo de trabajo ha realizado, podemos asumir que se cuenta con una experiencia y madurez Nominal pues contamos con experiencia en el área de desarrollo web suficiente para comprender la naturaleza del sistema Kv-wims que es la propuesta que se está realizando. Entonces, el valor correspondiente a la tasa de productividad (PROD) será de 13 lo cual nos ayudará para seguir con el desarrollo del cálculo.

Como siguiente punto vamos a obtener el “esfuerzo estimado”, mediante la siguiente fórmula:

Sustituyendo valores:

Finalmente, para calcular el costo más próximo a la realidad tendremos que utilizar la siguiente fórmula:

Dado que ya se cuenta con la estimación del esfuerzo, debemos de obtener cuál sería el costo por unidad de esfuerzo, para ello se aplica la siguiente fórmula:

La variable “Estimación del esfuerzo total” se debe de calcular también, entonces de acuerdo con la definición utilizaremos la fórmula:

Sustituyendo los valores obtenemos que:

Recopilando los datos, podemos sustituir en la fórmula del costo por unidad de esfuerzo:

Ahora, se calculará el costo real del proyecto mediante el uso de la fórmula antes planteada del “Costo del proyecto”, sustituimos los valores:

Entonces, el costo real aproximado de Kv-wims sería de 24,600 pesos mexicanos.

## **4.2 Análisis de riesgos**

El desarrollo de esta sección del documento es describir el análisis de riesgos realizado con el propósito de comprender la naturaleza de los riesgos y sus características incluyendo el caso en que podría suscitarse, el nivel de los riesgos mismos.

El análisis de riesgos implica una consideración detallada de la incertidumbre, fuentes de riesgo, consecuencias, probabilidades, eventos, escenarios, controles y su efectividad. Además, ayuda a comprender el nivel e impacto que podría tener en el sistema o en las etapas que constituyen al desarrollo de este.

### **4.2.1 Identificación**

La identificación de los riesgos es una actividad oportuna que toda empresa debe realizar para poder hacerle frente a los problemas que día a día tienen una posibilidad de surgir y que representan una amenaza para las operaciones de las empresas y los servicios que ofrece.

Durante las etapas de desarrollo de un proyecto de software es común tener riesgos que afecten a la operatividad del sistema; entonces como resultado de una revisión detallada de las etapas de implementación de Kv-wims se detectaron los siguientes riesgos.

|  |  |  |  |
| --- | --- | --- | --- |
| **Id** | **Riesgo** | **Descripción** | **Tipo** |
| **1** | Cambios en las historias de usuario | Realizar modificaciones repentinas a las historias de usuario que representan los requisitos desde el punto de vista del cliente podría ser contraproducente pues demasiadas iteraciones en el desarrollo de módulos del sistema prolongarían el tiempo de desarrollo, haciendo que no se cumplan los plazos de entrega. | Análisis |
| **2** | Falta de recursos | Los recursos tecnológicos y humanos son sustanciales para el desarrollo del sistema, sin embargo, en ocasiones por las cargas de trabajo y las limitaciones tanto de equipos de cómputo como del personal de desarrollo pueden ocasionar problemas impactando negativamente en el rendimiento de desarrollo de Kv-wims. | Tecnológico |
| **3** | Falta de comunicación cliente-desarrolladores | Conformar una buena y estable comunicación entre los desarrolladores y el cliente es sumamente importante pues se debe establecer una constante retroalimentación entre ambas partes con el fin de brindar claridad durante el proceso de desarrollo de Kv-wims acerca de los cambios y los requisitos que se implementan en el sistema. | Organizacional |
| **4** | Problemas de seguridad | Los riesgos vinculados a la seguridad están a la orden el día, frecuentemente la gente es curiosa y cuando ven montada una aplicación en la nube suelen tomar iniciativa para vulnerar las deficiencias de seguridad implementadas y con esto obtener información que en ocasiones es de índole privada pues se trata de datos sensibles de la empresa. | Tecnológico |
| **5** | Falta de pruebas exhaustivas | Durante el desarrollo del sistema se deben realizar pruebas por cada módulo implementado pues si el sistema llega a un caso límite de acuerdo con las especificaciones, este podría generar perdidas de datos, mal uso de de recursos o el fallo total del sistema en el peor de los casos. | Análisis |
| **6** | Falta de SEO | La Optimización para motores de búsqueda (SEO por sus siglas en inglés) permite siempre cumplir con los requerimientos mínimos de los motores de búsqueda que se van a utilizar para probar el despliegue del sistema. En caso de no existir, el dominio donde se encuentra la plataforma podría no ser mostrado como resultado relevante cuando los usuarios de internet busquen este tipo de sistemas mediante un motor de búsqueda. | Tecnológico |
| **7** | Incumplimiento de plazos | La calendarización de un proyecto es crucial durante todas las etapas que conforman a este, una buena distribución del tiempo de acuerdo con lapsos bien establecidos es un factor positivo para el desarrollador y los clientes como usuarios finales; entonces perjudicar los tiempos establecidos para mostrar un avance o un producto final podría representar pérdidas económicas que provoquen un valor negativo a la empresa que contrate el servicio y a los desarrolladores. | Proyecto |
| **8** | Errores de desarrollo | Una mala implementación de las funcionalidades principales, o poca robustez del código, podría ser fatal pues se comprometería la integridad de los datos, de los módulos desarrollados y del sistema en general pues en su mayoría los módulos pueden ocupar información recaba de un módulo anterior y esto impactaría negativamente por el manejo erróneo de información o de los reportes generados. | Análisis y Tecnológico |
| **9** | Pérdida de datos | La base de datos es un elemento sumamente importante y valioso en el funcionamiento de Kv-wims pues la naturaleza del sistema necesita de manera casi obligatoria el tener que obtener la información desde la base de datos por lo que, si se presenta algún error humano, un fallo técnico o bien un desastre de índole natural, impactaría negativamente en el funcionamiento ideal de los módulos implementados. | Tecnológico |
| **10** | Falta de capacitación al cliente | Saber usar completamente las funcionalidades de Kv-wims es un punto positivo para la inversión que se realiza por parte de la tienda de conveniencia, no aprovechar las facilidades que la herramienta ofrece sería perjudicial para el motivo antes mencionado y el rendimiento del establecimiento pues no se aprovecharía completamente cada módulo implementado. | Organizacional |
| **11** | Cambios regulatorios | El desconocimiento de los marcos legales de desarrollo puede representar problemas graves debido a que se debe de llevar a cabo un manejo seguro de los datos sensibles provenientes de los usuarios; al no realizar correctamente esta práctica, se estaría infringiendo alguna regla y generaría gastos extras proponiendo alternativas de solución para los problemas existentes. | Organizacional |
| **12** | Dependencia de terceros | Los efectos negativos que implica contratar un servicio de terceros son en su mayoría graves, el hecho de confiar el sistema a un servicio externo que difícilmente podemos controlar ocasionaría tener que enfrentar interrupciones de servicio en caso de fallos, incertidumbre de los cambios hechos en el sistema, costos extra de acuerdo con las políticas de uso del servicio, limitantes en las actualizaciones del sistema, entre otras cosas. | Proyecto y Tecnológico |

1. Riesgos identificados para el desarrollo del proyecto

### **4.2.2 Valoración**

La valoración de riesgos tiene como principal propósito ser una herramienta que ayude en la toma de decisiones cuando se presente una de estas situaciones. Esta herramienta implica comparar los resultados del análisis de riesgos con los criterios para riesgos establecidos para determinar cuándo se requiere realizar una acción adicional.

Recabando los resultados obtenidos en el análisis de riesgos, se tomó la decisión de categorizar los riesgos de acuerdo con los niveles de probabilidad parcial de suscitarse y el impacto directo que tendría en el sistema, dicha categorización se resume en la siguiente tabla.

|  |  |  |  |
| --- | --- | --- | --- |
| **Tipo de impacto** | **Criterios** | **Probabilidad** | **Criterios** |
| Bajo | 0% - 25% | Baja | 0 % - 25% |
| Moderado | 26% - 50 % | Moderada | 26% – 50% |
| Alto | 51% - 100% | Alta | 51% - 100% |

1. Criterios categorización de los riesgos

Entonces una vez definida la escala que permitirá categorizar a los riesgos identificados en el análisis, a continuación, se presenta una tabla resumen detallando cuál es el riesgo, la parcial probabilidad de que suceda y finalmente la escala del impacto que tendría directamente en el sistema.

|  |  |  |  |
| --- | --- | --- | --- |
| **Id** | **Riesgo** | **Impacto** | **Probabilidad** |
| 1 | Cambios en las historias de usuario | Alto | Moderada |
| 2 | Falta de recursos | Alto | Moderada |
| 3 | Falta de comunicación cliente-desarrolladores | Moderado | Baja |
| 4 | Problemas de seguridad | Alto | Moderada |
| 5 | Falta de pruebas exhaustivas | Moderado | Moderada |
| 6 | Falta de SEO | Alto | Moderada |
| 7 | Incumplimiento de plazos | Alto | Baja |
| 8 | Errores de desarrollo | Alto | Moderada |
| 9 | Pérdida de datos | Moderado | Moderada |
| 10 | Falta de capacitación al cliente | Moderado | Baja |
| 11 | Cambios regulatorios | Alto | Moderada |
| 12 | Dependencia de terceros | Alto | Moderada |

1. Resumen categorización de los riesgos

### **4.2.3 Plan de contingencia**

Una vez que se han analizado y categorizado los riesgos, es importante que se tengan soluciones para hacerles frente en caso de ser necesario; es por eso que, el plan de contingencia toma como objetivo primordial implementar acciones que permitan minimizar el impacto de los riesgos, garantizar la continuidad del proyecto, toma de decisiones contra riesgos, entre otras acciones que brinden seguridad y responsabilidad al sistema y el equipo que lo está desarrollando.

