

**Prototipo de software “AGUSTINV” como herramienta didáctica para la gestión de inventarios en
Universitaria Agustiniana**

Valentina Peña Carvajal

Universitaria Agustiniana
Facultad de Ingenierías
Programa Tecnología en Desarrollo de Software
Bogotá. D.C.
2021

**Prototipo de software “AGUSTINV” como herramienta didáctica para la gestión de inventarios en
Universitaria Agustiniana**

Valentina Peña Carvajal

Director

Mauricio Alonso Villalba

Trabajo de grado para optar al título de Tecnólogo en Desarrollo de Software

Universitaria Agustiniana

Facultad de Ingenierías

Programa Tecnología en Desarrollo de Software

Bogotá. D.C.

2021

Agradecimientos

Dios,

por permitirme avanzar y vencer cada obstáculo que he afrontado

a lo largo de la vida.

Mi familia por apoyarme y enseñarme que en la vida,

si uno hace las cosas a consciencia todo es posible.

A la profesora Marta Segura Ruiz y el profesor Mauricio Alonso Villalba,

por apoyarme en todo el paso de la carrera,

sus experiencias y conocimientos fueron claves

para lograr este objetivo de ser un profesional.

Resumen

En el presente documento se lleva a cabo el desarrollo de una herramienta didáctica para la gestión de inventarios en la Universitaria Agustiniiana. El nombre del sistema es AgustInv, aplicativo web que tiene como objetivo principal maximizar la eficacia y eficiencia de los procesos contables, administrativos y comerciales en los cursos de producción. La metodología que se va a implementar para el desarrollo de proyecto es la RUP (Proceso Unificado de Racional). El sistema se vinculará a una base de datos desarrollada en PHP como lenguaje de desarrollo y MYSQL como gestor de base de datos, en el que se almacenará toda la información de todos los procesos contables. Este proyecto será dirigido para los estudiantes de las facultades ciencias administrativas, con el propósito de brindarles una enseñanza educativa por medio de una plataforma web que les permita realizar procesos en tiempo real, establecido en el ámbito laboral.

Palabras claves: Inventarios, Software, Herramienta, Vinculo, Procesos.

Abstract

This document carries out the development of a didactic tool for inventory management in the Universitaria Agustiniana. The name of the system is AgustInv, a web application that has as its main objective to maximize the efficacy and efficiency of accounting, administrative and commercial processes in production courses. The methodology to be implemented for project development is the RUP (Unified Rational Process). The system will be linked to a database developed in PHP as a development language and MYSQL as a database manager, in which all information from all accounting processes will be stored. This project will be aimed at students of the administrative science faculties, with the purpose of providing them with an educational teaching through a web platform that allows them to carry out processes in real time, established in the field of work.

Keywords: Inventories, Software, Tool, Link, Processes.

Tabla de contenidos

Introducción	13
1. Título del proyecto	14
2. Planeación del proyecto	15
2.1. Objetivos del proyecto	15
2.1.1. Objetivo general	15
2.1.2. Objetivos específicos	15
2.2. Planteamiento del problema	15
2.2.1. Prediagnóstico	15
2.3. Alcance del proyecto	16
2.4. Metodología RUP	16
2.4.1. Desarrollo de software iterativo	16
2.4.2. Administración de requisitos	17
2.4.3. Generalidades del uso de arquitecturas que están basadas en componentes	17
2.4.4. Software de modelado visual	18
2.4.5. Verificación de la calidad del software	18
2.4.6. Control en los cambios para el software	18
2.4.7. Fases e interacciones	18
2.4.8. Fase de Inicio	19
2.4.9. Fase de Elaboración	19
2.4.10. Fase de Construcción	19
2.4.11. Fase de Transición	20
2.4.12. Iteraciones	20
2.4.13. Estructura estática del proceso	20
3. Marco Teórico y estado del arte	22
3.1. Marco Teórico	22
3.1.1. Inventarios	22
3.1.2. Sistema de inventario periódico	22
3.1.3. Sistema de inventario permanente	22
3.1.4. Métodos para la valoración de inventarios	23
3.1.5. Kárdex o auxiliar de la cuenta	23
3.1.6. Características de un sistema de gestión de inventario	25

3.2. Estado del arte	25
3.2.1. Actualidad del desarrollo de software.....	25
3.2.2. Legislación de inventarios.	26
3.2.3. Software de inventarios.....	27
4. Especificación de requisitos de software	35
4.1. Perspectiva del producto.....	35
4.2. Funcionalidad del producto	35
4.3. Características del usuario	37
4.4. Restricciones.....	37
4.5. Suposiciones y dependencias.....	37
4.6. Requisitos Específicos.....	38
4.6.1. Actores.	38
4.6.2. Requisitos funcionales.	39
4.6.3. Diagrama de casos de uso.	39
4.6.4. Especificaciones de los casos de uso.	40
4.7. Requisitos de rendimiento	50
4.7.1. Rendimiento.....	50
4.7.2. Seguridad.	50
4.7.3. Mantenibilidad.	50
4.7.4. Usabilidad.	50
4.7.5. Disponibilidad.....	50
4.8. Restricciones de diseño	50
4.9. Atributos de calidad del software del sistema	51
5. Diseño del software (ISO -12207-1).....	52
5.1. Diseño de la arquitectura del software	52
5.2. Diseño detallado del software.....	52
5.2.1. Diagrama de clases.	52
5.2.2. Diagrama de paquetes.	53
5.2.3. Diagrama de despliegue.....	54
5.3. Diseño de interfaz.....	55
5.3.1. Interfaz gráfica de usuario.	55
5.3.2. Interfaces de entrada.	55

5.3.3. Interfaces de salida.....	56
6. Implementación.....	58
6.1. Plataformas de desarrollo	58
6.2. Base de datos	58
6.2.1. Diseño de la base de datos.	58
6.2.2. Modelo entidad relación.	58
6.2.3. Infraestructura de hardware y redes.	59
7. Pruebas del software	60
7.1. Pruebas del software	60
7.2. Pruebas de usabilidad`	60
Conclusiones	64
Referencias	65

Lista de figuras

Figura 1. Metodología RUP. Adaptado de Metodología RUP.....	17
Figura 2. Casos de uso metodología RUP. Adaptado de Casos de uso RUP.....	17
Figura 3. Administración de Requerimientos. Adaptado de Administración de requerimientos RUP.....	18
Figura 4. Disciplinas RUP. Adaptado de Metodología RUP	19
Figura 5. Fases del método RUP. Adaptado de Fases RUP, por L.	20
Figura 6. Estructura metodología RUP. Adaptado de Estructura estática RUP.....	20
Figura 7. Estructura del Kardex. Adaptado de ejemplo de Kardex,.....	23
Figura 8. Logo ofima S.A.S. Adaptado de Ofima SAS	27
Figura 9. Logo Orión Plus. Adaptado de Orión Plus	28
Figura 10. Logo Alegra. Adaptado de Alegra Colombia	29
Figura 11. Logo Siigo. Adaptado de SIIGO.....	30
Figura 12. Logo POS. Adaptado de POS Colombia	30
Figura 13. Logo Loggro. Adaptado de Loggro	31
Figura 14. Logo Logimov. Adaptado de Logimov	31
Figura 15. Logo Softland go. Adaptado de Softland HCM	32
Figura 16. Logo TNS. Adaptado de Software TNS	32
Figura 17. Logo FCS. Adaptado de FCS	33
Figura 18. Logo Consensus. Adaptado de Consensus	33
Figura 19. Logo Helisa. Adaptado de Helisa	34
Figura 20. Logo Xubio. Adaptado de Xubio.....	34
Figura 21. Diagrama general casos de uso.....	40
Figura 22. Modelo Vista – Controlador	52
Figura 23. Modelo entidad – Relación.....	52
Figura 24. Diagrama de paquetes.....	53
Figura 25. Diagrama de interfaces	53
Figura 26. Diagrama de despliegue.....	54
Figura 27. Diagrama de despliegue BD	54
Figura 28. Interfaz Gráfica de usuario	55
Figura 29. Interfaz de login.....	55

Figura 30. Registro de usuarios.....	55
Figura 31. Actualización de usuario.....	56
Figura 32. Listado de usuarios.	56
Figura 33. Eliminación de usuarios.....	57
Figura 34. Modelo entidad-relación.....	58
Figura 35. Pruebas de software.....	60
Figura 36. Pruebas de usabilidad.....	60
Figura 37. Pregunta de claridad.....	61
Figura 38. Tipo de audiencia.....	61
Figura 39. Calidad del sitio.....	62
Figura 40. Objetivo del sitio.....	62
Figura 41. Buscadores.....	63

Lista de tablas

Tabla 1. Actor Administrador	38
Tabla 2. Actor Usuario estudiante.....	38
Tabla 3. Caso de Uso Autenticación	40
Tabla 4. Caso de uso creación de productos.	41
Tabla 5. Caso de Uso modificación de productos.	41
Tabla 6. Caso de Uso eliminación de productos.	41
Tabla 7. Caso de uso búsqueda de productos.....	42
Tabla 8. Caso de uso creación de clientes	42
Tabla 9. Caso de Uso modificación de clientes.	42
Tabla 10. Caso de Uso eliminación de clientes.....	43
Tabla 11. Caso de uso búsqueda de clientes.	43
Tabla 12. Caso de uso creación de operaciones.....	43
Tabla 13. Caso de Uso modificación de operaciones.....	44
Tabla 14. Caso de Uso eliminación de operaciones.....	44
Tabla 15. Caso de uso búsqueda de operaciones.	44
Tabla 16. Caso de uso creación de facturas.	45
Tabla 17. Caso de Uso modificación de facturas.....	45
Tabla 18. Caso de Uso eliminación de facturas.	45
Tabla 19. Caso de uso búsqueda de facturas.	46
Tabla 20. Caso de uso creación de proveedores.....	46
Tabla 21. Caso de Uso modificación de proveedores.	46
Tabla 22. Caso de Uso eliminación de proveedores.	47
Tabla 23. Caso de uso búsqueda de proveedores.....	47
Tabla 24. Caso de uso creación de usuarios.....	47
Tabla 25. Caso de Uso modificación de usuarios.	48
Tabla 26. Caso de Uso eliminación de usuarios.....	48
Tabla 27. Caso de uso búsqueda de usuarios.	48
Tabla 28 Caso de Uso alerta stock mínimo.....	49

Tabla 29. Caso de Uso stock máximo. 49

Tabla 30. Caso de uso generación de reportes. 49

Tabla 31. Caso de Uso inventario total. 50

Introducción

Actualmente los procesos de los cursos Uniagustinianos se realizaban de manera manual, comenzando por intermedio de tarjetas Kardex completamente ordenadas para cada artículo, este sistema dio muy buenos resultados y fue una decisión acertada en su momento, pues en épocas pasadas únicamente se llevaba el inventario a mano, con el paso del tiempo el inventario paso a ser completamente diligenciado en hojas de cálculo del programa de Office, Microsoft Excel; pero con el crecimiento de la economía y la competencia, la cantidad de productos que se ofrecen y sus referencias han aumentado drásticamente. Actualmente se calcula que los inventarios constan de más de 200 referencias lo que generan graves problemas para controlar el stock de todos los productos presentando casos que repercuten en las ventas de las empresas, además afecta procesos de la organización.

Ahora bien, en lo respectivo con el inventario de las organizaciones que manejan muchas referencias, es esencial saber lo que se tiene en el almacén y sobre todas las cosas donde se encuentran los productos, encontrar un sistema de gestión de inventarios correcto, logra encaminar acciones inmediatas orientadas a aumentar los resultados en la toma de decisiones, la superación de obstáculos en otras áreas que afectan el rendimiento, la satisfacción de los empleados y el ahorro de tiempo para los clientes.

1. Título del proyecto

Prototipo de software “AgustInv” como herramienta didáctica para la gestión de inventarios en Universitaria Agustiniana.

2. Planeación del proyecto

2.1. Objetivos del proyecto

2.1.1. Objetivo general.

Desarrollar un prototipo de Software para la gestión de los inventarios “AgustInv” que maximice la eficacia y eficiencia de los procesos contables, administrativos y comerciales en los cursos de producción.

2.1.2. Objetivos específicos

Identificar el proceso de construcción de software con relación al curso de producción, y las restricciones de costos, tiempo y tecnología.

Establecer los requisitos para el análisis del sistema de inventarios en los cursos de producción.

Evaluar las diferentes herramientas de modelado para el seguimiento del proceso tecnológico del sistema de inventarios.

Realizar la implementación del software o codificación y la integración del sistema.

2.2. Planteamiento del problema

2.2.1. Prediagnóstico.

En la actualidad los cursos de producción en Uniagustiniana manejan su inventario con archivos de Excel, en donde cada archivo representa una categoría de los productos y cada hoja de este archivo es un producto específico, así las cosas, cada vez que un artículo tiene movimiento el estudiante encargado debe ingresar al archivo buscar el producto y digitar si se produce una compra, una venta, una devolución entre otras opciones y digitar aparte cualquier novedad pendiente.

Todo este proceso dificulta que el inventario permanezca actualizado pues en ocasiones cuando se necesitan ingresar varias operaciones, el proceso se vuelve lento y se procede a escribir todo para después almacenarlo en las hojas de cálculo pues se pierde demasiado tiempo y se retrasan los pedidos lo que conlleva a una mala experiencia por parte del usuario.

Una de las peores consecuencias, es que por olvidos no intencionales no se apuntan el total de transacciones lo que genera que el inventario físico sería diferente al lógico y no se puedan tomar decisiones acertadas con respecto al stock de este.

Estas fallas en el área de inventario conllevan graves problemas en todas las demás de la empresa, pues, cuando no se conoce el inventario en tiempo real, ni su movimiento en un periodo de tiempo, no se pueden tomar decisiones gerenciales acertadas, ya que la información base es

incorrecta, así mismo, en ocasiones se venden referencias que aparecen en el inventario lógico pero cuando se verifica en bodega no se encuentra el producto, generando el malestar del cliente y ocasionando pérdidas de tiempo para los empleados.

