

# **UNIVERSIDAD PÚBLICA DE EL ALTO**

## **CARRERA DE INGENIERÍA DE SISTEMAS**



## **PROYECTO DE GRADO**

### **SISTEMA WEB DE GESTIÓN DE INVENTARIOS PARA ALMACENES**

### **CASO: CAJA NACIONAL DE SALUD REGIONAL LA PAZ – SECCIÓN DE EQUIPOS ELECTRÓNICOS**

**Para Optar al Título de Licenciatura en Ingeniería de Sistemas**

**Mención: INFORMÁTICA Y COMUNICACIONES**

**Postulante: Juan Carlos Choque Mamani**

**Tutor Metodológico: Ing. Marisol Arguedas Balladares**

**Tutor Especialista: Lic. Mario Torrez Cupiticona**

**Tutor Revisor: Lic. Beatriz Colque Condori**

**EL ALTO – BOLIVIA**

**2020**

## **Dedicatoria**

A Dios por todos las bendiciones que me dio el cual agradezco infinitamente.

A mi madre a la persona que más quiero en la vida, Natividad Mamani Mamani quien con su apoyo y comprensión me dio la oportunidad de crecer profesionalmente.

Gracias, de todo corazón te quiero mucho.

## **Agradecimientos**

Agradezco a Dios por guiar todos mis pasos y darme la fuerza necesaria para seguir adelante

A mi familia por el apoyo incondicional que me brindaron siempre. A mi madre Natividad y mis hermanas Anahi, Katherin, Yoselin y Luz quienes me apoyaron en momentos difíciles y por confiar en mí.

A mis tutores Lic. Mario Torrez Cupiticona, Lic. Beatriz Colque Condori e Ing. Marisol Argedas Ballarades, por brindarme su tiempo, comprensión y paciencia al realizar el seguimiento y culminación de este presente proyecto de grado. Por brindarme las valiosas sugerencias para este proyecto el cual me ayudara mucho en vida personal y profesional.

Al Jefe de Unidad de Sistemas Regional La Paz Ing. Christian Maldonado Titirico por otorgarme la oportunidad de desarrollar el sistema e implementarlo en su unidad.

Agradecer a todo el personal de la Unidad de Sistemas Regional La Paz por la orientación, colaboración y confianza depositada en mi persona.

A mis compañeros de la universidad por brindarme la ayuda, amistad y el compañerismo necesario para esos momentos difíciles.

A todos los docentes de la carrera por la ayuda la guía y formación académica.

De todo corazón, muchísimas gracias.

## RESUMEN

El presente proyecto de grado pretende automatizar los procesos más importantes que tiene la Caja Nacional de Salud Regional sección equipos electrónicos realizando el ingreso, salida y bajas de equipos electrónicos ya que esos procesos son vital importancia para la institución y es necesario controlar la información que se genera día a día además de centralizar y gestionar la información de equipos, usuarios, proveedores, centros, ingresos y salidas así para asegurar los niveles de existencia y disponibilidad de equipos electrónicos en caso de que un funcionario de la caja lo requiera para desempeñar mejor su labor hacia sus asegurados. El proyecto de grado titulado “**SISTEMA WEB DE GESTIÓN DE INVENTARIOS PARA ALMACENES CASO: CAJA NACIONAL DE SALUD REGIONAL LA PAZ – SECCIÓN DE EQUIPOS ELECTRÓNICOS**” se ha desarrollado con el objetivo de optimizar los procesos de gestión de información de equipos electrónicos asegurando la disponibilidad de pedidos sin dejar que falte o exceda los niveles de existencia.

Para el desarrollo del proyecto se utilizó la metodología ágil de gestión de proyecto XP y SCRUM, que propone un modelo de proceso incremental, basado en iteraciones y revisiones continuas con el usuario. Lo cual se utilizó en cada una de las cinco iteraciones, la metodología UWE, que se especializa en el diseño de las aplicaciones WEB.

En conclusión del desarrollo del sistema WEB se utilizó como herramienta el Framework LARAVEL, VUEJS, para el gestor de base de datos se usó MariaDB y con la ayuda del servidor APACHE para la función correcta del sistema.