Como parte del desarrollo del plan de contingencia, se muestra la siguiente tabla detallando cuál es el riesgo que se desea enfrentar y las estrategias propuestas para contrarrestar los efectos de este.

|  |  |
| --- | --- |
| **Id** | **Riesgo** |
| 1 | Cambios en las historias de usuario |
| Estrategias | * Establecer un proceso de revisión formal antes de aceptar cualquier cambio en las historias de usuario. * Mantener una documentación clara de los requisitos acordados para facilitar la detección de cambios. * Implementar un sistema de notificación automática para alertar a todas las partes interesadas sobre los cambios propuestos. * Evaluar el impacto de los cambios en los plazos y costos del proyecto antes de su implementación. |

1. Plan de contingencia riesgo 1

|  |  |
| --- | --- |
| **Id** | **Riesgo** |
| 2 | Falta de recursos |
| Estrategias | * Mantener una reserva de recursos humanos y tecnológicos para cubrir posibles ausencias o picos de trabajo. * Establecer acuerdos con proveedores externos para obtener recursos adicionales cuando sea necesario. * Optimizar la carga de trabajo del equipo mediante una gestión eficiente de proyectos y asignación de tareas. * Realizar capacitaciones periódicas para mejorar las habilidades del equipo y aumentar la eficiencia. |

1. Plan de contingencia riesgo 2

|  |  |
| --- | --- |
| **Id** | **Riesgo** |
| 3 | Falta de comunicación cliente-desarrolladores |
| Estrategias | * Implementar herramientas de gestión de proyectos que faciliten la colaboración y la comunicación en tiempo real. * Programar reuniones periódicas para revisar el progreso del proyecto y discutir cualquier cambio en los requisitos. * Designar un representante del cliente y un líder de equipo de desarrollo para asegurar una comunicación fluida. * Utilizar canales de comunicación variados, como correos electrónicos, reuniones presenciales o videoconferencias, según sea necesario. |

1. Plan de contingencia riesgo 3

|  |  |
| --- | --- |
| **Id** | **Riesgo** |
| 4 | Problemas de seguridad |
| Estrategias | * Implementar medidas de seguridad robustas, como encriptación, cortafuegos y detección de intrusiones. * Realizar auditorías de seguridad y pruebas de penetración de manera regular. * Establecer un equipo de respuesta a incidentes para abordar y mitigar cualquier brecha de seguridad de manera rápida. * Educar a todo el equipo sobre prácticas seguras y concientizar a los usuarios sobre la importancia de mantener la seguridad de sus credenciales. |

1. Plan de contingencia riesgo 4

|  |  |
| --- | --- |
| **Id** | **Riesgo** |
| 5 | Falta de pruebas exhaustivas |
| Estrategias | * Establecer un proceso de pruebas riguroso para cada módulo del sistema. * Implementar la automatización de pruebas siempre que sea posible para garantizar una cobertura más completa. * Designar un equipo dedicado para realizar pruebas de rendimiento, de seguridad y de casos límite. * Establecer métricas claras para evaluar la efectividad de las pruebas y realizar revisiones periódicas del proceso de prueba. |

1. Plan de contingencia riesgo 5

|  |  |
| --- | --- |
| **Id** | **Riesgo** |
| 6 | Falta de SEO |
| Estrategias | * Realizar una auditoría SEO inicial para identificar áreas de mejora. * Incorporar prácticas SEO durante el desarrollo, como la optimización de contenido y la estructura del sitio. * Implementar herramientas de monitoreo para evaluar continuamente el rendimiento SEO. * Colaborar con expertos en SEO para ajustar estrategias según sea necesario. |

1. Plan de contingencia riesgo 6

|  |  |
| --- | --- |
| **Id** | **Riesgo** |
| 7 | Incumplimiento de plazos |
| Estrategias | * Establecer un cronograma realista con márgenes de tiempo para imprevistos. * Implementar metodologías de desarrollo ágiles para adaptarse a cambios y acelerar el desarrollo. * Realizar revisiones periódicas del progreso del proyecto y ajustar el cronograma según sea necesario. * Comunicar cualquier desviación en los plazos de manera oportuna y transparente. |

1. Plan de contingencia riesgo 7

|  |  |
| --- | --- |
| **Id** | **Riesgo** |
| 8 | Errores de desarrollo |
| Estrategias | * Implementar revisiones de código regulares y colaborativas. * Utilizar herramientas de análisis estático y dinámico de código para identificar posibles errores. * Establecer un protocolo de pruebas exhaustivas antes de la implementación de nuevas funcionalidades. * Mantener un sistema de control de versiones para rastrear cambios y facilitar la reversión en caso de errores críticos. |

1. Plan de contingencia riesgo 8

|  |  |
| --- | --- |
| **Id** | **Riesgo** |
| 9 | Pérdida de datos |
| Estrategias | * Implementar copias de seguridad regulares y automatizadas de la base de datos. * Establecer políticas de acceso y permisos para reducir el riesgo de errores humanos. * Tener un plan de recuperación de desastres que incluya la restauración de datos en caso de pérdida. * Realizar pruebas periódicas de restauración de datos para asegurar la integridad de las copias de seguridad. |

1. Plan de contingencia riesgo 9

|  |  |
| --- | --- |
| **Id** | **Riesgo** |
| 10 | Falta de capacitación al cliente |
| Estrategias | * Desarrollar materiales de capacitación detallados y accesibles para el cliente. * Ofrecer sesiones de capacitación presenciales o en línea según las necesidades del cliente. * Proporcionar recursos continuos, como manuales y tutoriales, para el aprendizaje autónomo. * Establecer un sistema de soporte rápido para abordar preguntas y problemas de capacitación. |

1. Plan de contingencia riesgo 10

|  |  |
| --- | --- |
| **Id** | **Riesgo** |
| 11 | Cambios regulatorios |
| Estrategias | * Mantenerse actualizado sobre los marcos legales relevantes y anticipar posibles cambios. * Colaborar con expertos legales para asegurar el cumplimiento normativo. * Establecer un proceso de revisión legal regular para identificar y abordar posibles riesgos. * Tener un plan de acción claro en caso de cambios regulatorios, incluyendo la implementación de actualizaciones rápidas si es necesario. |

1. Plan de contingencia riesgo 11

|  |  |
| --- | --- |
| **Id** | **Riesgo** |
| 12 | Dependencia de terceros |
| Estrategias | * Realizar una evaluación exhaustiva de la confiabilidad y estabilidad de los proveedores de servicios externos. * Establecer acuerdos de nivel de servicio (SLA) claros con los proveedores. * Implementar una estrategia de contingencia que incluya la posibilidad de cambiar a un proveedor alternativo si es necesario. * Mantener una comunicación regular con los proveedores para estar al tanto de cambios, actualizaciones y posibles problemas. |

1. Plan de contingencia riesgo 12

## **4.3 Historias de usuario**

De acuerdo con la metodología XP, las historias de usuario son la herramienta que permite obtener las principales funcionalidades del sistema. Realizando la investigación pertinente, se obtuvo que existen 3 tipos de historias de usuario con las que podemos definir la importancia con la que se tendrá que desarrollar cada funcionalidad descrita en la historia de usuario.

Es por eso por lo que, para esta sección del documento, se definen cuáles son las historias de usuario considerando 3 categorías:

* Historias de usuario de mayor prioridad (alta)
* Historias de usuario de mediana prioridad (Media)
* Historias de usuario de menos prioridad (Baja)

A continuación, se presentan las tablas correspondientes a las historias de usuario detallando en qué consisten, quién es el encargado, el tipo de prioridad y los criterios de validación.

|  |  |
| --- | --- |
| **Nombre: Iniciar sesión** | |
| Identificador: H1 | Usuario: Gerente, empleado |
| Prioridad: Alta | Riesgo en desarrollo: Medio |
| Responsable: Muñoz Ramírez Gabriel de Jesús. | |
| Descripción: Como usuario gerente o usuario empleado, quiero poder iniciar sesión para acceder al sistema y utilizar sus funcionalidades. | |
| Criterio de validación: el usuario puede iniciar sesión con éxito es dirigido según sus permisos. | |

1. Historia de usuario 1

|  |  |
| --- | --- |
| **Nombre: Cerrar sesión** | |
| Identificador: H2 | Usuario: Gerente, empleado |
| Prioridad: Alta | Riesgo en desarrollo: Bajo |
| Responsable: Muñoz Ramírez Gabriel de Jesús. | |
| Descripción: Como usuario gerente o usuario empleado, quiero poder cerrar sesión para no comprometer mis credenciales para ingresar al sistema. | |
| Criterio de validación: Permite a los usuarios Gerente y Empleado finalizar sesión, garantizando la protección de sus credenciales y la imposibilidad de acceder al sistema sin autorización. | |

1. Historia de usuario 2

|  |  |
| --- | --- |
| **Nombre: Visualización de Degradación de Producto.** | |
| Identificador: H3. | Usuario: Gerente. |
| Prioridad: Alta. | Riesgo en desarrollo: Medio. |
| Responsable: Sánchez Becerra Ernesto Daniel. | |
| Descripción: Como gerente, quiero poder visualizar gráficas detalladas que muestren la degradación de un producto específico a lo largo del tiempo para evaluar su rendimiento en términos de calidad y durabilidad. | |
| Criterio de validación: Se pueden generar gráficas para productos específicos que reflejen la degradación a lo largo del tiempo. | |

1. Historia de usuario 3

|  |  |
| --- | --- |
| **Nombre: Análisis de obsolescencia.** | |
| Identificador:H4. | Usuario: Gerente. |
| Prioridad: Alta. | Riesgo en Desarrollo: Medio. |
| Responsable: Garzón Domínguez Gerardo Ismael. | |
| Descripción: Como gerente, quiero poder analizar la tasa general de obsolescencia de los productos en el inventario para comprender cuán rápidamente se vuelven obsoletos o no utilizables. | |
| Criterio de Validación: Se presenta una métrica clara de la tasa de obsolescencia en el sistema. | |

1. Historia de usuario 4

|  |  |
| --- | --- |
| **Nombre: Métrica de eficiencia en gestión de inventario.** | |
| Identificador: H5 | Usuario: Gerente |
| Prioridad: Alta. | Riesgo en desarrollo: Medio. |
| Responsable: Muñoz Ramírez Gabriel de Jesús. | |
| Descripción: Como gerente, necesito una métrica que indique el porcentaje de productos perdidos en comparación con el inventario total para evaluar la eficiencia en la gestión del inventario. | |
| Criterio de validación: Se muestra con precisión el porcentaje de pérdidas en comparación con el inventario total. | |

1. Historia de usuario 5

|  |  |
| --- | --- |
| **Nombre: Análisis de impacto financiero por pérdidas.** | |
| Identificador: H6 | Usuario: Gerente |
| Prioridad: Alta | Riesgo en desarrollo: Medio |
| Responsable: Sánchez Becerra Ernesto Daniel. | |
| Descripción: Como gerente, quiero calcular y visualizar la pérdida económica neta asociada con la pérdida de productos para comprender el impacto financiero y guiar las decisiones sobre reposición y gestión de inventario. | |
| Criterio de validación: El sistema presenta de manera clara y precisa la pérdida económica neta asociada con las pérdidas de productos. | |