Por esta situación para Uniagustiniana es de vital importancia este proyecto, pues ofrecería una herramienta didáctica para los estudiantes de diferentes programas, que se beneficiarían de información fiable y efectiva para la toma de decisiones en todas las áreas de una empresa, manteniendo correcto el nivel de inventario.

2.3. Alcance del proyecto

El proyecto se basa en un sistema de gestión de inventarios para los cursos de producción de los estudiantes de la facultad de ingeniería de Universitaria Agustiniiana el cual consiste en:

Un sistema web para simular los procesos de inventarios.

Debe tener un sistema de ingreso con usuario y contraseña que valide al docente y al estudiante.

El sistema web se conectará a una base de datos donde se almacenará la información de productos, factura, usuarios, proveedores y clientes.

El sistema de gestión de inventarios debe generar informes compra y venta de productos.

2.4. Metodología RUP

Proceso Unificado Racional

Este proceso de Ingeniería de Software proporciona un enfoque concienzudo para la asignación de trabajos y responsabilidades dentro de un desarrollo organizado. El objetivo de este proceso es asegurar la producción de software con los más altos estándares de alta calidad que cumpla las necesidades de los usuarios finales, dentro de unos tiempos y presupuestos predecibles. RUP promueve la productividad del trabajo en equipo proporcionando a cada miembro del equipo un fácil acceso a una base de conocimiento con una serie de directrices, plantillas y herramientas para actividades de desarrollo críticas.

RUP es soportado por herramientas que automatizan grandes partes del proceso. Son usadas para crear y mantener artefactos, modelos en particular, del proceso de Ingeniería de Software: modelado visual, programación, testing, etc. (López Roschiano & Pech Montejo, 2015), a continuación, se presentan sus principales características.

2.4.1. Desarrollo de software iterativo.

Por los sistemas complejos y sofisticados de la actualidad, ya no es posible trabajar de manera secuencial, para lograrlo es necesario buscar en ambiente iterativo, con lo cual se establezca un

entendimiento mayor de los problemas analizados a través de divisiones sucesivas y crecer a una solución efectiva a través de múltiples iteraciones. RUP se ocupa de los elementos de más alto riesgo en todas las etapas del ciclo de vida, lo que reduce significativamente el perfil de riesgo de un proyecto. (López Roschiano & Pech Montejó, 2015)

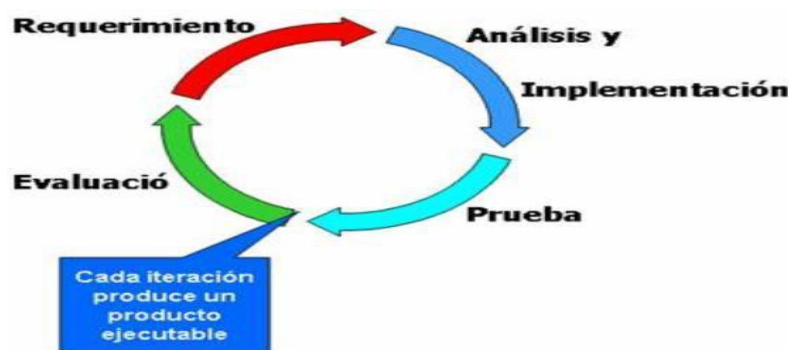


Figura 1. Metodología RUP. Adaptado de Metodología RUP, por Organización y métodos, (2011)

2.4.2. Administración de requisitos.

El método RUP explica claramente como se deben obtener, organizar y también documentar las funciones y limitaciones necesarias de un sistema; y fácilmente captura y comunica requerimientos del negocio. (López Roschiano & Pech Montejó, 2015)

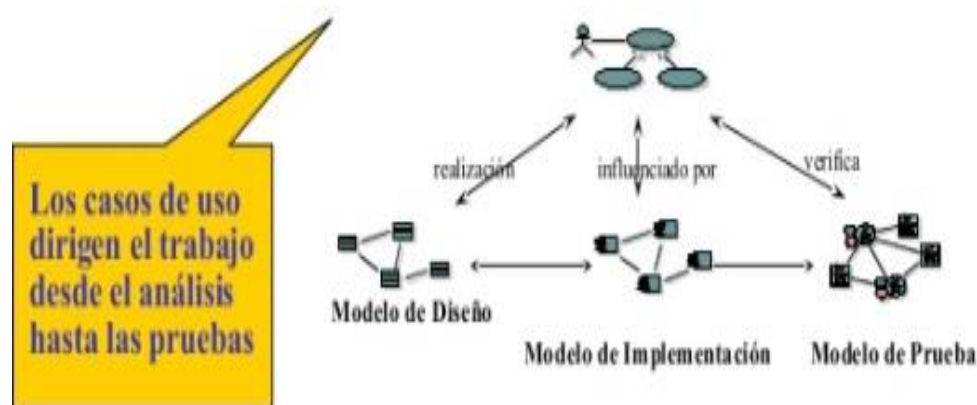


Figura 2. Casos de uso metodología RUP. Adaptado de Casos de uso RUP, por Google Sites, (2014)

2.4.3. Generalidades del uso de arquitecturas que están basadas en componentes.

Cuando las arquitecturas son utilizadas se presenta el desarrollo rápido de un potente ejecutable que no compromete los recursos para el desarrollo en mayor escala. En especial la arquitectura RUP se enfoca en diseñar un método flexible, el cual es propicio para entornos cambiantes, con

una forma de comprensión intuitiva y la ventaja de promover la reutilización de código como gran arma de presentación. (López Roschiano & Pech Montejó, 2015)

2.4.4. Software de modelado visual.

El proceso se enfoca en presentar la modelación del software de manera visual con el objetivo de capturar la estructura y el comportamiento de todos los componentes. Lo anterior, permite ocultar detalles y escribir código usando bloques gráficos de construcción. (López Roschiano & Pech Montejó, 2015)

2.4.5. Verificación de la calidad del software.

Dentro de este enfoque el bajo rendimiento de aplicaciones y la escasa fiabilidad son factores comunes que afectan drásticamente la aceptabilidad de las aplicaciones. Por lo tanto, la calidad debe ser el factor principal de revisiones periódicas que se encuentren centradas en la identificación de los requisitos mínimos de fiabilidad, funcionalidad, el rendimiento de las aplicaciones y también el rendimiento del sistema en general. (López Roschiano & Pech Montejó, 2015)



Figura 3. Administración de Requerimientos. Adaptado de Administración de requerimientos RUP, por Nery Becerra, (2002)

2.4.6. Control en los cambios para el software.

El desarrollo de software es un entorno especial para los cambios, y en este sentido RUP ofrece la capacidad de gestionar estas variantes, es por esto, que la arquitectura busca controlar, rastrear y monitorizar los cambios en el desarrollo con el fin de establecer un modelo iterativo eficiente. (López Roschiano & Pech Montejó, 2015)

2.4.7. Fases e interacciones.

Ésta es la organización dinámica del proceso en el tiempo. El ciclo de vida del software está compuesto por ciclos, cada ciclo de trabajo en una nueva generación del producto.

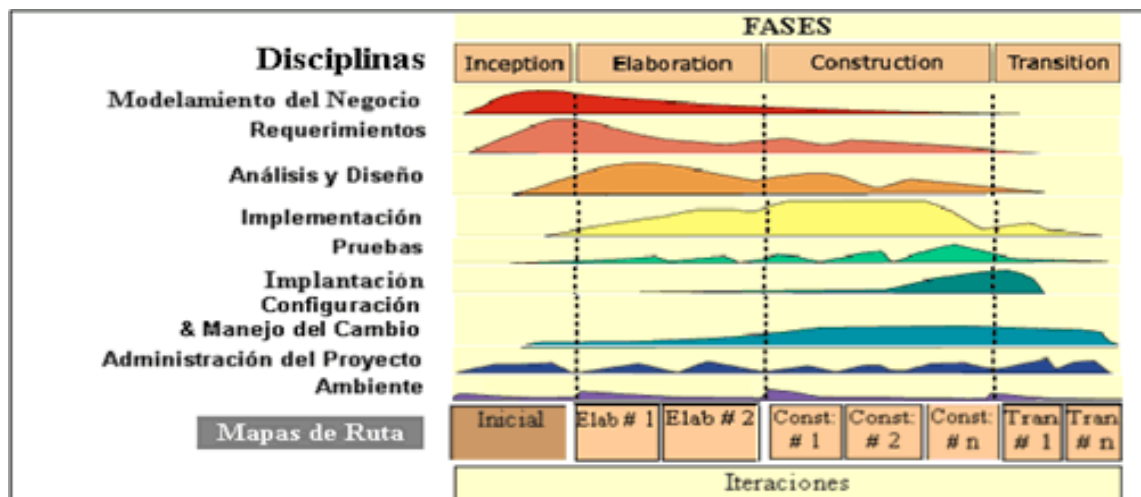


Figura 4. Disciplinas RUP. Adaptado de Metodología RUP, por Organización métodos, (2011)

2.4.8. Fase de inicio.

Para el desarrollo de la fase inicial es fundamental establecer el objetivo del negocio del sistema, pero además se debe delimitar el alcance del proyecto, con este fin se identifican todas las variables externas que interactúan con el desarrollo, los cuales se conocen como actores, pero además se definen las características de la interacción en forma general. (López Roschiano & Pech Montejo, 2015)

2.4.9. Fase de elaboración.

Esta fase tiene como el análisis del dominio, el establecimiento de la arquitectura, el desarrollo del plan de proyecto y la búsqueda de la eliminación de los elementos de riesgo del proyecto. Para cumplir estos objetivos se debe tener una visión global del sistema, la cual debe ser adquirida en la fase inmediatamente anterior.

En la fase de elaboración, un prototipo ejecutable de la arquitectura es construido en una o más iteraciones, dependiendo del alcance, tamaño, riesgo y grado de innovación del proyecto. (López Roschiano & Pech Montejo, 2015)

2.4.10. Fase de construcción.

En el transcurso de esta fase todos los componentes y sus correspondientes características pertenecientes a la aplicación son desarrolladas e integradas en el prototipo, y todos los componentes son probados exhaustivamente. Esta fase se define como un proceso que se centra en la gestión de los recursos y el control de operaciones que tiene como objetivo la optimización de costes, tiempo y calidad. (López Roschiano & Pech Montejo, 2015)

2.4.11. Fase de transición.

El propósito de la fase de transición es llevar el productor software a la comunidad de usuarios. Una vez el producto se haya entregado al usuario final, surgen problemas que requieren el desarrollo de una nueva versión, la corrección de los problemas o la finalización de las características que fueron pospuestas. Al final de esta fase se encuentra el cuarto hito, el hito del producto entregado. (López Roschiano & Pech Montejo, 2015)

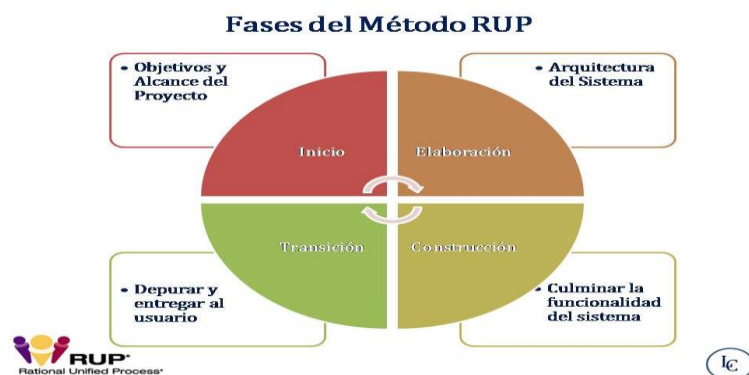


Figura 5. Fases del método RUP. Adaptado de Fases RUP, por L. Castellanos, (2016)

2.4.12. Iteraciones.

Cada fase de Rational Unified Process puede ser dividida en iteraciones. Una iteración es un ciclo de desarrollo completo que da como resultado una nueva versión, ya sea interna o externa, de un producto ejecutable. (López Roschiano & Pech Montejo, 2015)

2.4.13. Estructura estática del proceso.

Teniendo en cuenta que un proceso describe quién hace qué, cómo y cuándo. La metodología RUP se puede representar usando 4 elementos del modelado. (López Roschiano & Pech Montejo, 2015)



Figura 6. Estructura metodología RUP. Adaptado de Estructura estática RUP, por A. Valdez, (2015)

- Roles, definen el “quien”.
- Actividades, establecen el “cómo”.
- Artefactos, definen el “qué”.
- Flujo de trabajo, caracterizan el “cuando”. (López Roschiano & Pech Montejó, 2015)

Se utilizará la metodología RUP, ya que esta se acomoda al software que se va a implementar, en esta se cumple explícitamente los roles y procesos, y se adapta, perfectamente al tipo de desarrollo web a estructurar.

3. Marco teórico y estado del arte

3.1. Marco teórico

3.1.1. Inventarios.

El inventario representa la existencia de bienes almacenados destinados a realizar una operación, sea de compra, alquiler, venta, uso o transformación y de esta manera asegurar el servicio a los clientes internos y externos. El inventario de mercancías lo constituyen todos aquellos bienes que le pertenecen a la empresa bien sea comercial o mercantil, los cuales los compran para luego venderlos sin ser modificados. En esta Cuenta se mostrarán todas las mercancías disponibles para la Venta. (Expert, 2020)

3.1.2. Sistema de inventario periódico.

Este tipo de sistema busca determinar el costo total de las existencias de mercancías por medio de la realización de un conteo realizado de forma física y en forma periódica, el cual según su periodo de realización se puede denominar como:

- Inventario inicial: Es la relación detallada y minuciosa de las existencias de mercancías que tiene una empresa al iniciar sus actividades, después de hacer un conteo físico
- Inventario final: Es la relación de existencias al finalizar un periodo contable.

3.1.3. Sistema de inventario permanente.

Por medio de este sistema la empresa conoce el valor de la mercancía en existencia en cualquier momento, sin necesidad de realizar un conteo físico, porque los movimientos de compra y venta de mercancías se registra directamente en el momento de realizar la transacción a su precio de costo.