1. Historia de usuario 6

|  |  |
| --- | --- |
| **Nombre: Clasificación ABC de productos.** | |
| Identificador: H7 | Usuario: Gerente |
| Prioridad: Alta | Riesgo en desarrollo: Medio |
| Responsable: Garzón Domínguez Gerardo Ismael. | |
| Descripción: Como gerente, quiero que el sistema clasifique los productos en categorías ABC según su importancia y demanda para priorizar la gestión del inventario. | |
| Criterio de validación: Los productos se clasifican correctamente en categorías ABC. | |

1. Historia de usuario 7

|  |  |
| --- | --- |
| **Nombre: Visualización de ingresos por producto.** | |
| Identificador: H8 | Usuario: Gerente |
| Prioridad: Alta | Riesgo en desarrollo: Medio |
| Responsable: Muñoz Ramírez Gabriel de Jesús. | |
| Descripción: Como gerente, necesito ver el porcentaje de ingresos asociado con cada producto o categoría de productos para identificar las fuentes principales de ingresos y concentrar esfuerzos estratégicos. | |
| Criterio de validación: El sistema muestra con precisión el porcentaje de ingresos para cada producto o categoría. | |

1. Historia de usuario 8

|  |  |
| --- | --- |
| **Nombre: Establecer puntos de reorden.** | |
| Identificador: H9 | Usuario: Gerente |
| Prioridad: Alta | Riesgo en desarrollo: Alto |
| Responsable: Sánchez Becerra Ernesto Daniel. | |
| Descripción: Como gerente, quiero poder establecer puntos de reorden para los productos en el inventario, de modo que se generen alertas automáticas cuando los niveles de existencias alcancen cierto umbral, facilitando la reposición oportuna. | |
| Criterio de validación: El sistema genera alertas correctamente al alcanzar los puntos de reorden. | |

1. Historia de usuario 9

|  |  |
| --- | --- |
| **Nombre: Identificación de productos con bajo stock.** | |
| Identificador: H10 | Usuario: Gerente |
| Prioridad: Alta | Riesgo en desarrollo: Medio |
| Responsable: Garzón Domínguez Gerardo Ismael. | |
| Descripción: Como gerente, quiero recibir notificaciones claras sobre los productos que tienen niveles de existencias por debajo de un umbral crítico, para tomar medidas rápidas y evitar la falta de stock | |
| Criterio de validación: Se reciben notificaciones precisas sobre los productos con bajo stock. | |

1. Historia de usuario 10

|  |  |
| --- | --- |
| **Nombre: Registro de ingreso de nuevos productos.** | |
| Identificador: H11 | Usuario: Empleado, Gerente |
| Prioridad: Alta | Riesgo en desarrollo: Medio |
| Responsable: Muñoz Ramírez Gabriel de Jesús. | |
| Descripción: Como gerente, necesito un proceso eficiente para registrar la llegada de nuevos productos al inventario, asegurando una actualización precisa de los niveles de existencias. | |
| Criterio de validación: Los nuevos productos se registran correctamente en el inventario. | |

1. Historia de usuario 11

|  |  |
| --- | --- |
| **Nombre: Generar pedidos de reposición.** | |
| Identificador: H12 | Usuario: Gerente. |
| Prioridad: Alta. | Riesgo en desarrollo: Medio. |
| Responsable: Sánchez Becerra Ernesto Daniel. | |
| Descripción: Como gerente, quiero poder crear pedidos de reposición para los productos que están por debajo de los puntos de reorden, asegurando una reposición eficiente y oportuna. | |
| Criterio de validación: Los pedidos se generan correctamente y reflejan las necesidades reales de reposición. | |

1. Historia de usuario 12

|  |  |
| --- | --- |
| **Nombre: Creación y edición de almacenes.** | |
| Identificador: H13 | Usuario: Gerente. |
| Prioridad: Alta | Riesgo en desarrollo: Alto. |
| Responsable: Garzón Domínguez Gerardo Ismael. | |
| Descripción: Como gerente, quiero poder crear nuevos almacenes y editar o eliminar los existentes para adaptar la estructura del almacenamiento según las necesidades cambiantes del inventario. | |
| Criterio de validación: El sistema permite la creación, edición o eliminación de almacenes y los cambios se muestra de manera visual en 2D. | |

1. Historia de usuario 13

|  |  |
| --- | --- |
| **Nombre: Gestión completa de anaqueles.** | |
| Identificador: H14 | Usuario: Gerente. |
| Prioridad: Alta | Riesgo en desarrollo: Alto. |
| Responsable: Muñoz Ramírez Gabriel de Jesús. | |
| Descripción: Como gerente, quiero realizar las operaciones agregar, editar o eliminar anaqueles según las necesidades específicas de organización. | |
| Criterio de validación: El sistema permite la creación, edición o eliminación de anaqueles y los cambios se muestra de manera visual en 2D. | |

1. Historia de usuario 14

|  |  |
| --- | --- |
| **Nombre: Gestión de productos en anaqueles.** | |
| Identificador: H15 | Usuario: Gerente. |
| Prioridad: Alta | Riesgo en desarrollo: Alto. |
| Responsable: Sánchez Becerra Ernesto Daniel. | |
| Descripción: Como gerente, quiero poder agregar, editar y eliminar productos directamente en los anaqueles para una gestión eficiente y precisa del inventario. | |
| Criterio de validación: El sistema permite la creación, edición o eliminación de productos y los cambios se muestra de manera visual en 2D. | |

1. Historia de usuario 15

|  |  |
| --- | --- |
| **Nombre: Optimización de distribución de productos.** | |
| Identificador: H16 | Usuario: Gerente. |
| Prioridad: Alta | Riesgo en desarrollo: Alto. |
| Responsable: Garzón Domínguez Gerardo Ismael. | |
| Descripción: Como gerente, quiero contar con una función de optimización que ajuste automáticamente la disposición de los productos en los anaqueles para maximizar el espacio y facilitar el acceso. | |
| Criterio de validación: La optimización de la disposición de productos se realiza de manera efectiva, mejorando la eficiencia del almacenamiento. | |

1. Historia de usuario 16

|  |  |
| --- | --- |
| **Nombre: Administración de usuarios.** | |
| Identificador: H17 | Usuario: Gerente. |
| Prioridad: Alta. | Riesgo en desarrollo: Bajo. |
| Responsable: Muñoz Ramírez Gabriel de Jesús. | |
| Descripción: Como gerente, necesito realizar operaciones de agregar nuevos usuarios, ver la información existente, actualizar datos y desactivar cuentas según sea necesario. | |
| Criterio de validación: Se pueden realizar las operaciones de manera efectiva, asegurando la integridad y seguridad de la información de cada cuenta. | |

1. Historia de usuario 17

|  |  |
| --- | --- |
| **Nombre: Venta de productos.** | |
| Identificador: H18 | Usuario: Empleado. |
| Prioridad: Alta. | Riesgo en desarrollo: Bajo. |
| Responsable: Sánchez Becerra Ernesto Daniel. | |
| Descripción: Como empleado, necesito poder realizar ventas registrando la salida de productos del inventario para completar transacciones con los clientes. | |
| Criterio de validación: Los niveles de inventario se actualizan correctamente después de realizar una venta. | |

1. Historia de usuario 18

|  |  |
| --- | --- |
| **Nombre: Recuperar contraseña.** | |
| Identificador: H19 | Usuario: Gerente, Empleado. |
| Prioridad: Media | Riesgo en Desarrollo: Bajo |
| Responsable: Garzón Domínguez Gerardo Ismael. | |
| Descripción: Como usuario gerente o usuario empleado, quiero tener la capacidad de recuperar mi contraseña en caso de olvido | |
| Criterio de validación: Los usuarios, ya sea gerente o empleado, deben poder recuperar su contraseña a través de un proceso que involucre la verificación por correo electrónico y la creación de una nueva contraseña. | |

1. Historia de usuario 19

|  |  |
| --- | --- |
| **Nombre: Pantalla de inicio.** | |
| Identificador: H20 | Usuario: Gerente |
| Prioridad: Media | Riesgo en desarrollo: Medio |
| Responsable: Muñoz Ramírez Gabriel de Jesús. | |
| Descripción: Como usuario gerente, necesito tener una pantalla de inicio, que tenga elementos visuales y acceso rápido a las funciones clave del sistema. | |
| Criterio de validación: La pantalla de inicio del sistema para el usuario gerente debe presentar elementos visuales intuitivos y proporcionar acceso rápido a las funciones clave de manera eficiente. | |

1. Historia de usuario 20

|  |  |
| --- | --- |
| **Nombre: Representación visual de demanda de productos.** | |
| Identificador: H21 | Usuario: Gerente. |
| Prioridad: Media | Riesgo en desarrollo: Medio |
| Responsable: Sánchez Becerra Ernesto Daniel. | |
| Descripción: Como gerente, quiero visualizar gráficas que representen la demanda de productos a lo largo del tiempo, incluyendo tendencias de venta | |
| Criterio de validación: Las gráficas ofrecen una representación visual clara y precisa de la demanda de productos. | |

1. Historia de usuario 21

|  |  |
| --- | --- |
| **Nombre: Implementación de “Cross-Docking”.** | |
| Identificador: H22 | Usuario: Empleado, Gerente |
| Prioridad: Medio | Riesgo en desarrollo: Medio |
| Responsable: Garzón Domínguez Gerardo Ismael. | |
| Descripción: Como gerente, quiero implementar la técnica de Cross-Docking para agilizar el proceso de ingreso de mercancía al transferir directamente los productos desde el área de recepción al mostrador, minimizando el tiempo de almacenamiento. | |
| Criterio de validación: Los nuevos productos se registran correctamente en el inventario. | |

1. Historia de usuario 22

|  |  |
| --- | --- |
| **Nombre: Seguimiento de rotación de inventario.** | |
| Identificador: H23 | Usuario: Gerente |
| Prioridad: Medio | Riesgo en desarrollo: Medio |
| Responsable: Muñoz Ramírez Gabriel de Jesús. | |
| Descripción: Como gerente, quiero tener un seguimiento claro de la rotación de inventario para identificar productos con baja rotación y tomar decisiones informadas sobre estrategias de marketing y liquidación. | |
| Criterio de validación: El sistema proporciona informes precisos sobre la rotación de inventario. | |