Las empresas que adoptan este tipo de sistema deben llevar un auxiliar de mercancías denominado «Kárdex», en el cual se registra cada artículo que se compre o que se venda. La suma

y la resta de todas las operaciones en un periodo da como resultado el saldo final de mercancías. (Giovanny, 2001)

3.1.4. Métodos para la valoración de inventarios.

Las empresas deben valorar sus mercancías, para así valorar sus inventarios, calcular el costo, determinar el nivel de utilidad y fijar la producción con su respectivo nivel de ventas. Actualmente se utilizan los siguientes métodos para valorar los inventarios:

Valoración por identificación específica: En las empresas cuyo inventario consta de mercancías iguales, pero cada una de ellas se distingue de los demás por sus características individuales de número, marca o referencia y un costo determinado.

Valoración a costo estándar: Este método facilita el manejo del auxiliar de mercancías «Kárdex» por cuanto sólo requiere llevarse en cantidades por unidades homogéneas.

Valoración a precio de costo: Valorar el inventario a precio de costo significa que la empresa relaciona las mercancías al precio de adquisición. (Giovanny, 2001)

3.1.5. Kárdex o auxiliar de la cuenta.

KÀRDEX												
Artículo:			lavadoras			Existencia mínima:			60			
Método:			Promedio ponderado			Existencia máxima:			495			
Fecha			Detalle	Entradas			Salidas			Existencias		
D	M	A		Cantidad	V/ Unitario	V/ Total	Cantidad	V/ Unitario	V/ Total	Cantidad	V/ Unitario	V/ Total
3	5	11	Saldo anterior							98	94	9212
5	5	11	Compra según factura N°20	18	134	2412				116	100,21	11624
7	5	11	Venta según factura N°01				67	100,21	6714,07	49	100,2	4909,93
9	5	11	Venta según factura N°02				17	100,2	1703,4	32	100,2	3206,53
11	5	11	Compra según factura N°35	95	135	12825				127	126,23	16031,5
			Inventario Final							127	126,23	16031,5

Figura 7. Estructura del Kardex. Adaptado de ejemplo de Kardex, por Economía y negocios, (2019)

El kárdex constituye un auxiliar de la cuenta Mercancías de productos terminados fabricados o no por la empresa, en la que la suma de los saldos de las tarjetas representa el total de las mercancías en existencia, a precio de costo. Los rayados de estas tarjetas son muy variados, dependiendo de las necesidades de información de cada empresa, y se consiguen preimpresas en el mercado. El que aquí mostraremos es un rayado que nos parece muy práctico y sencillo. En la actualidad los kárdex generalmente se manejan por computadora, recordemos que la máquina hace las mismas operaciones, pero con mayor velocidad y precisión. (Mendez, 2017)

Dentro de este sistema la cuenta que se establece para cada artículo reemplaza la cuenta de compras, que solamente se usa en el sistema de inventarios periódicos. En dicha cuenta se lleva el registro completo del movimiento al valor de costo, tanto de las entradas por compras o devoluciones en ventas, como de las salidas por venta o por devoluciones en compras. (Mendez, 2017)

Resuelto el problema de costo de cada una de las unidades en existencia, bien sea por valor de compra o de producción, la decisión siguiente es el valor que se da a las unidades que salen del inventario y a las que quedan cuando existen unidades homogéneas con distintos costos unitarios. Con este fin se han inventado muchísimos métodos, de los cuales aquí vamos a enseñar solamente los más conocidos y que estimamos de mayor uso en nuestro medio. (Mendez, 2017)

Como primera medida, para cumplir con el principio conservador o de prudencia generalmente aceptado en la contabilidad, se deben registrar los inventarios al costo y nunca al precio de venta. Más aún, hay empresas que registran al costo o al precio del mercado, el que sea más bajo. Pero cuando se habla de precio de mercado nos estamos refiriendo al valor en que podemos comprar en el mercado, o costo de reposición, y no al precio en que esperamos vender. (Mendez, 2017)

3.1.6. Características de un sistema de gestión de inventario.

- Dinámico: debe actualizarse automáticamente al registrar nuevas compras o ventas.
- Segmentado: debe permitir llevar el inventario segmentado por categorías.
- Flexible: Debe permitir actualizar los datos ingresados en tiempo real y realizar modificaciones de manera manual.
- Compatible: Que sea compatible con Excel permitirá importar listas de precios y productos cuando sea necesario.
- Sencillo e intuitivo: sencillo e intuitivo a la hora de utilizar las funciones que corresponden a todas las áreas de la empresa y tener un fácil acceso al historial de movimientos de productos, para entender qué está pasando con el stock. (Vergelin, 2017)

3.2. Estado del arte

3.2.1. Actualidad del desarrollo de software.

Hoy en día es casi imposible imaginar una vida sin dispositivos tecnológicos, ya que se han convertido en una gran herramienta de trabajo, ayudando a que podamos hacer actividades de manera sencilla y sin mucho esfuerzo. Aunque nadie le da importancia, todos estos beneficios se dan gracias a la implementación de software, el cual en vista de los avances tecnológicos se ha popularizado, convirtiéndose en una herramienta muy demandada, no solo por las empresas y personas, sino también por las instituciones de educación, tanto para su propio uso como para sus programas educativos, siendo al día de hoy, la ingeniería; una de las carreras más buscadas por los jóvenes. De este modo, el software ha pasado de ser una herramienta solo para genios en matemáticas y computación, al estar al alcance de cualquier persona.

El software en cualquiera de sus tipos (de sistema, aplicación o programación) en conjunto con el hardware (la parte de un sistema informático que podemos tocar) nos han ayudado con el avance

de la comunicación en tiempo real, la digitalización de documentos, la programación de actividades, la traducción de textos, el almacenamiento de información y en muchas otras diversas actividades o procesos cotidianos que de cierto modo han mejorado nuestra calidad de vida y que seguramente lo seguirán haciendo.

El desarrollo de software es una actividad muy común actualmente y las grandes empresas que se dedican a esta labor, como Microsoft, IBM y Oracle nunca paran de trabajar e innovar, ya que estas deben mantener sus productos actualizados y a la vanguardia tecnológica para cumplir con la demanda de sus consumidores; lo que implica para estas marcas un arduo trabajo de investigación y desarrollo, para poder conocer a fondo las necesidades de sus usuarios, pero también un desborde de creatividad, para impresionarlos y crear productos mucho más impactantes. Esto muchas veces nos hace creer que a futuro ya no habrá nada nuevo por hacer en el ámbito del software, independientemente del sector o la finalidad de los productos desarrollados. (Comunicaciones, 2019).

3.2.2. Legislación de inventarios.

Implementación de norma internacional de inventarios en Colombia, en este artículo de la revista Innovar de la Universidad Nacional de Colombia se describe como la implementación de las normas internacionales de información financiera (NIIF) en Colombia siguen la tendencia de globalización y armonización de la información contable en nuestro país, con el fin de permitir que los estados financieros de las empresas colombianas sean comparables entre países.

Se explica como dicha comparabilidad es vital, toda vez que les permite ahorrar tiempo y dinero a las empresas, en cuanto al proceso de conversión de estados financieros. Se mencionado que dentro de las partidas que van a sufrir un cambio importante se encuentra la partida de inventarios, la cual es nuestro punto de atención, pues, la valuación, el costo y presentación de los inventarios

se modifican, lo cual tendrá un efecto en los estados financieros, sus indicadores y la determinación de los impuestos de las compañías colombianas, y al afectar estos procesos se deben tener en cuenta a la hora de diseñar el software de gestión de inventarios correspondiente. (Universidad Nacional de Colombia, 2014).

Se realiza un análisis comparativo entre el antiguo PCGA, la norma fiscal y obviamente las NIIF para pymes (Sección 13) que se refiere al tema de Inventarios.

Para iniciar en esta comparación se debe conocer el Decreto 3022 de 2013, en su sección 13. Inventarios, donde reza textualmente “Establece los principios para el reconocimiento y medición de los inventarios. Inventarios son activos con los siguientes requisitos: (a) mantenidos para la venta en el curso normal de las operaciones; (b) en proceso de producción con vistas a esa venta; o (c) en forma de materiales o suministros, 'para ser consumidos en el proceso de producción, o en la prestación de servicios”.

Lo que para el autor genera una diferencia importante con la norma colombiana, es que el literal c) no lo contemplaba el PCGA. (Castrillon & Zapata, 2016)

3.2.3. Software de inventarios.

Ofima



Figura 8. Logo ofima S.A.S. Adaptado de Ofima SAS, por OFIMA, (2018)

Es un software de inventarios, que administra, controla inventarios de forma rápida, realiza seguimiento a pedidos y despachos en tiempo real, realiza el control y valorización de los inventarios, ofrece productos en proceso y productos terminados. Maneja el control y valorización de inventarios, ofrece traslados entre bodegas, solicitudes, reservas, conteo físico, administra todo el proceso comercial iniciando por cotizaciones, generando automáticamente los pedidos cotizados, finalizando con la facturación y entrega de mercancías; además apoya eficientemente el proceso de cobranza de la compañía.

Con este software, además, puede conocer exactamente la rotación que tiene cada uno de los productos y los de consignación, se mantiene informado al cliente sobre el valor total de su inventario en tiempo real, y no se producirá más sobre stock en el inventario. (OFIMA S.A.S Lideres en Software Empresarial, 2016)

Orión plus Software contable



Figura 9. Logo Orión Plus. Adaptado de Orión Plus, por ORIÓN PLUS, (2019)

Se caracteriza por ser un completo sistema POS para cualquier punto de venta, ofrece un sistema de facturación a crédito con funciones parametrizables, su búsqueda de artículos se realiza por la función autocompletar, tiene lectura de código de barras, todas sus versiones tienen múltiples medios de pago, las ventas se pueden realizar de contado y/a crédito y además es configurable para múltiples usuarios. (ABP Soluciones S.A.S, 2014)

Alegra

Alegra es un software diseñado específicamente para pequeñas y medianas empresas, ayuda a gestionar como propietario o gerente la parte contable, de facturación y administración de su empresa. Esta aplicación cuenta con un servicio en la nube, robusto y potente pero sencillo e intuitivo. Lo más grandioso de todo es lo fácil de usar. Si aún no estás usando Alegra te contamos las ventajas de usarlo.



Figura 10. Logo Alegra. Adaptado de Alegra Colombia, por Soluciones Alegra SAS, (2018)

Es un servicio a través de internet, lo cual brinda la libertad de acceder desde donde quieras y cuando quieras. Sólo se requiere de un equipo o dispositivo móvil conectado a Internet. (ALEGRA, 2019)

Siigo

Es un software contable y administrativo para micro, pequeñas y medianas empresas. Cotiza factura y recauda en pocos minutos.

Con el fin de transformar la vida de empresarios y contadores a través de herramientas tecnológicas enfocadas en las PYMES de Colombia y Latinoamérica, nace hace 30 años Siigo, empresa colombiana que ofrece un software contable y administrativo 100% en la nube. Es una forma ágil, sencilla y eficiente para controlar las finanzas de las empresas.



Figura 11. Logo Siigo. Adaptado de SIIGO, por Logo SIIGO, (2019)

Durante estos años en el mercado, el trabajo de Siigo ha sido de gran impacto, logrando que en el 2017 ingresaran al fondo de capital privado Accel -KKR, líder del sector en Silicon Valley, convirtiendo esta inversión en la primera que se hizo en América Latina. (SIIGO, 2018)

POS

Es un software de punto de venta hecho en Bucaramanga. El software POS Colombia por su sigla Point Of Sale(Punto de venta) es un programa para tiendas, micro mercados, droguerías, restaurantes, almacenes de ropa, calzado entre otros, que desean sistematizar el control de sus ventas e inventarios. Es un programa para facturación y para administrar cualquier comercio.



Figura 12. Logo POS. Adaptado de POS Colombia, por Software POS Colombia, (2018)

Este sistema POS Colombia ha sido creado en Bucaramanga por el equipo de desarrollo de PromWebSoft. Un software POS está dirigido al propietario del negocio y cualquier usuario puede entenderlo. (POS Colombia, 2018)

Loggro

Loggro es desarrollada por SENSAS, que actualmente comercializa y desarrolla soluciones de gestión empresarial ERP con más de 30 años de trayectoria en el mercado. Sensas cuenta con más

de 150 especialistas en soluciones empresariales ERP y hoy por hoy es una compañía líder en software ERP en Latinoamérica. Compañía multinacional de origen colombiano que ha desplegado miles de proyectos ERP exitosos en el país. (LOGGRO, 2019)



Figura 13. Logo Loggro. Adaptado de Loggro, por Sensas, (2020)

Logimov

Logimov es un software de logística desarrollado por el holding Grupo NW S.A.S desde hace 8 años, el cual se actualiza constantemente de acuerdo a las necesidades que el mercado presente. Este sistema brinda un completo control de inventarios y facturación, con la ventaja de ser un diseño totalmente WEB, permitiéndole a sus usuarios conectarse desde cualquier lugar del mundo a cualquier hora del día. (LogiMov, 2019)



Figura 14. Logo Logimov. Adaptado de Logimov, por Logimov Software de Logística, 2020

Softland Pyme

Es un software completo para pymes el cliente adapta su solución de acuerdo a sus necesidades cambiantes en módulos y usuarios que requiere para su gestión. Desde 1982, Softland emprendió

un camino, mediante el cual más de 35 mil clientes han logrado la digitalización de sus procesos y la simplificación en sus actividades diarias. (Softland Go, 2020)



Figura 15. Logo Softland go. Adaptado de Softland HCM, por Softland Colombia, (2020)

Software TNS

El Software VISUAL TNS es un sistema contable y administrativo integrado para entidades del sector oficial y privado. VISUAL TNS es un Software Contable.

Sencillo, Robusto y con muchos años en el mercado.

En el año 2018 estrena su nuevo software llamado portal tns en la nube eficaz para las empresas colombianas. (TNS Software, 2019)



Figura 16. Logo TNS. Adaptado de Software TNS, por TNS software, (2020)

Fuel Control Solutions

Fuel Control Solutions fue creada en el año 2011 para desarrollar el sistema de información de soporte a los procesos comerciales, cartera e inventarios de una de las principales redes de estaciones de servicios en Colombia. (Fuel Control Solutions, 2019)



Figura 17. Logo FCS. Adaptado de FCS, por Fuel Control Solutions, (2020)

Consensus(SAP)

Consensus es una compañía colombiana establecida por un grupo de consultores experimentados que se apasionaban por resolver los problemas de sus clientes. Después de muchos años de crecimiento continuo, Consensus ha ayudado a más de 900 compañías en todo el mundo a simplificar, racionalizar y automatizar sus organizaciones. (Consensus SAP, 2019)



Figura 18. Logo Consensus. Adaptado de Consensus, por Consensus SAP, (2020)

Helisa

Helisa es un software diseñado y desarrollado para satisfacer las necesidades de las pequeñas y medianas empresas en aspectos económicos orientado a la organización de la contabilidad, además de tener diferentes módulos para cumplir con funciones administrativas y realizar documentación e informes de las transacciones que se llevan a cabo. Helisa NIIF ofrece al usuario la posibilidad de interactuar de una manera ágil y segura respecto de la información almacenada por tablas en Bases de Datos específicas para el cumplimiento y capacidad necesaria y requerida por la empresa para contener información. (Helisa NIIF, 2020)



Figura 19. Logo Helisa. Adaptado de Helisa, por Helisa Software para el trabajo, (2020)

Xubio

Xubio es una solución para pequeñas empresas que se encuentra en la nube, con este sencillo software, el usuario podrá llevar a cabo diferentes procesos, tales como facturación, cotizaciones, contabilidad, impuestos, entre otras actividades con las cuales una empresa realiza comúnmente sus actividades normales. (Xubio SAS, 2020)



Figura 20. Logo Xubio. Adaptado de Xubio, por Xubio LLC, (2020)

4. Especificación de requisitos de software

4.1. Perspectiva del producto

La aplicación desarrollada ofrece una interfaz gráfica que permite simular el comportamiento de un sistema de inventarios, dentro del cual se ofrece información de los productos, así como de las múltiples operaciones que se desarrollan en el día a día de este tipo de sistemas.

4.2. Funcionalidad del producto

La funcionalidad de la aplicación se presenta a continuación, con la característica que el programa solo tiene un usuario único que tiene todas las funciones presentadas y el administrador correspondiente.

Usuario estudiante

- Autenticación: Cada usuario debe autenticarse en el sistema para utilizar sus servicios, para este proceso se presentan dos cajas de texto en las cuales se debe incluir el nombre de usuario y la contraseña correcta para utilizar la aplicación.
- Crear usuario: Cualquier usuario puede crear una persona, que para el sistema puede ser un cliente o un proveedor, se diligencian los diferentes campos obligatorios, validar la creación y el cuadro de confirmación.
- Crear proveedor: Todos los usuarios pueden crear bodegas, esta entidad no tiene restricciones únicamente se debe diligenciar los campos necesarios y confirmar su creación.
- Crear producto: Cualquier usuario puede crear un producto, es importante tener una categoría en la cual relacionarlo diligenciar todos los campos de texto que se presentan y validar la creación y el cuadro de confirmación.

- Crear factura: Todos los usuarios pueden crear facturas tanto de venta como de compra, igual que las anteriores entidades se deben diligenciar los campos obligatorios y confirmar su creación.
- Eliminar cliente: Cualquier usuario puede eliminar una persona, ya sea cliente o proveedor.
- Eliminar producto: Se eliminan los productos del sistema.
- Eliminar factura: Las facturas se pueden eliminar por parte de cualquier usuario.
- Modificar usuario: Cualquier campo de la entidad persona se puede modificar.
- Modificar producto: Cualquier campo de la entidad producto se puede modificar.
- Modificar factura: Cualquier campo de la entidad factura se puede modificar.
- Consultar cliente: Se pueden consultar un cliente o varias e igual que cada uno de sus campos.
- Consultar producto: Se pueden consultar un producto o varios e igual que cada uno de sus campos.
- Consultar factura: Se pueden consultar una factura o varias e igual que cada uno de sus campos.

*Todas las acciones están supeditadas a las reglas de integridad de la base de datos.

- Presentar informes de los productos actuales: El usuario puede presentar diferentes informes catalogados por todos los criterios que maneja la base de datos del programa.
- Calcular el costo del inventario actual: El usuario puede realizar el cálculo del costo para el inventario total en una fecha específica.
- Calcular el valor de las ventas por fecha: las ventas este usuario las puede calcular de acuerdo a un periodo de tiempo determinado.
- Cerrar sesión: Este usuario puede finalizar su sesión, para ser enviado a la página de inicio.

Usuario docente

- Crear usuario: Este usuario puede crear nuevos usuarios.
- Eliminar usuario: Este tipo de usuario puede eliminar los usuarios del sistema.
- Modificar datos usuarios: El usuario administrar puede modificar la información correspondiente a los usuarios del programa.

*Además este usuario cuenta con todas las funciones del usuario anterior.

4.3. Características del usuario

El programa cuenta con dos tipos de usuarios, un administrador y un usuario general que tiene todas las funciones necesarias para simular el sistema de inventarios.

Usuario estudiante

Este tipo de usuario cuenta con todas las características necesarias para realizar la simulación de un sistema de administración de inventario.

Administrador(docente)

Este usuario tiene a su cargo la gestión de la base de datos, por lo tanto, tiene la potestad de crear los usuarios generales del programa.

4.4. Restricciones

Por el tipo de desarrollo web, para su utilización los usuarios deben contar con por lo menos un equipo de cómputo que posea un navegador instalado y una conexión a internet debidamente establecida.

4.5. Suposiciones y dependencias

La aplicación informática desarrollada puede ser utilizada sin importar el tipo de hardware o software que sea utilizado por lo que es conocida como una aplicación multiplataforma. La dependencia única se encuentra relacionada con el servidor web dentro del cual se encuentre

alojado el índice de la aplicación porque este debe estar en capacidad de soportar el lenguaje PHP y MySQL.

También se debe suponer que como la interfaz del programa se realizó en observación del navegador Chrome, este tipo de navegador sea el apropiado para su utilización debido a que en otros navegadores se pueden producir errores en su despliegue, correspondientes a características de diseño que no producen problemas en el desarrollo de las actividades funcionales del programa.

4.6. Requisitos específicos

4.6.1. Actores.

Tabla 1

Actor Administrador

ID	Act-01
Rol	Docente
Función en el programa	Encargado de gestión de la base de datos y la calificación de los ejercicios.
Nivel de conocimiento	Alto
Privilegios en el sistema	Tiene como superusuario todas las funciones creadas para el sistema de administración de almacenes e inventarios.

Nota: Autoría propia.

Tabla 2

Actor Usuario estudiante

ID	Act-02
Rol	Usuario estudiante
Función en el programa	Encargado de realizar los ejercicios correspondientes a la administración de almacenes e inventarios.
Nivel de conocimiento	Medio
Privilegios en el sistema	Tiene los privilegios para crear, modificar y eliminar todas las entidades que hacen parte de un sistema de administración de almacenes e inventarios.

Nota: Autoría propia.

4.6.2. Requisitos funcionales.

- El sistema contara con un control de acceso, por lo tanto, administrador y usuarios deben ingresar al sistema mediante el usuario y la contraseña válidos.
- El sistema realizara la búsqueda, creación, modificación, eliminación de productos.
- El sistema realizara la búsqueda, creación, modificación, eliminación de clientes.
- El sistema realizara la búsqueda, creación, modificación, eliminación de operaciones.
- El sistema realizara la búsqueda, creación, modificación, eliminación de facturas.
- El sistema realizara la búsqueda, creación, modificación, eliminación de proveedor.
- El sistema realizara la búsqueda, creación, modificación, eliminación de usuarios.
- Los actores del sistema pueden visualizar los proveedores del sistema.
- Los actores del sistema pueden visualizar las facturas del sistema.
- Los actores del sistema pueden visualizar los productos del sistema.
- El sistema actualizara automáticamente el inventario de todas las operaciones realizadas.
- El sistema generara reportes de la cantidad de productos dentro de un periodo de tiempo especificado.
- El sistema genera un valor del costo total del inventario para un día especificado.

4.6.3. Diagrama de casos de uso.

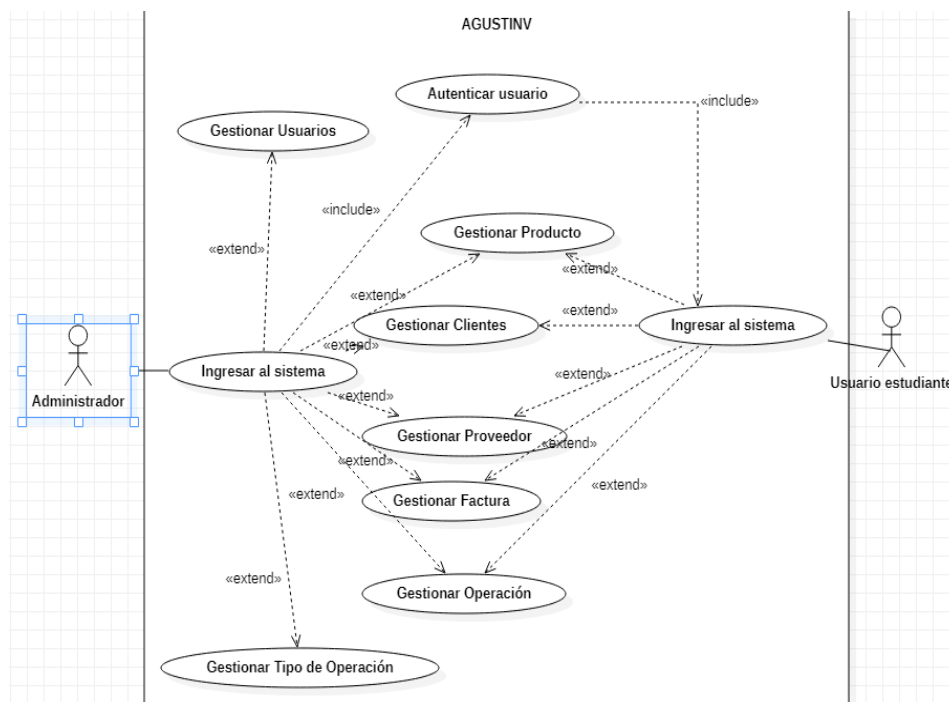


Figura 21. Diagrama general casos de uso. Autoría propia.

Los casos de uso denominados gestionar, agrupan los casos de usos generales para cada entidad, esto quiere decir que agrupa los casos de uso de creación, modificación y eliminación de cada una de las tablas, esto se realiza para facilitar el diseño de la ilustración y su presentación en la herramienta StarUML.

4.6.4. Especificaciones de los casos de uso.

Tabla 3

Caso de Uso Autenticación

Caso de Uso	CS01	Tipo requerimiento	Funcional
Supuesto	El docente y el usuario estudiante debe encontrarse correctamente validados.		
Descripción	La aplicación debe solicitar los datos de ingreso a los diferentes usuarios.		
Razón	La aplicación en la interfaz inicial solicitada los datos de ingreso.		
Criterio de medición	La aplicación permite a los usuarios autenticarse para desarrollar sus actividades.		
Prioridad	Alta.		

Nota: Autoría propia.

Tabla 4

Caso de uso creación de productos

Caso de Uso	CS06	Tipo requerimiento	Funcional
Supuesto	El docente y el usuario estudiante deben encontrarse correctamente validados.		
Descripción	La aplicación debe permitir la creación de productos.		
Razón	La aplicación debe ofrecer un módulo que permita a los usuarios la creación de productos.		
Criterio de medición	La aplicación permite a los usuarios crear productos de forma rápida y eficiente.		
Prioridad	Alta.		

Nota: Autoría propia.

Tabla 5

Caso de Uso modificación de productos

Caso de Uso	CS07	Tipo requerimiento	Funcional
Supuesto	El docente y el usuario estudiante deben encontrarse correctamente validados.		
Descripción	La aplicación debe permitir la modificación de productos.		
Razón	La aplicación debe ofrecer un módulo que permita a los usuarios la modificación de productos.		
Criterio de medición	La aplicación permite a los usuarios modificar los productos de forma rápida y eficiente.		
Prioridad	Alta.		

Nota: Autoría propia.

Tabla 6

Caso de Uso eliminación de productos

Caso de Uso	CS08	Tipo requerimiento	Funcional
Supuesto	El docente y el usuario estudiante deben encontrarse correctamente validados.		
Descripción	La aplicación debe permitir la eliminación de productos.		
Razón	La aplicación debe ofrecer un módulo que permita a los usuarios la eliminación de productos.		
Criterio de medición	La aplicación permite a los usuarios eliminar los productos de forma rápida y eficiente.		
Prioridad	Alta.		

Nota: Autoría propia.

Tabla 7

Caso de uso búsqueda de productos

Caso de Uso	CS09	Tipo requerimiento	Funcional
Supuesto	El docente y el usuario estudiante deben encontrarse correctamente validados.		
Descripción	La aplicación debe permitir la búsqueda de productos.		
Razón	La aplicación debe ofrecer un módulo que permita a los usuarios la búsqueda de productos.		
Criterio de medición	La aplicación permite a los usuarios la búsqueda de productos de forma rápida y eficiente.		
Prioridad	Alta.		

Nota: Autoría propia.

Tabla 8

Caso de uso creación de clientes

Caso de Uso	CS10	Tipo requerimiento	Funcional
Supuesto	El docente y el usuario estudiante deben encontrarse correctamente validados.		
Descripción	La aplicación debe permitir la creación de clientes.		
Razón	La aplicación debe ofrecer un módulo que permita a los usuarios la creación de clientes.		
Criterio de medición	La aplicación permite a los usuarios crear clientes de forma rápida y eficiente.		
Prioridad	Alta.		