1. Historia de usuario 23

|  |  |
| --- | --- |
| **Nombre: Visualización del porcentaje de “stock-out”.** | |
| Identificador: H24 | Usuario: Gerente. |
| Prioridad: Media | Riesgo en desarrollo: Medio. |
| Responsable: Sánchez Becerra Ernesto Daniel. | |
| Descripción: Como gerente, quiero tener acceso al porcentaje de stock-out para evaluar la efectividad de las estrategias de reposición y minimizar la falta de stock. | |
| Criterio de validación: El sistema muestra el porcentaje de stock-out de manera precisa y actualizada. | |

1. Historia de usuario 24

|  |  |
| --- | --- |
| **Nombre: Seguimiento de la tasa de caducidad.** | |
| Identificador: H25 | Usuario: Gerente. |
| Prioridad: Media | Riesgo en desarrollo: Medio. |
| Responsable: Garzón Domínguez Gerardo Ismael. | |
| Descripción: Como gerente, quiero tener información detallada sobre la tasa de caducidad de los productos perecederos para gestionar eficientemente la reposición y minimizar las pérdidas económicas. | |
| Criterio de validación: El sistema muestra informes precisos sobre la tasa de caducidad de productos perecederos. | |

1. Historia de usuario 25

|  |  |
| --- | --- |
| **Nombre: Cálculo del tiempo promedio de estancia.** | |
| Identificador: H26 | Usuario: Gerente. |
| Prioridad: Media | Riesgo en desarrollo: Medio. |
| Responsable: Muñoz Ramírez Gabriel de Jesús. | |
| Descripción: Como gerente, quiero conocer el tiempo promedio que los productos perecederos permanecen en el inventario para mejorar las estrategias de gestión y reducir las pérdidas. | |
| Criterio de validación: El sistema calcula con precisión el tiempo promedio de estancia de los productos perecederos. | |

1. Historia de usuario 26

|  |  |
| --- | --- |
| **Nombre: Edición de productos.** | |
| Identificador: H27 | Usuario: Gerente. |
| Prioridad: Media | Riesgo en desarrollo: Medio. |
| Responsable: Sánchez Becerra Ernesto Daniel. | |
| Descripción: Como gerente, quiero poder editar la información detallada de los productos en el inventario, lo que incluye ajustar descripciones, precios, y otras características específicas. | |
| Criterio de validación: Los cambios realizados en la edición de productos se reflejan correctamente en el sistema, y la información detallada es precisa. | |

1. Historia de usuario 27

|  |  |
| --- | --- |
| **Nombre: Retiro de productos del inventario.** | |
| Identificador: H28 | Usuario: Gerente. |
| Prioridad: Media | Riesgo en desarrollo: Medio. |
| Responsable: Garzón Domínguez Gerardo Ismael. | |
| Descripción: Como gerente, necesito la capacidad de dar de baja o retirar productos del inventario, ya sea debido a obsolescencia, daño o cualquier otra razón que requiera la eliminación del producto. | |
| Criterio de validación: Los productos dados de baja se eliminan del inventario de manera efectiva, y se registra adecuadamente la baja. | |

1. Historia de usuario 28

|  |  |
| --- | --- |
| **Nombre: Búsqueda de productos.** | |
| Identificador: H29 | Usuario: Empleado. |
| Prioridad: Media. | Riesgo en desarrollo: Bajo. |
| Responsable: Muñoz Ramírez Gabriel de Jesús. | |
| Descripción: Como empleado, quiero poder buscar productos en el sistema para proporcionar información precisa a los clientes y agilizar el proceso de ventas. | |
| Criterio de validación: Se muestra un resultado preciso al buscar productos por nombre, categoría o código de barras. | |

1. Historia de usuario 29

|  |  |
| --- | --- |
| **Nombre: Diseño responsivo** | |
| Identificador: H30 | Usuario: Empleado, Gerente |
| Prioridad: Media | Riesgo en desarrollo: Medio |
| Responsable: Garzón Domínguez Gerardo Ismael. | |
| Descripción: Como usuario empleado/gerente, quiero que el sistema me permita acceder al mismo desde diferentes dispositivos. | |
| Criterio de validación: se permite desplegar el sistema desde distintos tipos de dispositivos como celular, Tablet, ordenador, laptop. | |

1. Historia de usuario 31

|  |  |
| --- | --- |
| **Nombre: Paleta de colores** | |
| Identificador: H31 | Usuario: Empleado/Gerente |
| Prioridad: Baja | Riesgo en desarrollo: Baja |
| Responsable: Muñoz Ramírez Gabriel de Jesús. | |
| Descripción: Como usuario empleado/gerente, quiero contar con una paleta de colores visualmente aceptable al momento de interactuar en el sistema. | |
| Criterio de validación: se mostrará una paleta de colores adecuada y que haga contraste con los módulos del sistema. | |

1. Historia de usuario 32

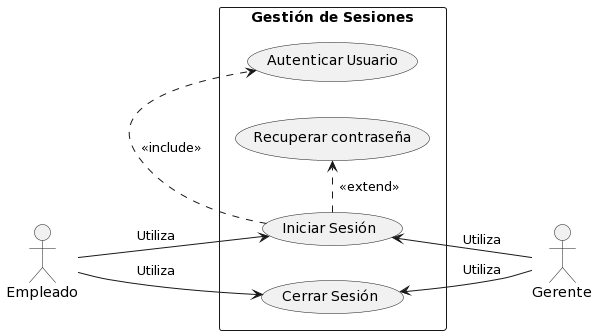
|  |  |
| --- | --- |
| **Nombre: Barra de navegación** | |
| Identificador: H32 | Usuario: Empleado/Gerente |
| Prioridad: Baja | Riesgo en desarrollo: Baja |
| Responsable: Sánchez Becerra Ernesto Daniel. | |
| Descripción: Como usuario empleado/gerente, quiero contar con una barra de navegación que me permita desplazar entre los distintos módulos del sistema. | |
| Criterio de validación: se muestra una barra de navegación del sistema para facilitar la selección del módulo que se requiera utilizar. | |

1. Historia de usuario 33

|  |  |
| --- | --- |
| **Nombre: Botones de selección** | |
| Identificador: H33 | Usuario: Empleado/Gerente |
| Prioridad: Baja | Riesgo en desarrollo: Baja |
| Responsable: Muñoz Ramírez Gabriel de Jesús. | |
| Descripción: Como usuario empleado/gerente, quiero contar con botones que me permitan hacer uso de las diferentes herramientas que ofrece el sistema. | |
| Criterio de validación: se muestran y habilitan botones que brinden al usuario las diferentes funcionalidades del sistema con el que podrá interactuar y utilizar de acuerdo con las necesidades que tenga. | |

1. Historia de usuario 34

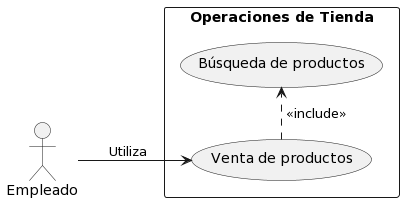
## **4.4 Casos de uso.**



1. Diagrama de casos de uso 1

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Caso de Uso | Actores | Resumen | Tipo | Referencias Cruzadas |
| Iniciar Sesión | Gerente, Empleado | Usuarios inician sesión en el sistema. | Primario | Autenticar Usuario, Recuperar Contraseña |
| Autenticar Usuario | Gerente, Empleado | Verificar la identidad del usuario durante el inicio de sesión. | Incluido | Iniciar Sesión |
| Cerrar Sesión | Gerente, Empleado | Usuarios cierran sesión en el sistema. | Primario | - |
| Recuperar Contraseña | Gerente, Empleado | Usuarios recuperan sus contraseñas en caso de olvido. | Extendido | Iniciar Sesión |

1. Descripción del diagrama de caso de uso 1



1. Diagrama de casos de uso 2

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Caso de Uso | Actores | Resumen | Tipo | Referencias Cruzadas |
| Venta de productos | Empleado | Realizar ventas de productos. | Primario | Búsqueda de productos |
| Búsqueda de productos | Empleado | Buscar productos en el inventario. | Incluido | - |

1. Descripción del diagrama de casos de uso 2

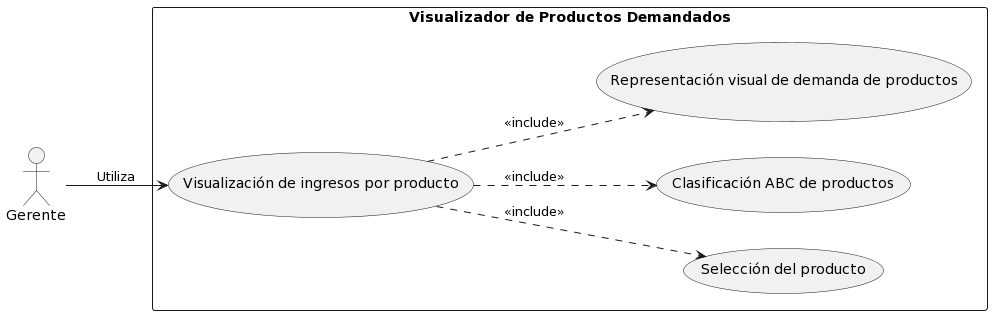
Diagrama

Descripción generada automáticamente

1. Diagrama de casos de uso 3

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Caso de Uso | Actores | Resumen | Tipo | Referencias cruzadas |
| Tasa de daños. | Gerente | El gerente quiere evaluar y gestionar la tasa de daños de productos. | Primario | Análisis de impacto financiero por pérdidas, Análisis de obsolescencia. |
| Métrica de eficiencia en gestión de inventario. | Gerente | El gerente busca evaluar la eficiencia en la gestión de inventario mediante métricas específicas. | Primario | - |
| Análisis de impacto financiero por pérdidas. | Gerente | El gerente necesita calcular y visualizar el impacto financiero de las pérdidas de productos. | Incluido | - |
| Visualización de degradación de producto. | Gerente | El gerente desea visualizar gráficas detalladas que muestren la degradación de productos a lo largo del tiempo. | Primario | - |
| Análisis de obsolescencia. | Gerente | El gerente busca analizar la tasa general de obsolescencia de los productos en el inventario. | Incluido | - |
| Seguimiento de la tasa de caducidad. | Gerente | El gerente desea realizar un seguimiento detallado de la tasa de caducidad de productos perecederos. | Primario | - |
| Cálculo del tiempo promedio de estancia. | Gerente | El gerente quiere calcular el tiempo promedio que los productos perecederos permanecen en el inventario. | Primario | - |