Nota: Autoría propia.

Tabla 9

Caso de Uso modificación de clientes

Caso de Uso	CS11	Tipo requerimiento	Funcional
Supuesto	El docente y el usuario estudiante deben encontrarse correctamente validados.		
Descripción	La aplicación debe permitir la modificación de clientes.		
Razón	La aplicación debe ofrecer un módulo que permita a los usuarios la modificación de clientes.		
Criterio de medición	La aplicación permite a los usuarios modificar los clientes de forma rápida y eficiente.		
Prioridad	Alta.		

Nota: Autoría propia.

Tabla 10

Caso de Uso eliminación de clientes

Caso de Uso	CS12	Tipo requerimiento	Funcional
Supuesto	El docente y el usuario estudiante deben encontrarse correctamente validados.		
Descripción	La aplicación debe permitir la eliminación de clientes.		
Razón	La aplicación debe ofrecer un módulo que permita a los usuarios la eliminación de clientes.		
Criterio de medición	La aplicación permite a los usuarios eliminar las personas de forma rápida y eficiente.		
Prioridad	Alta.		

Nota: Autoría propia.

Tabla 11

Caso de uso búsqueda de clientes

Caso de Uso	CS13	Tipo requerimiento	Funcional
Supuesto	El docente y el usuario estudiante deben encontrarse correctamente validados.		
Descripción	La aplicación debe permitir la búsqueda de personas.		
Razón	La aplicación debe ofrecer un módulo que permita a los usuarios la búsqueda de clientes.		
Criterio de medición	La aplicación permite a los usuarios la búsqueda de clientes de forma rápida y eficiente.		
Prioridad	Alta.		

Nota: Autoría propia.

Tabla 12

Caso de uso creación de operaciones

Caso de Uso	CS14	Tipo requerimiento	Funcional
Supuesto	El docente y el usuario estudiante deben encontrarse correctamente validados.		
Descripción	La aplicación debe permitir la creación de operaciones.		
Razón	La aplicación debe ofrecer un módulo que permita a los usuarios la creación de operaciones.		
Criterio de medición	La aplicación permite a los usuarios crear operaciones de forma rápida y eficiente.		
Prioridad	Alta.		

Nota: Autoría propia.

Tabla 13

Caso de Uso modificación de operaciones

Caso de Uso	CS15	Tipo requerimiento	Funcional
Supuesto	El docente y el usuario estudiante deben encontrarse correctamente validados.		
Descripción	La aplicación debe permitir la modificación de operaciones.		
Razón	La aplicación debe ofrecer un módulo que permita a los usuarios la modificación de operaciones.		
Criterio de medición	La aplicación permite a los usuarios modificar las operaciones de forma rápida y eficiente.		
Prioridad	Alta.		

Nota: Autoría propia.

Tabla 14

Caso de Uso eliminación de operaciones

Caso de Uso	CS16	Tipo requerimiento	Funcional
Supuesto	El docente y el usuario estudiante deben encontrarse correctamente validados.		
Descripción	La aplicación debe permitir la eliminación de operaciones.		
Razón	La aplicación debe ofrecer un módulo que permita a los usuarios la eliminación de operaciones.		
Criterio de medición	La aplicación permite a los usuarios eliminar las operaciones de forma rápida y eficiente.		
Prioridad	Alta.		

Nota: Autoría propia.

Tabla 15

Caso de uso búsqueda de operaciones

Caso de Uso	CS17	Tipo requerimiento	Funcional
Supuesto	El docente y el usuario estudiante deben encontrarse correctamente validados.		
Descripción	La aplicación debe permitir la búsqueda de operaciones.		
Razón	La aplicación debe ofrecer un módulo que permita a los usuarios la búsqueda de operaciones.		
Criterio de medición	La aplicación permite a los usuarios la búsqueda de operaciones de forma rápida y eficiente.		
Prioridad	Alta.		

Nota: Autoría propia.

Tabla 16

Caso de uso creación de facturas

Caso de Uso	CS18	Tipo requerimiento	Funcional
Supuesto	El docente y el usuario estudiante deben encontrarse correctamente validados.		
Descripción	La aplicación debe permitir la creación de facturas.		
Razón	La aplicación debe ofrecer un módulo que permita a los usuarios la creación de facturas.		
Criterio de medición	La aplicación permite a los usuarios crear facturas de forma rápida y eficiente.		
Prioridad	Alta.		

Nota: Autoría propia.

Tabla 17

Caso de Uso modificación de facturas

Caso de Uso	CS19	Tipo requerimiento	Funcional
Supuesto	El docente y el usuario estudiante deben encontrarse correctamente validados.		
Descripción	La aplicación debe permitir la modificación de facturas.		
Razón	La aplicación debe ofrecer un módulo que permita a los usuarios la modificación de facturas.		
Criterio de medición	La aplicación permite a los usuarios modificar las facturas de forma rápida y eficiente.		
Prioridad	Alta.		

Nota: Autoría propia.

Tabla 18

Caso de Uso eliminación de facturas

Caso de Uso	CS20	Tipo requerimiento	Funcional
Supuesto	El docente y el usuario estudiante deben encontrarse correctamente validados.		
Descripción	La aplicación debe permitir la eliminación de facturas.		
Razón	La aplicación debe ofrecer un módulo que permita a los usuarios la eliminación de facturas.		
Criterio de medición	La aplicación permite a los usuarios eliminar las facturas de forma rápida y eficiente.		
Prioridad	Alta.		

Nota: Autoría propia.

Tabla 19

Caso de uso búsqueda de facturas

Caso de Uso	CS21	Tipo requerimiento	Funcional
Supuesto	El docente y el usuario estudiante deben encontrarse correctamente validados.		
Descripción	La aplicación debe permitir la búsqueda de facturas.		
Razón	La aplicación debe ofrecer un módulo que permita a los usuarios la búsqueda de facturas.		
Criterio de medición	La aplicación permite a los usuarios la búsqueda de facturas de forma rápida y eficiente.		
Prioridad	Alta.		

Nota: Autoría propia.

Tabla 20

Caso de uso creación de proveedores

Caso de Uso	CS22	Tipo requerimiento	Funcional
Supuesto	El docente y el usuario estudiante deben encontrarse correctamente validados.		
Descripción	La aplicación debe permitir la creación de proveedores.		
Razón	La aplicación debe ofrecer un módulo que permita a los usuarios la creación de proveedores.		
Criterio de medición	La aplicación permite a los usuarios crear proveedores de forma rápida y eficiente.		
Prioridad	Alta.		

Nota: Autoría propia.

Tabla 21

Caso de Uso modificación de proveedores

Caso de Uso	CS23	Tipo requerimiento	Funcional
Supuesto	El docente y el usuario estudiante deben encontrarse correctamente validados.		
Descripción	La aplicación debe permitir la modificación de proveedores.		
Razón	La aplicación debe ofrecer un módulo que permita a los usuarios la modificación de proveedores.		
Criterio de medición	La aplicación permite a los usuarios modificar un proveedor de forma rápida y eficiente.		
Prioridad	Alta.		

Nota: Autoría propia.

Tabla 22

Caso de Uso eliminación de proveedores

Caso de Uso	CS24	Tipo requerimiento	Funcional
Supuesto	El docente y el usuario estudiante deben encontrarse correctamente validados.		
Descripción	La aplicación debe permitir la eliminación de proveedores.		
Razón	La aplicación debe ofrecer un módulo que permita a los usuarios la eliminación de proveedores		
Criterio de medición	La aplicación permite a los usuarios eliminar un proveedor de forma rápida y eficiente.		
Prioridad	Alta.		

Nota: Autoría propia.

Tabla 23

Caso de uso búsqueda de proveedores

Caso de Uso	CS25	Tipo requerimiento	Funcional
Supuesto	El docente y el usuario estudiante deben encontrarse correctamente validados.		
Descripción	La aplicación debe permitir la búsqueda de un proveedor.		
Razón	La aplicación debe ofrecer un módulo que permita a los usuarios la búsqueda de proveedores.		
Criterio de medición	La aplicación permite a los usuarios la búsqueda de proveedores de forma rápida y eficiente.		
Prioridad	Alta.		

Nota: Autoría propia.

Tabla 24

Caso de uso creación de usuarios

Caso de Uso	CS30	Tipo requerimiento	Funcional
Supuesto	El administrador debe encontrarse correctamente validado.		
Descripción	La aplicación debe permitir la creación de usuarios.		
Razón	La aplicación debe ofrecer un módulo que permita al administrador la creación de usuarios.		
Criterio de medición	La aplicación permite al administrador crear usuarios de forma rápida y eficiente.		
Prioridad	Alta.		

Nota: Autoría propia.

Tabla 25

Caso de Uso modificación de usuarios

Caso de Uso	CS31	Tipo requerimiento	Funcional
Supuesto	El administrador debe encontrarse correctamente validado.		
Descripción	La aplicación debe permitir la modificación de usuarios.		
Razón	La aplicación debe ofrecer un módulo que permita al administrador la modificación de usuarios.		
Criterio de medición	La aplicación permite al administrador modificar los usuarios de forma rápida y eficiente.		
Prioridad	Alta.		

Nota: Autoría propia.

Tabla 26

Caso de Uso eliminación de usuarios

Caso de Uso	CS32	Tipo requerimiento	Funcional
Supuesto	El administrador debe encontrarse correctamente validado.		
Descripción	La aplicación debe permitir la eliminación de usuarios.		
Razón	La aplicación debe ofrecer un módulo que permita al administrador la eliminación de usuarios.		
Criterio de medición	La aplicación permite al administrador eliminar usuarios de forma rápida y eficiente.		
Prioridad	Alta.		

Nota: Autoría propia.

Tabla 27

Caso de uso búsqueda de usuarios

Caso de Uso	CS33	Tipo requerimiento	Funcional
Supuesto	El administrador debe encontrarse correctamente validado.		
Descripción	La aplicación debe permitir la búsqueda de usuarios.		
Razón	La aplicación debe ofrecer un módulo que permita al administrador la búsqueda de usuarios.		
Criterio de medición	La aplicación permite al administrador la búsqueda de usuarios de forma rápida y eficiente.		
Prioridad	Alta.		

Nota: Autoría propia.

Tabla 28

Caso de Uso alerta stock mínimo

Caso de Uso	CS34	Tipo requerimiento	Funcional
Supuesto	Los usuarios deben encontrarse correctamente validados.		
Descripción	La aplicación emite una alerta cuando un producto alcanza su stock mínimo.		
Razón	La aplicación lanza una alerta cuando una operación alcanza el stock mínimo de algún producto.		
Criterio de medición	La aplicación alerta a los usuarios con un cuadro de dialogo y un sonido de sistema por stock mínimo de cualquier producto.		
Prioridad	Alta.		

Nota: Autoría propia.

Tabla 29

Caso de Uso stock máximo

Caso de Uso	CS35	Tipo requerimiento	Funcional
Supuesto	Los usuarios deben encontrarse correctamente validados.		
Descripción	La aplicación emite una alerta cuando un producto alcanza su stock máximo.		
Razón	La aplicación lanza una alerta cuando una operación alcanza el stock máximo de algún producto.		
Criterio de medición	La aplicación alerta a los usuarios con un cuadro de dialogo y un sonido de sistema por stock máximo de cualquier producto.		
Prioridad	Alta.		

Nota: Autoría propia.

Tabla 30

Caso de uso generación de reportes

Caso de Uso	CS36	Tipo requerimiento	Funcional
Supuesto	Los usuarios deben encontrarse correctamente validados.		
Descripción	La aplicación debe permitir la generación de reportes de acuerdo a un periodo de tiempo especificado.		
Razón	La aplicación debe ofrecer un módulo para la emisión de reportes periódicos establecidos por los usuarios.		
Criterio de medición	La aplicación permite a los usuarios la creación de reportes específicos.		
Prioridad	Alta.		

Nota: Autoría propia.

Tabla 31

Caso de Uso inventario total

Caso de Uso	CS31	Tipo requerimiento	Funcional
Supuesto	Los usuarios deben encontrarse correctamente validados.		
Descripción	La aplicación debe permitir el cálculo del costo total del inventario en una fecha específica.		
Razón	La aplicación debe ofrecer un módulo que permita encontrar el costo total del inventario en una fecha específica.		
Criterio de medición	La aplicación permite a los usuarios encontrar el costo total del inventario para un día en específico.		
Prioridad	Alta.		

Nota: Autoría propia.

4.7. Requisitos de rendimiento

4.7.1. Rendimiento.

La aplicación debe informar a cada usuario la finalización exitosa de cada acción o en su defecto informar el error presentado y su solución más próxima, una vez ejecutada cada operación.

4.7.2. Seguridad.

La aplicación debe guardar rigurosamente los datos del usuario y sus transacciones, además, debe almacenar el tipo de información utilizada y el usuario que realice cada transacción.

4.7.3. Mantenibilidad.

La aplicación debe ser creada estrictamente por módulos, con el objetivo principal de facilitar su actualización o reemplazo por versiones nuevas.

4.7.4. Usabilidad.

La aplicación debe estar en capacidad de ofrecer de forma rápida y efectiva toda la información que los clientes necesitan para ejecutar sus acciones, la creación de los usuarios debe ser una acción supremamente fácil de entender y realizar.

4.7.5. Disponibilidad.

La aplicación debe estar en la capacidad de soportar la carga total de trabajo, sin que su uso permita bloquear ni cerrar sus aplicativos.

4.8. Restricciones de diseño

La aplicación web usará hardware de forma extra, este dispositivo se trata de un escáner de código de barras, el cual se utiliza a través del puerto USB del equipo.

4.9. Atributos de calidad del software del sistema

La aplicación web será por el momento un prototipo de desarrollo en proceso, que se ira mejorando en el transcurso y tiempo de su elaboración total.

5. Diseño del software (ISO -12207-1)

5.1. Diseño de la arquitectura del software

Esta aplicación se emplearon los métodos y las funciones del patrón de arquitectura MVC (Modelo-vista-controlador).