1. Descripción del diagrama de casos de uso 3



1. Diagrama de casos de uso 4

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Caso de Uso | Actores | Resumen | Tipo | Referencias Cruzadas |
| Visualización de ingresos por producto | Gerente | Visualizar el porcentaje de ingresos asociado con cada producto o categoría. | Primario | Selección del producto, Clasificación ABC de productos, Representación visual de demanda de productos |
| Selección del producto. | Gerente | Seleccionar un producto para ver detalles específicos. | Incluido | Visualización de ingresos por producto |
| Clasificación ABC de productos. | Gerente | Clasificar productos en categorías ABC según importancia y demanda. | Incluido | Visualización de ingresos por producto |
| Representación visual de demanda de productos. | Gerente | Ver gráficas que representan la demanda de productos a lo largo del tiempo. | Incluido | Visualización de ingresos por producto |

1. Descripción del diagrama de casos de uso 4

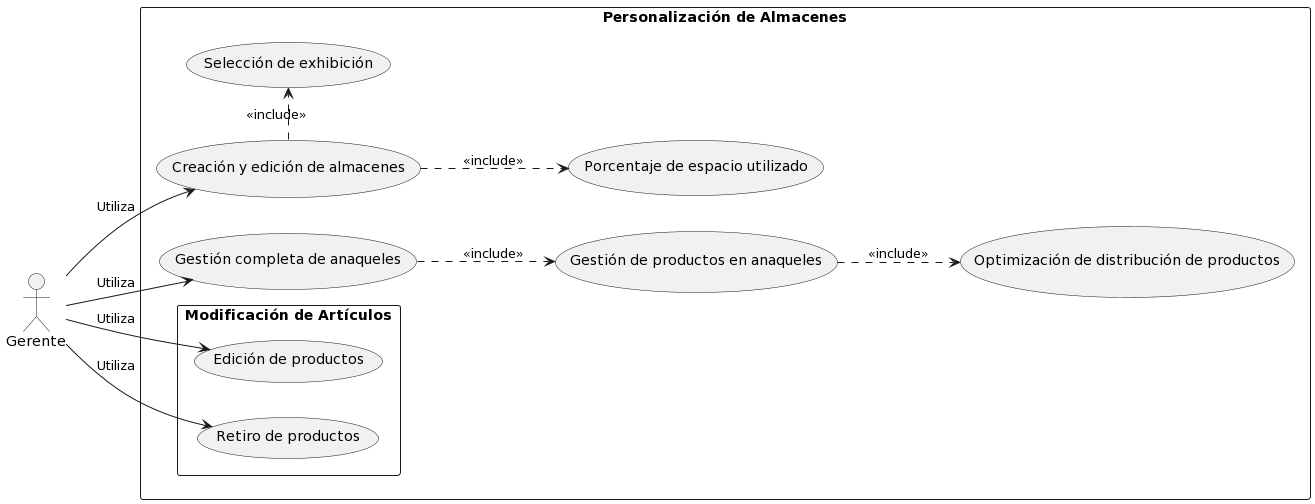
Diagrama

Descripción generada automáticamente

1. Diagrama de casos de uso 5

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Caso de Uso | Actores | Resumen | Tipo | Referencias Cruzadas |
| Registro de ingreso de nuevos productos | Gerente | Registrar la llegada de nuevos productos al inventario. | Primario | - |
| Seguimiento de rotación de inventario | Gerente | Realizar un seguimiento claro de la rotación de inventario. | Primario | Establecer puntos de reorden |
| Gestión de entregas | Gerente | Generar pedidos de reposición para productos por debajo de puntos de reorden. | Primario | - |
| Identificación de productos con bajo stock | Gerente | Recibir notificaciones sobre productos con niveles de existencias bajos. | Primario | Visualización del porcentaje de stock-out |
| Implementación de Cross-Docking | Gerente | Agilizar el ingreso de mercancía transfiriendo productos directamente al mostrador. | Incluido | - |
| Establecer puntos de reorden | Gerente | Establecer puntos de reorden para productos con alertas automáticas. | Incluido | Generar pedidos de reposición |
| Generar pedidos de reposición | Gerente | Crear pedidos de reposición para productos bajo puntos de reorden. | Incluido | Visualización del porcentaje de stock-out |
| Visualización del porcentaje de stock-out | Gerente | Acceder al porcentaje de stock-out y evaluar estrategias de reposición. | Incluido | - |

1. Descripción del diagrama de casos de uso 5



1. Diagrama de casos de uso 6

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Caso de Uso | Actores | Resumen | Tipo | Referencias Cruzadas |
| Creación y edición de almacenes | Gerente | Crear nuevos almacenes y editar o eliminar existentes. | Primario | Selección edición de almacenes, Porcentaje de espacio utilizado, Selección de exhibición |
| Selección edición de almacenes | Gerente | Operaciones de selección y edición de almacenes. | Incluido | - |
| Porcentaje de espacio utilizado | Gerente | Visualizar el porcentaje de espacio utilizado en los almacenes. | Incluido | - |
| Selección de exhibición | Gerente | Seleccionar la forma de exhibición de los productos. | Incluido | - |
| Gestión completa de anaqueles | Gerente | Operaciones de agregar, editar o eliminar anaqueles. | Primario | Gestión de productos en anaqueles |
| Gestión de productos en anaqueles | Gerente | Agregar, editar y eliminar productos directamente en anaqueles. | Incluido | - |
| Optimización de distribución de productos | Gerente | Ajustar automáticamente la disposición de productos para maximizar espacio y facilitar acceso. | Primario | - |
| Edición de productos | Gerente | Editar información detallada de productos en el inventario. | Primario | Retiro de productos |
| Retiro de productos | Gerente | Dar de baja o retirar productos del inventario. | Incluido | - |

1. Descripción del diagrama de casos de uso 6

Diagrama

Descripción generada automáticamente

1. Diagrama de casos de uso 7

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Caso de Uso | Actores | Resumen | Tipo | Referencias Cruzadas |
| Administración de usuarios | Gerente | Realizar operaciones de como agregar, eliminar, actualizar usuarios. | Primario | - |

1. Descripción del diagrama de casos de uso 7

## **4.5 Requerimientos de la base de datos**

La base de datos es una de las partes fundacionales del proyecto, su utilidad principal es proporcionar una forma de registrar y controlar datos mediante su interoperabilidad en un lenguaje host que permita la integración de esta con la aplicación, además de ofrecer cuatro características fundamentales para que la base de datos siempre se encuentre en un estado válido; atomicidad, consistencia, aislamiento y durabilidad, (ACID por su sigla en inglés) mediante el sistema gestor de base de datos (Oracle MySQL). [61]

### **4.5.1 Requerimientos de datos**

El tipo de datos a almacenar, se encuentran codificados en su mayoría mediante lo que se podría considerar como tipos primitivos en varios lenguajes de programación, es decir los datos que requerimos almacenar para la aplicación son una colección datos numéricos enteros y flotantes, booleanos y cadenas.

Los datos del sistema son en su mayoría estructurados, por lo que se utilizará un base de datos relacional, que ofrezca una base sólida para realizar transacciones donde siempre se conozcan los atributos que puede tener un dato. Sin embargo, debido a la complejidad que presentan ciertos aspectos en las tiendas de conveniencia, como productos con atributos adicionales los cuales no fueron contemplados en el diseño estructurado o los distintos tipos de ofertas que también se pueden aplicar además de aquellas por descuento (vea 4.5.2); Es que decidimos agregar columnas dentro de estas mismas entidades que ofrezcan la capacidad de almacenar datos no estructurados, esto para facilitar la implementación de posibles nuevas funcionalidades que podrían solicitar los clientes y que requieran el uso de nuevos datos no contemplados en el esquema relacional.

### **4.5.1 Volumen de datos**

El volumen de datos a considerar depende directamente de la cantidad de transacciones que realiza la tienda de conveniencia en un periodo de tiempo. Según datos oficiales de Oxxo, estos atienden 13 millones de personas diario y cuentan con al menos 21000 sucursales en la República Mexicana [62]. Si asumimos que el volumen de personas que atiende cada sucursal es uniforme, obtenemos que:

Según datos de “El Financiero” [63], un Oxxo genera 30000MXN diarios y cada cliente hace compras de al menos 50MXN. Tomaremos como referencia los primeros 30000MXN que puede generar un Oxxo de manera diaria mediante las primeras 620 personas que pueden comprar en una sucursal.

Según un análisis que realizamos utilizando la base de datos de precios promedio del INEGI [28], obtenemos que el precio promedio de productos en un rango de costo máximo 100MXN, es de 38MXN. Mediante este límite de 100MXN exploramos la variedad de productos de clase C (según Pareto): aquellos que no tienen un valor tan alto en el inventario, pero conforman la mayor parte de este. Además, teniendo la estadística de que cada cliente gasta como mínimo 50MXN, es plausible asumir que las compras tienden a ser, por lo general, de varios productos de menor precio.

Mediante todos los datos mencionados anteriormente, obtenemos el modelo siguiente, donde buscamos maximizar la cantidad de productos vendidos diarios (Z):

Donde:

Aplicando el método GRG No Linear (Generalized Reduced Gradient Non Linear), mediante la asistencia del solucionador de Excel [64], obtenemos los siguientes valores para las variables P y C:

Es decir, que para maximizar el número de productos vendidos en un Oxxo de manera diaria dentro del rango de las primeras posibles 620 personas que compren, se necesita que la tienda tenga al menos 312 clientes diarios que compren 2 productos cada uno. Mediante esta estimación podemos pasar al siguiente paso para el análisis de volumen de datos; La cantidad de datos que genera nuestro sistema por transacción de venta.