Figura 22. Modelo Vista – Controlador. Autoría propia

5.2. Diseño detallado del software

5.2.1. Diagrama de clases.

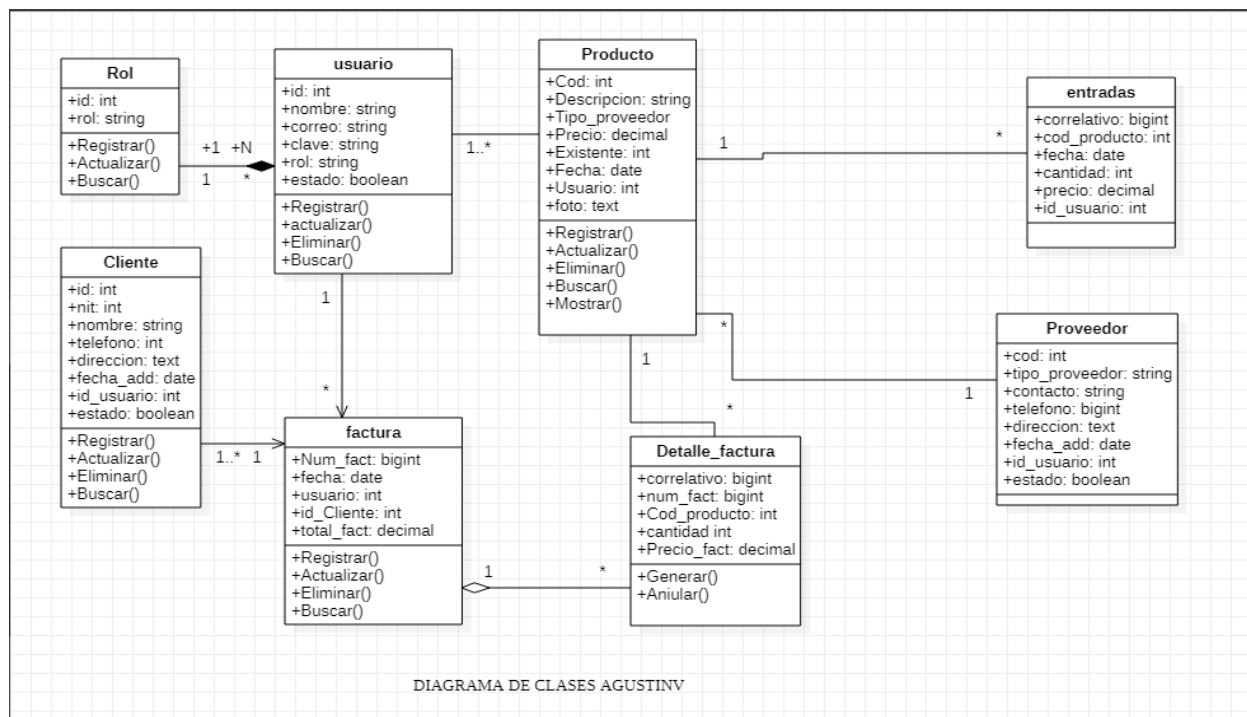


Figura 23. Modelo entidad – Relación. Autoría propia

5.2.2. Diagrama de paquetes.

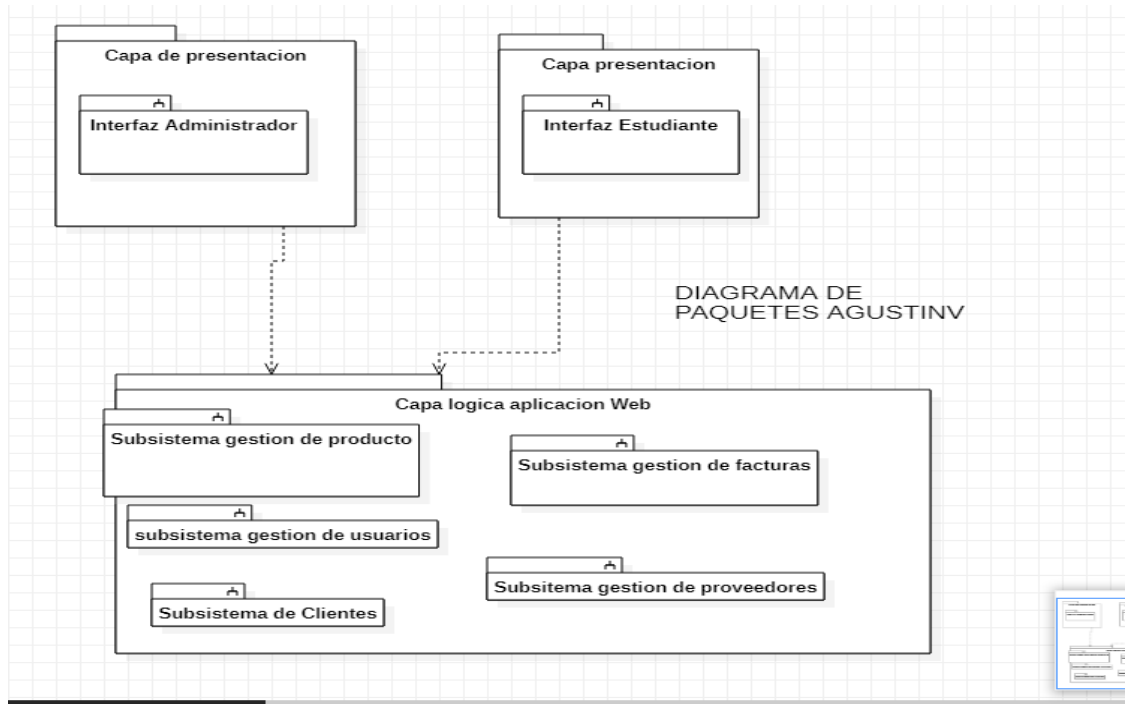


Figura 24. Diagrama de paquetes. Autoría propia

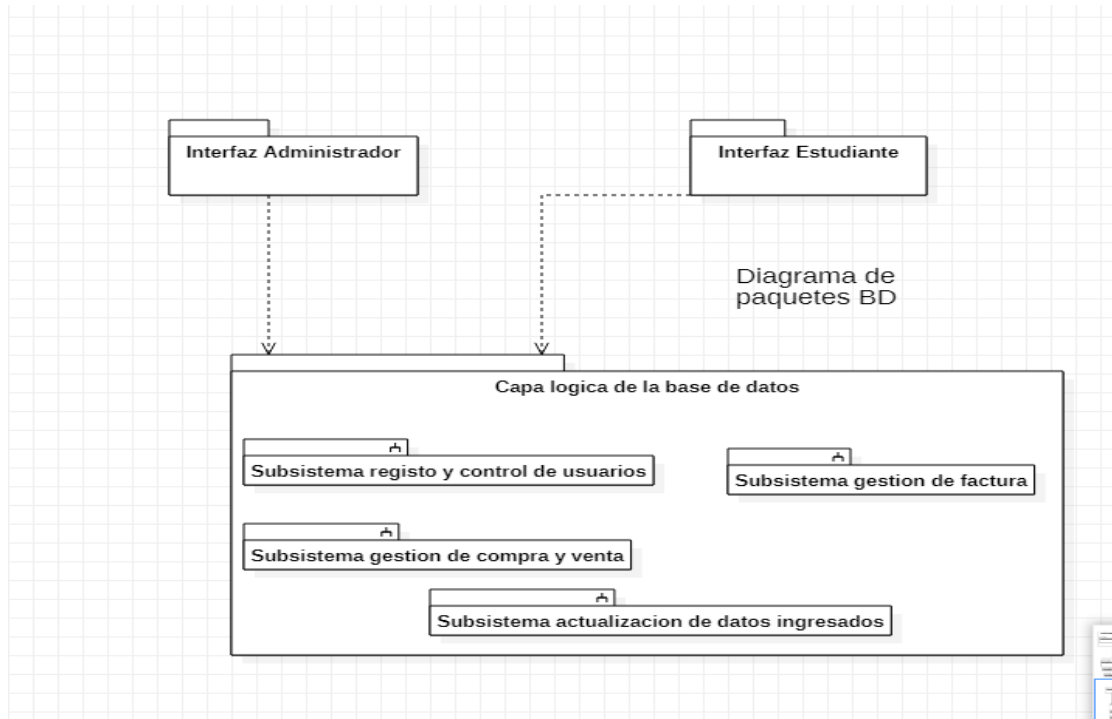


Figura 25. Diagrama de interfaces. Autoría propia

5.2.3. Diagrama de despliegue.

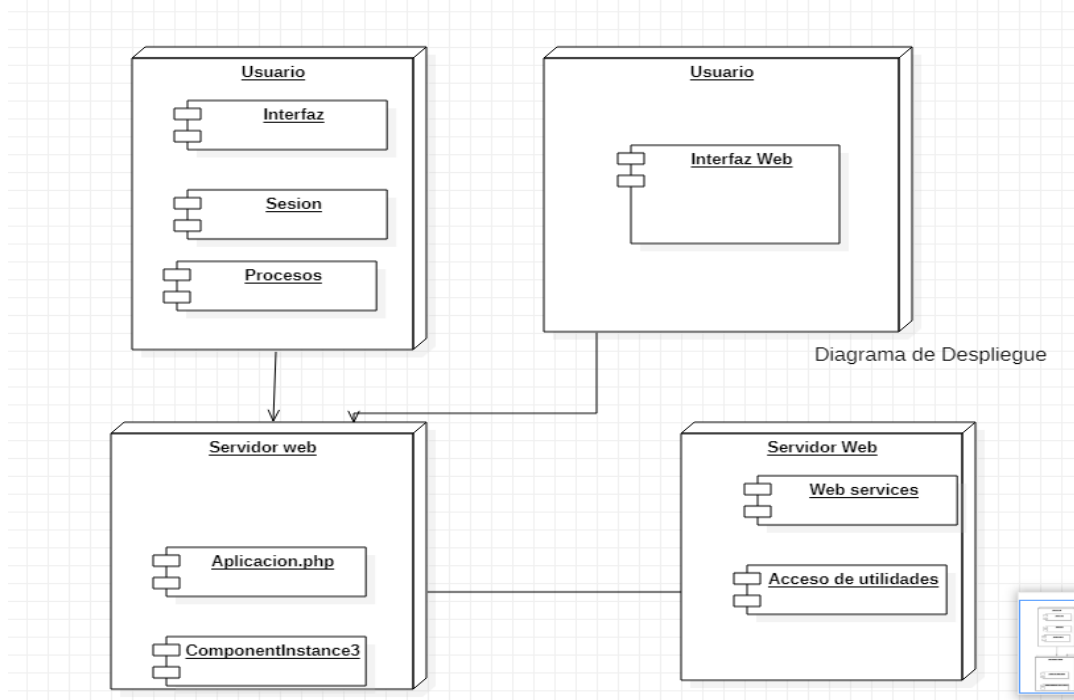


Figura 26. Diagrama de despliegue. Autoría propia

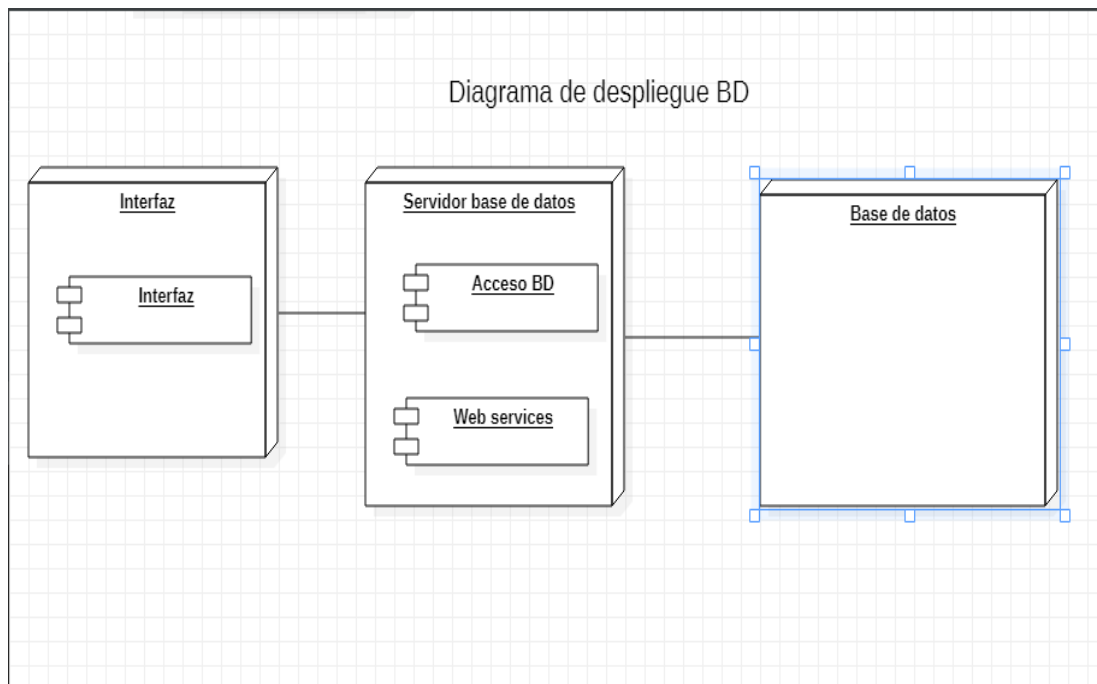


Figura 27. Diagrama de despliegue BD. Autoría propia

5.3. Diseño de interfaz

5.3.1. Interfaz gráfica de usuario.

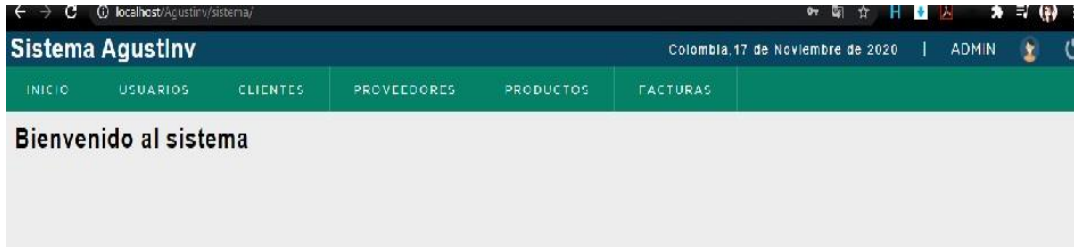


Figura 28. Interfaz Gráfica de usuario. Autoría propia

5.3.2. Interfaces de entrada.

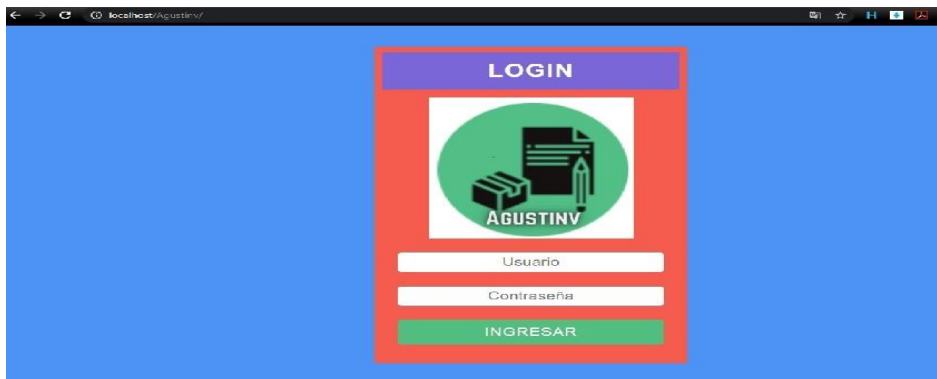


Figura 29. Interfaz de login. Autoría propia

A screenshot of a user registration form titled 'Registro usuario'. The form is on a light gray background. It contains the following fields: 'Nombre' with a text input field containing 'Nombre completo'; 'Correo' with a text input field containing 'Correo electronico'; 'Usuario' with a text input field containing 'Usuario'; 'Clave' with a text input field containing 'Clave de acceso'; and 'Tipo Usuario' with a dropdown menu showing 'Administrador'. At the bottom of the form is a blue button with the text 'Crear usuario'.

Figura 30. Registro de usuarios. Autoría propia

Actualizar usuario

Nombre

Correo electrónico

Usuario

Clave

Tipo Usuario

[Actualizar usuario](#)

Figura 31. Actualización de usuario. Autoría propia

5.3.3. Interfaces de salida.

ID	Nombre	Correo	usuario	Rol	Acciones
1	valentina	valentykar@live.com	admin	Administrador	Editar Eliminar
2	felipe	felipe@gmail.com	Estudiante	Estudiante	Editar Eliminar

Figura 32. Listado de usuarios. Autoría propia



Figura 33. Eliminación de usuarios. Autoría propia

6. Implementación

6.1. Plataformas de desarrollo

Este proyecto se desarrolló en la herramienta gestora de web site “PHP”, es un lenguaje orientado a desarrollar aplicaciones web dinámicas; con información almacenada a una base de datos. En cuanto a la base de datos se utilizó la herramienta phpmyadmin junto a MYSQL como gestor de base de datos, donde se almacenara la información consultada por la sentencia de JQuery y guardada en el servidor localhost; Y por último las herramientas básicas del desarrollo web html y css para el diseño de la interfaz gráfica y JavaScript para realizar cada acción que solicite el sistema.

6.2. Base de datos

6.2.1. Diseño de la base de datos.

El diseño de la base de datos para la aplicación se inicia con el análisis y la implementación de la base de datos, para esta aplicación la base de datos cuenta con ocho (08) tablas y nueve (09) relaciones, las tablas seleccionadas son, producto, operación, factura, roles, clientes, usuario, proveedores y detalle factura.

6.2.2. Modelo entidad relación.

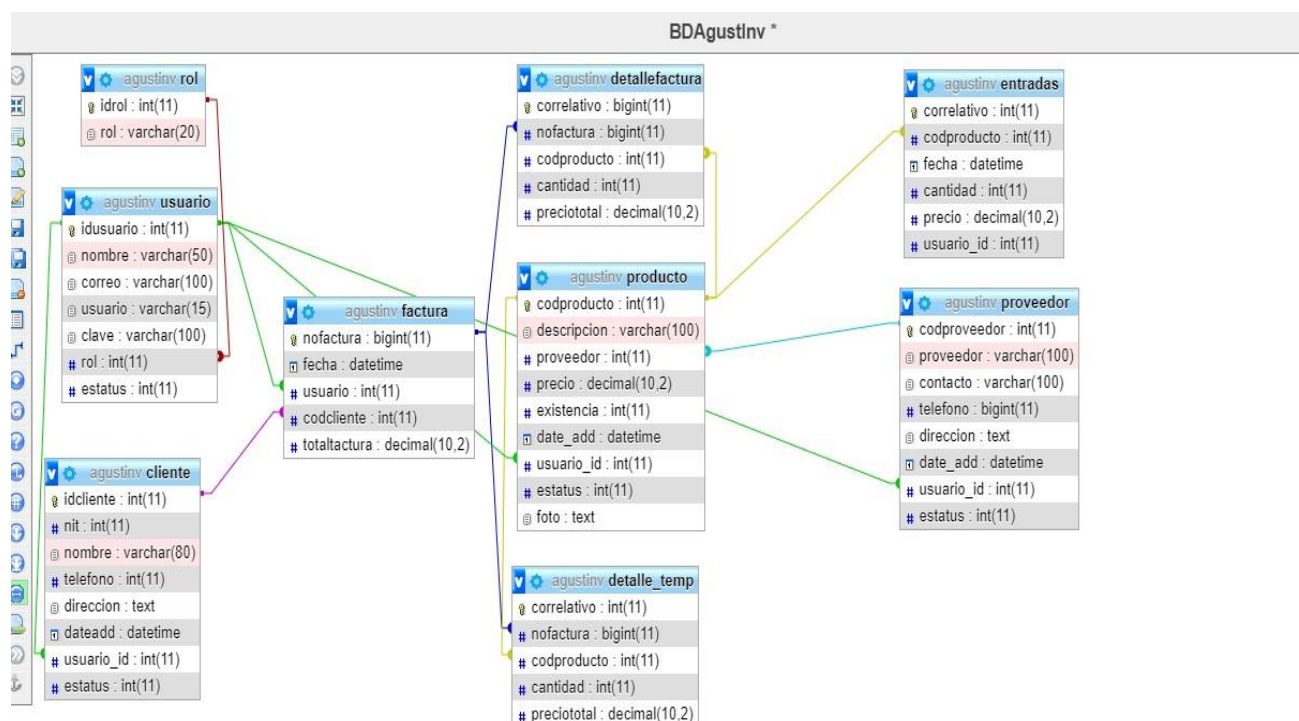


Figura 34. Modelo entidad-relación. Autoría propia

6.2.3. Infraestructura de hardware y redes.

Para desarrollar este proyecto, se trabajó desde un portátil que cuenta con las siguientes características:

Modelo: HP Laptop 15 dy0xxx.