Según la información anterior, la documentación oficial de Oracle MySQL [65], y partiendo del diagrama E-R del punto 4.5.2, obtenemos que una transacción de venta constará de las siguientes instancias de relación:

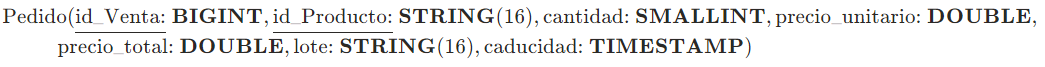
| **TIPO DE DATO** | **Espacio requerido** |
| --- | --- |
| TINYINT | 1 byte |
| SMALLINT | 2 bytes |
| MEDIUMINT | 3 bytes |
| INT, INTEGER | 4 bytes |
| BIGINT | 8 bytes |
| DOUBLE [PRECISION], REAL | 8 bytes |
| TIMESTAMP | 4 bytes |
| VARCHAR (16) | 16 + 1 bytes |
| DECIMAL(10,2) | 4 + 1 bytes |

1. Espacio de almacenamiento requerido en bytes para varios tipos de datos en Oracle MySQL

Una instancia de relación “Venta”:



Y 5 instancias de relación “Pedido”:



Por lo tanto, el volumen de datos generados por transacción y periodos de tiempo está dado por:

A number with numbers and symbols

Description automatically generated with medium confidence

Como se puede observar, el volumen de datos es relativamente bajo, con un aproximado de 16MB de datos en transacciones de ventas anuales, esto, sin considerar las transacciones que involucren llegada de stock, ofertas, mermas, y el resto de las tablas de la base de datos. Si Kv-Wims fuese a ser utilizado durante un año para el 10% transacciones de venta totales de Oxxo (2100 sucursales), el volumen de datos sería de 33.257972Gb, por lo que, incluso bastaría el segundo tamaño de disco duro (HDD) ofrecido por Microsoft Azure, el cual es de 64Gb. [66]

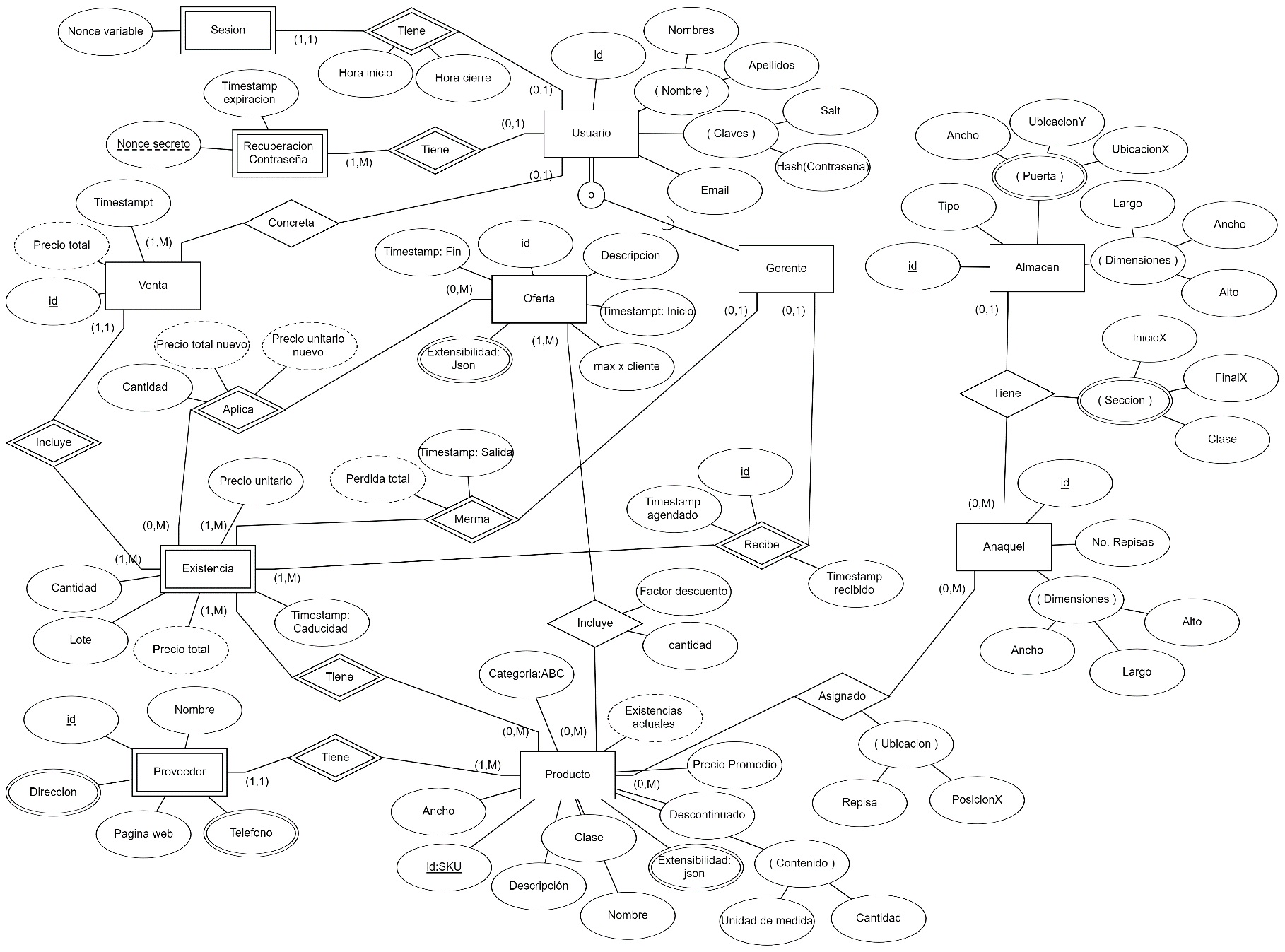
En conclusión, el volumen de datos estimado que puede generar una tienda de conveniencia utilizando el sistema Kv-wims es pequeño. Por lo tanto, consideramos que, respecto a volumen de datos, la necesidad de almacenamiento masivo no es una principal preocupación para el desarrollo del proyecto, sin embargo, en dado caso que sea necesario almacenar más datos de los que se estimaron, los servicios de la nube como Microsoft Azure ofrecen soluciones escalables o flexibles. [67]

### **4.5.2 Diagrama entidad-relación**

A continuación, se describe el esquema entidad relación propuesto para la base de datos, el cual está pensado para cubrir los requerimientos transaccionales mínimos del sistema. Primero, describimos las entidades, relaciones, y sus atributos principales seguidos de una breve justificación. Los atributos de identificación (id) no serán explicados repetidamente, ya que, estos tienen la utilidad de simplificar la llave primaria, y por extensión, simplifican otros aspectos de diseño de la base de datos, como la normalización, u otras operaciones como consultas mediante SQL.

* **Entidad almacén:** Los almacenes son una entidad clave para Kv-wims, mediante estos datos será posible generar las visualizaciones en el lado del cliente.
  + **Tipo:** Este atributo es necesario ya que, consideramos que algunas tiendas de conveniencia pueden tener más de un almacén, uno trasero y el de mostrador, mediante este atributo es posible distinguir ambas instancias de almacén.
  + **Dimensiones;** **(Largo, ancho, alto):** Este atributo compuesto almacena el tamaño del almacén, es necesario para realizar validaciones sobre que anaqueles pueden ser instanciados dentro de este o no.
  + **Relación identificada con entidad débil (tiene puerta):** Los almacenes deben tener al menos una puerta de la cual es necesario saber su ancho, y, mediante la relación identificada (tiene), es que se conoce la posición de la entrada dentro del perímetro del almacén.
* **Entidad anaquel:** Los anaqueles son otra parte grande de la visualización de Kv-wims, ya que, estos almacenan y clasifican los productos.
  + **No. Repisas:** Este atributo indica el número de niveles horizontales que tiene un anaquel.
  + **Dimensiones;** **(Largo, ancho, alto):** Este atributo compuesto almacena el tamaño de un anaquel, es necesario para realizar validaciones sobre si este puede ser instanciado dentro de un almacén o no.
* **Relación (almacén tiene anaqueles):** Mediante esta relación identificamos que anaqueles existen dentro de un almacén.
  + **Sección**: Mediante este atributo complejo (multivaluado y compuesto), es posible identificar secciones de un anaquel instanciado dentro de un almacén, es decir, estas secciones sirven para clasificar productos en forma de columna dentro del anaquel.
* **Entidad producto**: Los productos son una de las entidades de mayor importancia en cualquier sistema que maneje inventarios. En el caso de nuestro sistema, los productos son la entidad con más atributos, ya que, estos representan la parte de mayor complejidad puesto que son una unidad fundamental en varios tipos de transacciones principales para el funcionamiento de una tienda de conveniencia, como ventas y recibos de stock.
  + **Ancho** Este atributo registra el tamaño de un producto cuando este es acomodado en su dimensión más larga, esto considerando que los productos suelen ser acomodados de esta manera y con varias otras instancias del mismo producto hacia la parte trasera del anaquel.
  + **Precio:** Indica el precio del producto, lo único que se espera para este atributo en las instancias de productos, es que el tipo de cambio se consistente, es decir que se utilice un solo tipo de moneda para toda la base de datos.
  + **Contenido:** Este es un atributo compuesto, mediante este se puede especificar la cantidad neta de producto según la descripción de su presentación mediante un número y una unidad de medida, por ejemplo: “120g” o “310ml”.
  + **Categoría ABC:** Indica el tipo de producto para la tienda según la clasificación ABC.
  + **Descontinuado:** Este atributo indica si un producto en específico ya está descontinuado por su fabricante o si ya no se vende más en la tienda.
  + **Clase:** Este indica la clasificación general de un producto, por ejemplo, muchas las numerosas marcas y productos como Cheetos, Doritos, etc., pueden ser clasificados bajo la categoría “Frituras”.
  + **Descripción:** Este atributo está diseñado para colocar una descripción textual oficial o no del producto.
  + **Extensibilidad (Json):** Mediante este atributo se pretende que el usuario del sistema pueda agregar de manera opcional cualquier característica del producto que este considere necesaria o muy específica tal que no esté dentro del esquema relacional propuesto. Este atributo permite agregar atributos extras a un producto mediante el almacenamiento de datos no estructurados con Json.
  + **Nombre:** Este campo sirve para que se coloque el nombre de la presentación del producto, por ejemplo: “Leche Alpura deslactosada 1 litro”.
  + **Id (SKU):** Este identificador tiene la característica de estar formado por varias características del producto, por ejemplo: "COLA-500ML-001".
  + **Precio promedio:** Es el precio promedio del producto según una base de datos externa.
  + **Relación identificada con entidad débil (tiene proveedor):** Los productos están asociados con su proveedor correspondiente, la entidad proveedor tiene información relevante como nombre, su página web, posibles números telefónicos y dirección física.
* **Entidad usuario:** Los usuarios son los principales agentes de la base de datos, estos hacen las transacciones importantes como ventas y recibos de stock.
  + **Nombre:** Este atributo indica el nombre completo de un usuario mediante sus nombres y apellidos.
  + **Claves:** Mediante este campo compuesto se guarda un número aleatorio que debe ser concatenado con la contraseña del usuario para posteriormente obtener el hash de dicha concatenación, esto para no guardar la contraseña del usuario en texto claro.
  + **Email:** Esta información sirve para ciertas validaciones como la recuperación de contraseña y autentificación de usuarios para la creación de sesiones.
  + **Relación identificada con entidad débil (tiene recuperación de contraseña):** Los usuarios pueden pedir un cambio de contraseña, la entidad débil de cambio de contraseña cuenta con una etiqueta de tiempo de expiración y un numero de único uso secreto el cual es compartido con el usuario mediante su correo electrónico.
  + **Relación identificada con entidad débil (tiene sesión):** Mediante la relación con la entidad débil (sesión), se registra la hora a la que inicia y termina una sesión de usuario mediante la cual puede hacer transacciones. Además, las sesiones tienen también un número aleatorio que se verifica en cada petición y se actualiza en cada transacción para evitar ataques de repetición.
* **Entidad débil existencia:** Las existencias de un producto dependen de otras al menos dos entidades propietarias esto, mediante tres relaciones identificadas que son mutuamente exclusivas, y una relación identificada obligatoria en cada uno de los siguientes casos; Esta entidad existe cuando un usuario hace una venta, un gerente recibe stock, o un gerente registra una merma; por extensión, es necesario que esta entidad referencie al menos una entidad producto. Mediante la entidad existencia, se registra el precio unitario del producto, la cantidad de productos (de entrada, o de salida), el precio total según la cantidad de productos, y el lote al que pertenecen estos productos.
  + **Relación identificada con gerente (Recibe):** Mediante esta relación, se registran los recibos de stock, como atributos, se guarda una etiqueta de tiempo de la hora agendada del recibo, y de la hora real a la que llegó.
  + **Relación identificada con gerente (Merma):** Esta relación registra cuando un producto se registra como perdida debido a que ha caducado.
  + **Relación identificada con usuario (Venta):** Este nos permite rastrear el precio total de la transacción, y la hora a la que se realizó la venta.
  + **Relación identificada con producto (Tiene):** Esta relación identificada es obligatoria siempre que exista cualquiera de las 3 relaciones anteriores.
* **Entidad oferta:** Esta entidad tiene la peculiaridad de relacionarse con una instancia de existencia cuando una oferta es aplicable a la existencia mediante una venta realizada por un usuario. Las ofertas en la base de datos ofrecen la forma más general de hacer descuentos, es decir, por porcentaje aplicados a cada producto que conforma la oferta mediante la relación con producto “incluye”.
  + **Etiquetas de tiempo, inicio y fin:** Estos atributos indican el periodo de tiempo dentro del que la oferta es válida.
  + **Máximo por cliente:** Este atributo indica la cantidad de ofertas máximas de esta clase que puede comprar un cliente.
  + **Extensibilidad (Json):** Dado que las ofertas representan un punto de relativa complejidad en cualquier tienda de conveniencia (por los distintos tipos de ofertas que se pueden realizar, como por clase de productos, por días únicos de la semana, incluso para clientes específicos), se consideró viable agregar un atributo que permita una mayor personalización de ofertas mediante datos no estructurados, de tal manera, que a futuro se puedan implementar otros tipos de ofertas además de descuentos por porcentaje como es el caso actual.

A continuación, se muestra el diagrama entidad relación generado. Cabe mencionar que es probable que este modelo sufra pequeñas adiciones futuras, donde si bien, no se reestructurará por completo, es posible que se agreguen nuevas entidades o atributos a entidades ya existentes

****

# **CAPÍTULO 5 DISEÑO DEL SISTEMA**

El contenido de esta sección del documento aborda la estructura lógica, composición y diseño visual de Kv-wims. Se detallan diversos diagramas con el propósito de proporcionar una comprensión concisa de la interacción de los usuarios y el sistema, los distintos niveles de funcionalidad, así como una visión provisional del diseño final. Este último aspecto se logra mediante la presentación de “wireframes” que representan de manera anticipada las diversas pantallas que conformarán la propuesta de solución al ser desplegada en línea.

## **5.1 Tarjetas CRC**

|  |  |
| --- | --- |
| **Existencia** | |
| Lote de un producto  Cantidad de un lote  Precio total de las existencias  Fecha de caducidad | Usuario |

## **5.2 Diagrama de clases**

Se está trabajando

## **5.3 Diagramas UML**

Se están trabajando

## **5.4 Diagrama relacional de la base de datos**

Se está trabajando

## **5.5 Arquitectura lógica del sistema**

Se está trabajando

## **5.6 “Wireframes”**

Como parte del desarrollo correspondiente al diseño del sistema, es importante tener claro la forma gráfica en que podremos visualizar Kv-wims y es por ese motivo que a lo largo de esta sección podremos visualizar las pantallas preliminares del sistema mediante el uso de wireframes que son la representación de la estructura tentativa de la aplicación web mostrando de forma general el contenido y las funcionalidades que estarán presentes en el sistema. A continuación, se muestran un total de 16 pantallas que describen lo antes mencionado.

Interfaz de usuario gráfica, Aplicación, Sitio web

Descripción generada automáticamenteInterfaz de usuario gráfica

Descripción generada automáticamente con confianza media

**Interfaz de usuario gráfica

Descripción generada automáticamente con confianza media**

**Interfaz de usuario gráfica

Descripción generada automáticamente con confianza baja**

**Interfaz de usuario gráfica, Tabla

Descripción generada automáticamente**

**Tabla

Descripción generada automáticamente**

**Interfaz de usuario gráfica

Descripción generada automáticamente**

**Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente**

**Interfaz de usuario gráfica, Tabla

Descripción generada automáticamente**

**Interfaz de usuario gráfica

Descripción generada automáticamente**

**Tabla

Descripción generada automáticamente**

**Interfaz de usuario gráfica

Descripción generada automáticamente**

**Tabla

Descripción generada automáticamente**

# **Anexos**

Fotos de los anaqueles de un Oxxo, como se ven sus repisas y refrigeradores

# **BIBLIOGRAFÍA**

[1] M. Ten Hompel y T. Schmidt, *Warehouse management: automation and organisation of warehouse and order picking systems*. Berlin ; New York: Springer, 2007.

[2] Y. Duran, «Administración del inventario: elemento clave para la optimización de las utilidades en las empresas», vol. 1, pp. 55-78, 2012.

[3] N. D. S. Panzuto y P. C. C. Rodrigues, «ANALYSIS OF INVENTORY IN A SMALL BUSINESS», *Indep. J. Manag. Prod.*, vol. 1, n.o 1, pp. 37-50, ago. 2010, doi: 10.14807/ijmp.v1i1.30

[4] A. Jenkins, «Diferencia entre IMS y WMS», *Inventory Management vs. Warehouse Management: What’s the Difference?*, 5 de septiembre de 2023. Disponible en: https://www.netsuite.com/portal/resource/articles/inventory-management/inventory-management-warehouse-management.shtml. [Accedido: 17 de octubre de 2023]

[5] A. Körber, «Warehouse Management Systems», *Warehouse Management Systems*, 18 de octubre de 2023. Disponible en: https://www.koerber-supplychain-software.com/en/supply-chain-solutions/warehouse-management-systems. [Accedido: 18 de octubre de 2023]

[6] Lightspeed, «Vend by Lightspeed». 2023. Disponible en: https://www.lightspeedhq.com/vend/

[7] S. Monhattan, «Manhattan Active Warehouse Management», *Warehouse Management*, 18 de octubre de 2023. Disponible en: https://www.manh.com/es-xl/soluciones/software-de-gestion-de-la-cadena-de-suministro/sistema-de-gestion-de-almacenes. [Accedido: 18 de octubre de 2023]

[8] I. Sommerville, *Software engineering*, 10. ed. Boston, Munich: Pearson, 2016.

[9] Programación Fácil, «Los modelos MVC y MVT con Django». 18 de abril de 2023. Disponible en: https://programacionfacil.org/blog/los-modelos-mvc-y-mvt-con-django/

[10] G. Reese, *Database programming with JDBC and Java*, 2nd ed. en The Java series. Cambridge, Mass: O’Reilly, 2000. Disponible en: https://web.archive.org/web/20110406121920/http://java.sun.com/developer/Books/jdbc/ch07.pdf. [Accedido: 23 de octubre de 2023]

[11] A. Shaw, *CPython internals: your guide to the Python 3 interpreter*, First edition. United States? Real Python, 2021.

[12] Python Software Foundation, «What is Python? Executive Summary», *Python Essays*. Disponible en: https://www.python.org/doc/essays/blurb/. [Accedido: 2 de noviembre de 2023]

[13] Python Software Foundation, «Python Software Foundation: Modules». Python Software Foundation, 2023. Disponible en: https://docs.python.org/3/tutorial/modules.html. [Accedido: 2 de noviembre de 2023]

[14] J. Sturtz, «Python Modules and Packages – An Introduction». Real Python. Disponible en: https://realpython.com/python-modules-packages/#python-packages. [Accedido: 2 de noviembre de 2023]

[15] Python Software Foundation, «Python Packaging User Guide: Installing Packages». Python Software Foundation. Disponible en: https://packaging.python.org/en/latest/tutorials/installing-packages/. [Accedido: 2 de noviembre de 2023]

[16] MDN contributors, «Django introduction». Mozilla Foundation, 4 de octubre de 2023. Disponible en: https://developer.mozilla.org/en-US/docs/Learn/Server-side/Django/Introduction. [Accedido: 3 de noviembre de 2023]

[17] Django Software Foundation, «Documentation: Django at a glance». 2023. Disponible en: https://docs.djangoproject.com/en/4.2/intro/overview/. [Accedido: 3 de noviembre de 2023]

[18] J. Lehtosalo, «mypy Documentation: Welcome to mypy documentation!» Read the Docs, 2022. Disponible en: https://mypy.readthedocs.io/en/stable/index.html. [Accedido: 3 de noviembre de 2023]

[19] C. Winter y T. Lownds, «PEP 3107 – Function Annotations». Python Software Foundation, 2 de diciembre de 2006. Disponible en: https://peps.python.org/pep-3107/. [Accedido: 3 de noviembre de 2023]

[20] G. van Rossum y I. Levkivskyi, «PEP 483 – The Theory of Type Hints». Python Software Foundation, 19 de diciembre de 2014. Disponible en: https://peps.python.org/pep-0483/. [Accedido: 3 de noviembre de 2023]

[21] E. Gunn, «16 of the Best Python Machine Learning Libraries to Try». CareerFoundry, 17 de octubre de 2023. Disponible en: https://careerfoundry.com/en/blog/data-analytics/python-machine-learning-libraries/

[22] J. D. Gauchat, *El gran libro de HTML5, CSS3 y JavaScript*, Primera edición. Barcelona: Marcombo, S.A., 2012.

[23] Oracle Coorporation, «MySQL Documentation Library: What is MySQL?» Oracle Coorporation, 27 de octubre de 2023. Disponible en: https://docs.oracle.com/cd/E17952\_01/mysql-8.0-en/what-is-mysql.html

[24] José. Hernández Orallo, C. Ferri Ramírez, y M. José. Ramírez Quintana, *Introducción a la minería de datos*, Reimp. Madrid: Pearson, 2010.