Procesador: Intel(R)Core(TM) i5-8265U CPU @ 1.60GHz 1.80 GHz.

RAM:12.0GB

Sistema Operativo:64-bit Operating System.

Pen and Touch: Touch Support with 10 Touch Points.

El software requiere de una conexión a internet para vincular la base de datos con el servidor localhost phpmyadmin, también requiere de un editor de código en el cual se interrelacionaran las carpetas de los archivos, en el que se ejecutara todas las consultas y almacenamiento de datos y por ultimo MYSQL Worbench, aplicar la sentencia SQL.

7. Pruebas del software

7.1. Pruebas del software

Requisitos funcionales	Cumple	Observaciones
• El sistema contara con un control de acceso, por lo tanto, administrador y usuarios deben ingresar al sistema mediante el usuario y la contraseña válidos	x	
• El sistema realizara la búsqueda, creación, modificación, eliminación de productos.	x	
• El sistema realizara la búsqueda, creación, modificación, eliminación de clientes.	x	
• El sistema realizara la búsqueda, creación, modificación, eliminación de operaciones.		
• El sistema realizara la búsqueda, creación, modificación, eliminación de facturas.	x	
• El sistema realizara la búsqueda, creación, modificación, eliminación de proveedor.	x	
• El sistema realizara la búsqueda, creación, modificación, eliminación de usuarios.	x	
• El sistema actualizara automáticamente el inventario de todas las operaciones realizadas.		
• El sistema generara reportes de la cantidad de productos dentro de un periodo de tiempo especificado.		Falta implementarlo
• El sistema genera un valor del costo total del inventario para un día especificado.		Falta implementacion de graficas
Requisitos no funcionales	Cumple	Observaciones
El sistema funcionará unicamente en computadores de escritorio.	X	
El sistema no cerrará sesión a menos que el usuario lo haga manualmente.	X	
El aplicativo se desarrollara en php, mysql, javascript, jquery,html y css.	X	
Los datos que se inserten en la base de datos se actualiza en 15 segundos	X	
La aplicación web será por el momento un prototipo de desarrollo en proceso, que se ira mejorando en el transcurso y tiempo de su elaboración total	X	
Los permisos de acceso al sistema podrán ser cambiados solamente por el administrador de acceso a datos	X	
El sistema debe proporcionar mensajes de error que sean informativos y orientados a usuario final.		Se debe implementar

Figura 35. Pruebas de software. Autoría propia

7.2. Pruebas de usabilidad`

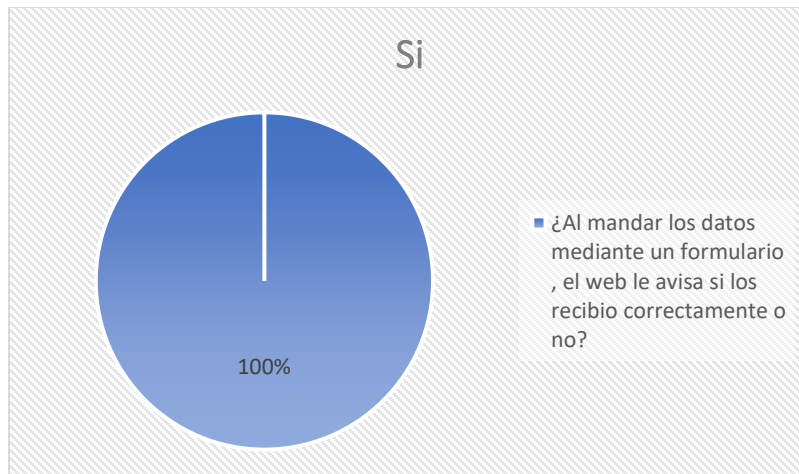


Figura 36. Pruebas de usabilidad. Autoría propia

En esta pregunta 1, el 100% afirmo que el sistema tiene un proceso de autenticación correcto.

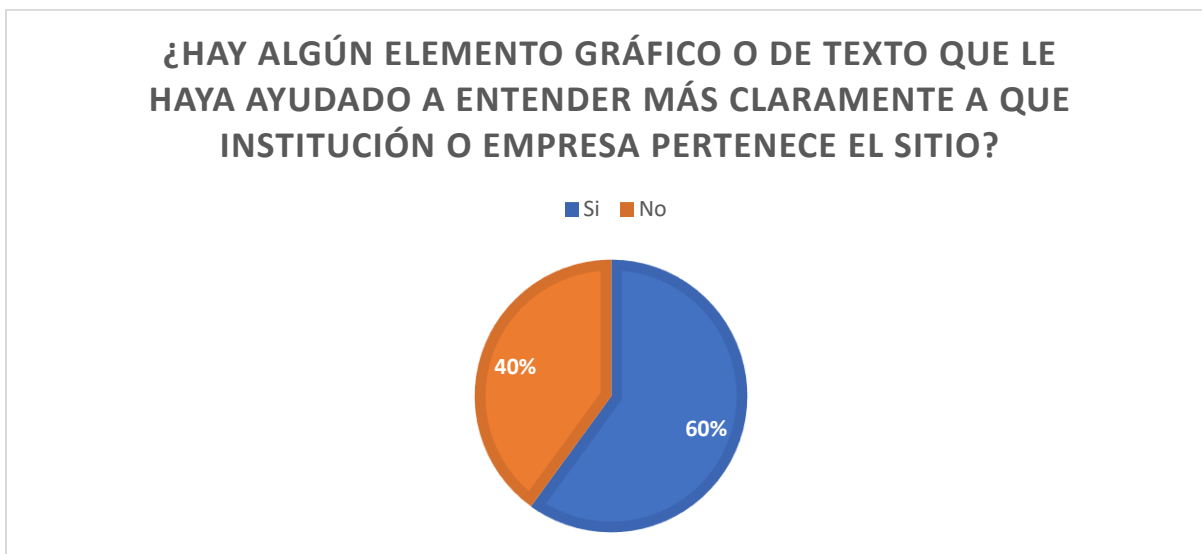


Figura 37. Pregunta de claridad. Autoría propia

En la pregunta 2, el 60 % afirmaron que si se pudo conceptualizar el contexto de interfaz y en cambio solo el 40% no quedo conforme a la temática.



Figura 38. Tipo de audiencia. Autoría propia

En la pregunta 3, el 60% cree que el tipo de audiencia va a las pymes, y en cambio solo el 40% la considera como herramienta didáctica para los estudiantes.



Figura 39. Calidad del sitio. Autoría propia

En la pregunta 4, el 60 % piensa que el sitio se ve muy recargado con el contexto y función, y solo el 40% me sugirió que maneja estaba el contexto simple.

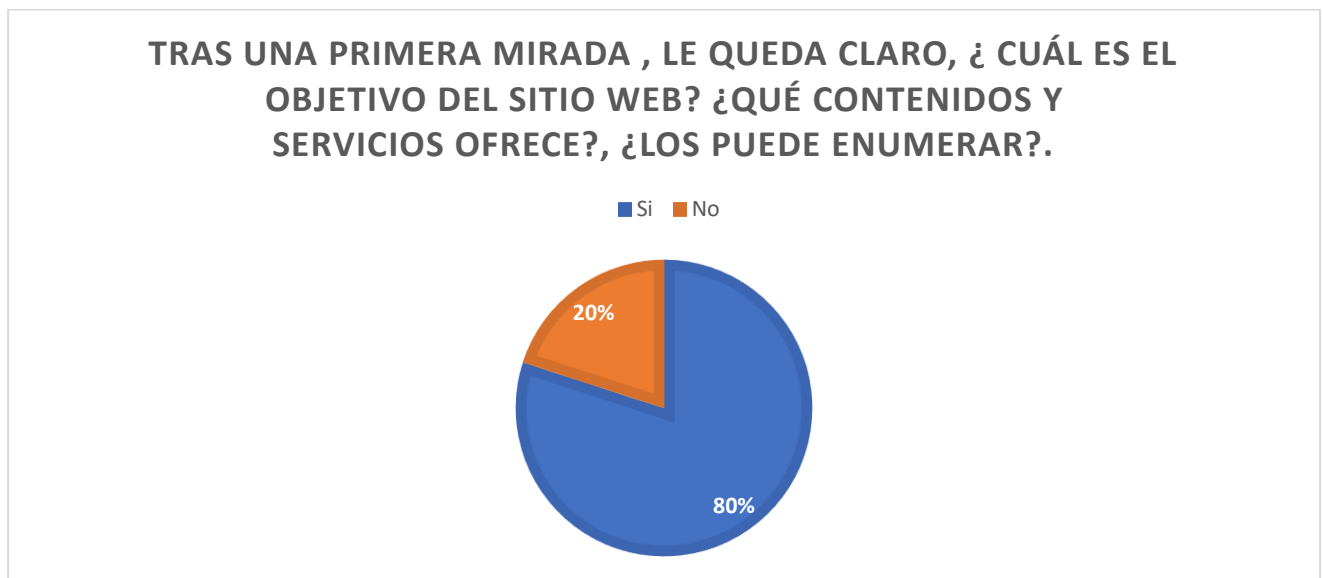


Figura 40. Objetivo del sitio. Autoría propia

En la pregunta 5, se evidenciar que el 80%, se pudo verificar el objetivo del Sistema y el 20%, no sabían mucho del tema.

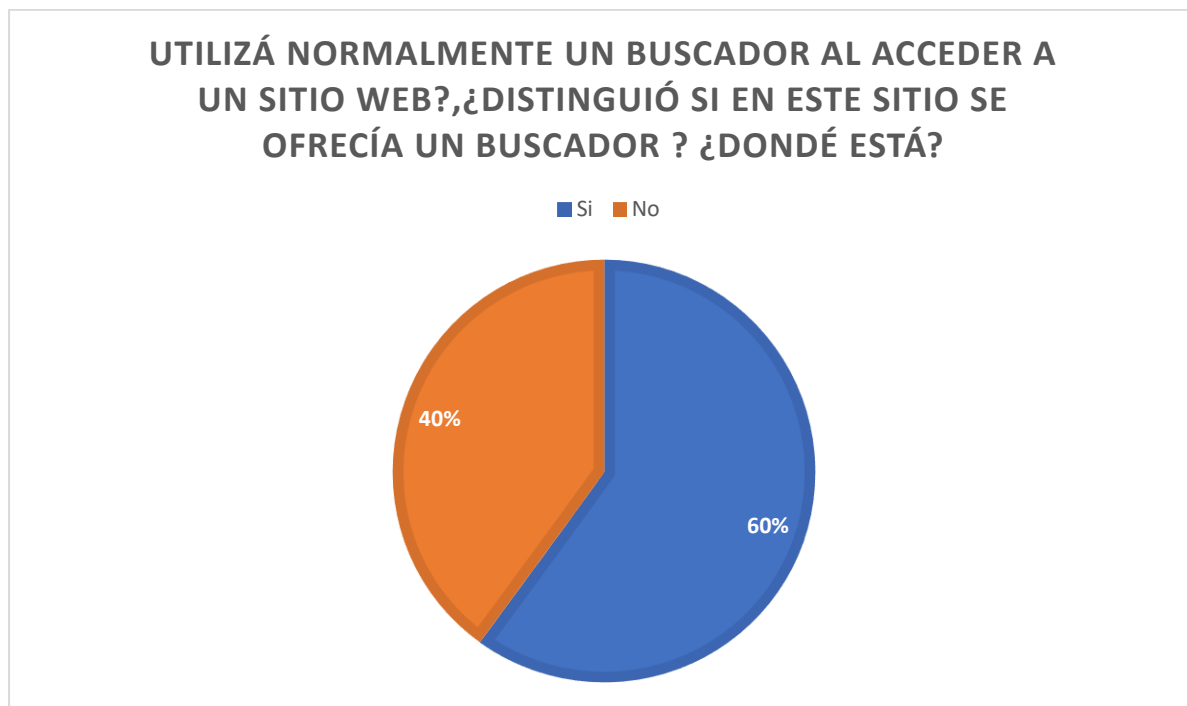


Figura 41. Buscadores. Autoría propia

Y por último se evidencio, que el 60% si se pudo distinguir un buscador para navegar, por la página, y el 40% restante, afirmo que no se pudo ubicar con uno.

Conclusiones

Se concluye del proceso de desarrollo la presentación de un prototipo de sistema de gestión de inventarios con capacidad de registrar diferentes usuarios y roles, y las funciones de creación, actualización, consulta y eliminación de las diferentes entidades externas que hacen parte de un sistema de inventarios general.

De acuerdo a los tiempos de entrega de los prototipos y la posibilidad de implementar una metodología de desarrollo iterativa que es fundamental para este proceso, se selecciona la metodología RUP, pues presenta las características adecuadas para la presentación rápida de prototipos, pero de igual forma asegura la calidad del desarrollo, pues se enfoca en la asignación de tareas específicas durante todas las etapas de desarrollo del software, actividad que produce una mayor eficiencia en el desarrollo y la minimización del tiempo del proyecto.

De acuerdo al análisis realizado es importante establecer las necesidades actuales de las empresas nacionales en lo correspondiente al manejo de inventarios, de acuerdo a lo anterior, se concluye que es necesario presentar un prototipo que permita crear las diferentes entidades de inventarios pero además maneje diferentes niveles de usuario para proteger la información fundamental, en este sentido fue primordial crear una base de datos confiable que garantice la disponibilidad y fiabilidad de los datos al mismo tiempo que presenta la información en tiempo real.

Una vez evaluadas las diferentes herramientas de modelado disponibles en el mercado se optó por la utilización del modelamiento UML, en razón al respaldo que esta herramienta ofrece, pues en la industria es casi un estándar único en este sentido, con lo anterior se asegura el escalamiento del software de forma fácil en periodos posteriores, pues los modelos de casos de uso presentados son entendibles en su totalidad y permiten su rápido análisis e implementación.

El prototipo fue implementado una vez se han aplicado diferentes pruebas funcionales que permiten establecer que realiza sus funcionalidades primordiales (crear, modificar, consultar y eliminar) a las entidades del proceso de inventario de forma eficiente y efectiva.

Referencias

- ABP Soluciones S.A.S. (2014). *ORION plus*. Obtenido de ORION plus:
http://www.orionplus.info/WEB/#info_orion
- ALEGRA. (2019). *ALEGRA*. Obtenido de <https://www.alegra.com>:
<https://www.alegra.com/colombia/>
- Becerra , N. (05 de 2011). *Gestion RRHH USM*. Obtenido de <http://gestionrrhhusm.blogspot.com>:
<http://gestionrrhhusm.blogspot.com/2011/05/modelo-rup-rational-unified-process-o.html>
- Castrillon, G., & Zapata, J. (30 de 01 de 2016). *Gerencie*. Obtenido de Gerencie:
<https://www.gerencie.com/que-debemos-saber-de-inventarios-seccion-13-niif-para-pymes.html>
- Comunicaciones. (20 de junio de 2019). *globalbit*. Obtenido de
<http://www.globalbit.co/2019/07/20/el-alcance-del-software-en-el-mundo-actual-y-su-impacto-en-el-futuro/>
- Consensus SAP. (12 de 2019). *Consensus*. Obtenido de <https://consensussap.com/>:
<https://consensussap.com/>
- DEBITOR. (2007). *DEBITOR*. Obtenido de DEBITOR SAS:
<https://debitoor.es/glosario/definicion-gestion-de-inventarios>
- DTyOC. (06 de 07 de 2016). *De Tecnología y Otras Cosas*. Obtenido de <https://dtyoc.com>:
<https://dtyoc.com/2016/06/07/modelo-rup-ibm/>
- Educa Play. (11 de 2018). *EDUCA PLAY*. Obtenido de <https://www.educaplay.com>:
https://www.educaplay.com/printablegame/2055227-metodologia_rup.html
- Expert, G. (19 de mayo de 2020). *GestioPolis.com*. Obtenido de <https://www.gestiopolis.com/que-es-inventario-tipos-utilidad-contabilizacion-y-valuacion/#autores>
- Fuel Control Solutions. (11 de 2019). *Fuel Control Solutions*. Obtenido de
<https://www.fuelcontrol.com.co/>: <https://www.fuelcontrol.com.co/>
- Garcia, R. (10 de 10 de 2019). *Tus Ejemplos*. Obtenido de <https://tusejemplos.com>:
<https://tusejemplos.com/ejemplos-de-kardex/>
- Giovanny, G. (11 de junio de 2001). *GestioPolis.com*. Obtenido de
<https://www.gestiopolis.com/sistemas-de-valuacion-de-inventarios/>
- Helisa NIIF. (15 de 01 de 2020). *Helisa NIIF*. Obtenido de <https://helisa.com/>: <https://helisa.com/>

- LOGGRO.* (12 de 05 de 2019). Obtenido de https://www.loggro.com:https://www.loggro.com/?GA_network=g&GA_device=c&GA_campaign=810954660&GA_adgroup=109091126318&GA_target=&GA_placement=&GA_creative=461115519304&GA_extension=&GA_keyword=&GA_loc_physical_ms=1003659&gclid=CjwKCAiA8Jf-BRB-EiwAWDtEGrxvGLjEprffIHjcDvaOZxvpUc5
- LogiMov.* (06 de 2019). *LogiMov.* Obtenido de <https://www.logimov.com/:https://www.logimov.com>
- López Roschiano, R., & Pech Montejó, J. (07 de 2015). *Universidad Politécnica de Madrid.* Obtenido de <https://core.ac.uk:https://core.ac.uk/download/pdf/148684658.pdf>
- López, A. (12 de 2015). *Universidad Tecnológica y pedagógica de Colombia.* Obtenido de <https://repositorio.uptc.edu.co:https://repositorio.uptc.edu.co/bitstream/001/1477/2/TGT-225.pdf>
- MAQALQ. (06 de 06 de 2017). *Sites Google.* Obtenido de https://sites.google.com:https://sites.google.com/a/utecnologica.edu.bo/maqalq_rup/matricula/matricula-1
- Mendez, P. (05 de 12 de 2017). *Patricia Mendez.* Obtenido de <https://patriciamendez.es.tl:https://patriciamendez.es.tl/Kardex.htm>
- Metodoss. (12 de 2018). *metodoss.com.* Obtenido de [metodologia RUP:https://metodoss.com/metodologia-rup/](https://metodoss.com/metodologia-rup/)
- OFIMA S.A.S Líderes en Software Empresarial. (2016). *OFIMA S.A.S.* Obtenido de OFIMA S.A.S: http://info.ofima.com/software-inventarios-empresas-ofima?gclid=EAIaIQobChMIhqPM_tiu1gIVSFmGCh3S8gb9EAAYASAAEgLBxvD_BwE.
- POS Colombia.* (12 de 2018). Obtenido de https://vendty.com/software-pos:https://vendty.com/software-pos/?utm_campaign=Website+traffic-Search-35&utm_source=adwords&utm_term=software%20pos&utm_medium=ppc&hsa_ad=372167193722&hsa_grp=75373457926&hsa_kw=software%20pos&hsa_ver=3&hsa_net=adwords&hsa_acc=9929922357&hsa_src=g&hsa_cam=
- RUPAN. (19 de 01 de 2017). <http://rupandcmmi.blogspot.com>. Obtenido de Metodología RUP y metodología CMMI: <http://rupandcmmi.blogspot.com/p/fases-de-lametodologia-rup-fase-de.html>

- SIIGO. (12 de 11 de 2018). *SIIGO Software Contable*. Obtenido de <https://www.siigo.com/>:
<https://www.siigo.com/>
- Softland Go. (12 de 01 de 2020). *Softland Go*. Obtenido de <https://softland.com.co>:
https://softland.com.co/softland-go/?gclid=CjwKCAiA8Jf-BRB-EiwAWDtEGt9m-Z4BoXYyEyyNFcbyzS9yjLK7axYusEW2L2TH994M4wOmNQPF9hoC33cQAvD_BwE
- TNS Software. (10 de 2019). *TNS Software*. Obtenido de <https://www.tns-software.co/>:
<https://www.tns-software.co/>
- Universidad Nacional de Colombia. (Julio de 2014). *Revista Innovar Journal*. Obtenido de Ciencias Administrativas y Sociales:
<https://revistas.unal.edu.co/index.php/innovar/article/view/51548/51894>
- Vergelin, M. J. (25 de 01 de 2017). *CONTAGRAM*. Obtenido de CONTAGRAM:
<https://contagram.com/blog/control-stock-administrar-inventario-pyme/>
- Xubio SAS. (01 de 02 de 2020). *XUBIO*. Obtenido de <https://xubio.com>:
<http://www.fce.unal.edu.co/unidad-de-informatica/proyectos-de-estudio/contaduria/2150-xubio-solucion-contable-para-pymes.html#:~:text=Xubio%20es%20una%20soluci%C3%B3n%20para,realiza%20com%C3%BAmente%20sus%20actividades%20normales>.