[25] Teachers College, Columbia University, «Research Instrument Examples». Columbia University. Disponible en: https://www.tc.columbia.edu/media/administration/institutional-review-board-/guide-amp-resources---documents/Published\_Study-Material-Examples.pdf

[26] T. Lauren, «Cross-Sectional Study | Definition, Uses & Examples». Scribbr, 8 de mayo de 2020. Disponible en: https://www.scribbr.com/methodology/cross-sectional-study/#:~:text=A%20cross-sectional%20study%20is,observe%20variables%20without%20influencing%20them.

[27] B. Pritha, «Naturalistic Observation | Definition, Guide, & Examples». Scribbr, 10 de febrero de 2022. Disponible en: https://www.scribbr.com/methodology/naturalistic-observation/#:~:text=a%20naturalistic%20observation%3F-,Naturalistic%20observation%20is%20a%20qualitative%20research%20method%20where%20you%20record,people%20watching”%20with%20a%20purpose.

[28] INEGI, «Consulta de Precios Promedio». septiembre de 2023. Disponible en: https://www.inegi.org.mx/app/preciospromedio/?bs=18

[29] A. Ali Awan, «A Complete Guide to Data Augmentation». noviembre de 2022. Disponible en: https://www.datacamp.com/tutorial/complete-guide-data-augmentation

[30] L. Schmidt, «Tabular Data Augmentation with Deep Learning». 10 de mayo de 2021. Disponible en: https://medium.com/analytics-vidhya/tabular-data-augmentation-with-deep-learning-7270584e7ff

[31] Google for Developers, «Machine learning: Imbalanced Data». 9 de junio de 2023. Disponible en: https://developers.google.com/machine-learning/data-prep/construct/sampling-splitting/imbalanced-data

[32] S. Onishi y S. Megro, «Rethinking Data Augmentation for Tabular Data in Deep Learning». 22 de mayo de 2023. Disponible en: https://arxiv.org/pdf/2305.10308.pdf#:~:text=Data%20augmentation%20has%20been%20widely,and%20other%20types%20of%20data.

[33] D. Haba, *DATA AUGMENTATION WITH PYTHON enhance deep learning accuracy with data augmentation methods for image, text, audio, and tabular data*, 1st edition. England: PACKT PUBLISHING LIMITED, 2023.

[34] L. Xu y K. Veeramachaneni, «Synthesizing Tabular Data using Generative Adversarial Networks». 27 de noviembre de 2018. Disponible en: https://arxiv.org/pdf/1811.11264.pdf

[35] F. López, «SMOTE: Synthetic Data Augmentation for Tabular Data». 28 de febrero de 2021. Disponible en: https://towardsdatascience.com/smote-synthetic-data-augmentation-for-tabular-data-1ce28090debc

[36] TABLEAU SOFTWARE, «Time Series Forecasting: Definition, Applications, and Examples». 2023. Disponible en: https://www.tableau.com/learn/articles/time-series-forecasting#:~:text=What%20is%20time%20series%20forecasting,and%20inform%20strategic%20decision-making.

[37] P. Pathak, «Time Series Forecasting — A Complete Guide». 8 de septiembre de 2021. Disponible en: https://medium.com/analytics-vidhya/time-series-forecasting-a-complete-guide-d963142da33f

[38] S. Hayasaka, «What is Clustering and How Does it Work?» 21 de junio de 2021. Disponible en: What is Clustering and How Does it Work?

[39] S. Kaushik, «Clustering | Introduction, Different Methods, and Applications (Updated 2023)». 11 de julio de 2023. Disponible en: https://www.analyticsvidhya.com/blog/2016/11/an-introduction-to-clustering-and-different-methods-of-clustering/#What\_Is\_Clustering?

[40] Instituto Nacional de Estadística y Geografía, «Las tiendas de conveniencia o minisúper en México». Disponible en: https://www.inegi.org.mx/contenidos/programas/ce/2019/doc/m\_ticonv\_ce19.pdf

[41] M. V. López Cisneros, «Propuesta de plan de gestión para el área de recepción de mercancías en la empresa supermercados Caribe de Jamundí», Fundación Universitaria Católica Lumen Gentium. Disponible en: http://hdl.handle.net/20.500.12237/494

[42] F. Hemeryth Charpentier y J. M. Sánchez Gutiérrez, «Implementación de un sistema de control interno operativo en los almacenes, para mejorar la gestión de inventarios de la constructora A&A S.A.C. de la ciudad de Trujillo - 2013», Universidad Privada Antenor Orrego, 2013. Disponible en: https://hdl.handle.net/20.500.12759/140

[43] P. S. Mauricio Timana y S. L. Mora Santur, «5S como técnica para orientar la gestión del almacén de la UnidadEjecutora 003 Región Policial Piura, 2021.», Universidad César Vallejo, 2021. Disponible en: https://hdl.handle.net/20.500.12692/80161

[44] J. Tompkins, J. A. White, Y. A. Bozer, y J. A. Tompkins, *Facilities planning*, 4th edition. Hoboken, NJ: Wiley, 2010.

[45] J. Abby, «What is inventory management? Benefits, types, & techniques», 2020. Disponible en: https://www.netsuite.com/portal/resource/articles/inventory-management/inventory-management.shtml

[46] A. M. Atieh *et al.*, «Performance Improvement of Inventory Management System Processes by an Automated Warehouse Management System», *Procedia CIRP*, vol. 41, pp. 568-572, 2016, doi: 10.1016/j.procir.2015.12.122

[47] J. M. Juran, *Juran y el liderazgo para la calidad: manual para ejecutivos*. Ediciones Díaz de Santos, 2000.

[48] S. J. Calderón Cortés, «Los agotados (stock outs) y su impacto en las ventas e inventarios en la categoría de detergentes líquidos en una cadena de autoservicio de América Latina», Universidad Nacional Autónoma de México, 2015. Disponible en: https://hdl.handle.net/20.500.14330/TES01000739224

[49] Lifeder, «Punto de reorden.», 2021. Disponible en: https://www.lifeder.com/punto-reorden/

[50] G. G. Suárez Gallegos y P. D. Cárdenas Miranda, «La rotación de los inventarios y su incidencia en el flujo de efectivo», *Revista Observatorio de la Economía Latinoamericana*, 2017. Disponible en: http://www.eumed.net/cursecon/ecolat/ec/2017/inventarios-flujo-efectivo.html http://hdl.handle.net/20.500.11763/ec17inventarios-flujo-efectivo

[51] G. R. Henríquez-Fuentes, Diego A. Cardona, J. A. Rada-Llanos, y N. R. Robles, «Medición de Tiempos en un Sistema de Distribución bajo un Estudio de Métodos y Tiempos», *Inf. Tecnológica*, vol. 29, pp. 277-286, dic. 2018.

[52] E. T. Peterson, *The big book of key performance indicators*. Web analytics demystified, 2006.

[53] O. Pak, M. Ferguson, O. Perdikaki, y S.-M. Wu, «Optimizing SKU Selection for Promotional Display Space at Grocery Retailers». noviembre de 2019. doi: 10.2139/ssrn.3506709. Disponible en: https://ssrn.com/abstract=3506709

[54] A. Tanwari, A. Q. Lakhiar, y G. Y. Shaikh, «ABC analysis as a inventory control technique», *Quaid-E-Awam Univ. Res. J. Eng. Sci. Technol.*, vol. 1, n.o 1, 2000.

[55] Okta, Inc, «What is Authentication?» 2023. Disponible en: https://auth0.com/intro-to-iam/what-is-authentication

[56] National Institute of Standards and Technology, «Digital Signature Standard (DSS)». 3 de febrero de 2023. Disponible en: https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.186-5.pdf

[57] National Institute of Standards and Technology, «Committee on National Security Systems (CNSS) Glossary». abr de 2015. Disponible en: https://rmf.org/wp-content/uploads/2017/10/CNSSI-4009.pdf

[58] E. Barker, «Recommendation for Digital Signature Timeliness». septiembre de 2009. Disponible en: https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-102.pdf

[59] R. Shirey, «Internet Security Glossary, Version 2». agosto de 2007. Disponible en: https://datatracker.ietf.org/doc/html/rfc4949

[60] *Ingeniería del software: un enfoque práctico*, Séptima edición. México; Bogotá; Buenos Aires; Caracas; Guatemala; Madrid: McGraw-Hill Education, 2013.

[61] MongoDB, «What are ACID Properties in Database Management Systems?» MongoDB, 2023. Disponible en: https://www.mongodb.com/basics/acid-transactions#:~:text=ACID%20is%20an%20acronym%20that,the%20event%20of%20unexpected%20errors.

[62] Oxxo, «ESTAMOS DE FIESTA Y NOS GUSTARÍA COMPARTIRTE PARTE DE LO QUE SOMOS. ALGUNOS DATOS SON CURIOSOS, OTROS DIVERTIDOS, ÚTILES O QUE NOS LLENAN DE ORGULLO.» 2023. Disponible en: https://www.oxxo.com/conocenos#:~:text=Diariamente%20se%20atienden%20a%20más,consumo%20diario%20para%20el%20hogar.

[63] J. Ruiz Torre, «¿Cuánto vende un Oxxo al día?» El Financiero, 9 de noviembre de 2022. Disponible en: https://www.elfinanciero.com.mx/opinion/jonathan-ruiz/2022/11/09/cuanto-vende-un-oxxo-al-dia/

[64] Microsoft Coorporation, «Definir y resolver un problema con Solver». Disponible en: https://support.microsoft.com/es-es/office/definir-y-resolver-un-problema-con-solver-5d1a388f-079d-43ac-a7eb-f63e45925040

[65] Oracle Coorporation, «11.7 Data Type Storage Requirements». 2023. Disponible en: https://dev.mysql.com/doc/refman/8.0/en/storage-requirements.html#data-types-storage-reqs-numeric

[66] Microsoft Coorporation, «Azure managed disk types». 15 de noviembre de 2023. Disponible en: https://learn.microsoft.com/en-us/azure/virtual-machines/disks-types

[67] VMware, «What is Cloud Scalability?» 2021. Disponible en: https://www.vmware.com/topics/glossary/content/cloud-scalability.html

1. gerardo.garzon.dominguez@gmail.com [↑](#footnote-ref-2)
2. munoz.ramirez.gabriel.ipn@gmail.com [↑](#footnote-ref-3)
3. sanchezbecerraernestodaniel@gmail.com [↑](#footnote-ref-4)