## ÍNDICE GENERAL

MARCO PRELIMINAR.....	1
1.1. INTRODUCCIÓN.....	1
1.2. ANTECEDENTES.....	2
1.2.1. Antecedentes Institucionales.....	2
1.2.1.1. Misión.....	3
1.2.1.2. Visión.....	3
1.2.1.3. Objetivo.....	3
1.2.2. Antecedentes Internacionales.....	3
1.2.3. Antecedentes Nacionales.....	4
1.3. PLANTEAMIENTO DEL PROBLEMA.....	5
1.3.1. Problema Principal.....	5
1.3.2. Problemas secundarios.....	6
1.4. OBJETIVOS.....	7
1.4.1. Objetivo General.....	7
1.4.2. Objetivos Específicos.....	7
1.5. JUSTIFICACIONES.....	7
1.5.1. Justificación Técnica.....	7
1.5.2. Justificación Económica.....	8
1.5.3. Justificación Social.....	8
1.6. Metodología.....	9
1.7. HERRAMIENTAS.....	9
1.8. LÍMITES Y ALCANCES.....	10
1.8.1. Límites.....	10
1.8.2. Alcances.....	11
1.9. APORTES.....	12
1.9.1. Aporte teórico.....	12
1.9.2. Aporte práctico.....	12
MARCO TEÓRICO.....	13
2.1. INTRODUCCIÓN.....	13
2.2. INVENTARIOS Y CONTROL DE INVENTARIOS.....	13
2.2.1. Tipos de inventarios.....	14
2.2.2. Criterios de evaluación de inventarios.....	14
2.2.2.1. Primera Entradas, Primeras Salidas(FIFO).....	14
2.2.2.2. Ultimas Entradas, Primeras Salidas(LIFO).....	15

2.2.2.3. Primero Caducar, Primeras Salidas(FEFO).....	15
2.3. CONCEPTOS BÁSICOS .....	15
2.3.1. Almacén .....	15
2.3.2. Definición de almacenista.....	16
2.3.3. Función de los almacenes .....	16
2.3.4. Función de las existencias.....	16
2.3.5. Diseño de almacenes .....	17
2.3.6. Movimientos de almacenes .....	17
2.3.7. Seguridad en almacenes .....	18
2.3.8. Almacenamiento de materiales.....	19
2.4. METODOLOGÍA PARA EL DESARROLLO DEL SOFTWARE .....	19
2.4.1. Metodología XP .....	19
2.4.1.1. Comunicación.....	19
2.4.1.2. Simplicidad .....	20
2.4.1.3. Retroalimentación.....	20
2.4.1.4. Respeto .....	20
2.4.1.5. Coraje.....	20
2.4.1.6. Roles XP .....	20
2.4.2. Metodología SCRUM.....	21
2.4.2.1. Características de SCRUM.....	22
2.4.2.2. Ciclo de vida.....	23
2.4.2.3. Roles de SCRUM .....	26
2.4.3. Metodología UWE .....	29
2.4.3.1. Fases de la metodología UWE .....	30
2.5. INGENIERÍA WEB.....	31
2.5.1. Sistema WEB .....	33
2.6. HERRAMIENTAS .....	33
2.6.1. Servidor WEB apache .....	33
2.6.2. Framework Laravel.....	34
2.6.3. Framework Vuejs.....	34
2.6.4. Base de datos MariaDB.....	34
2.7. CALIDAD DEL SOFTWARE ISO 9126 .....	34
2.7.1. Confiabilidad.....	35
2.7.2. Facilidad de Mantenimiento .....	36
2.7.3. Usabilidad.....	36
2.7.5. Portabilidad .....	36
2.8. SEGURIDAD DE SOFTWARE .....	37
2.8.1. Autenticación.....	37

2.8.2. Autorización.....	37
2.8.3. Cifrado.....	37
2.8.4. Encriptación.....	37
2.8.5. Seguridad de base de datos.....	38
2.9. PRUEBAS DE CALIDAD NORMA ISO/IEC 25010 .....	38
2.10. ESTUDIO DE COSTOS DEL SISTEMA WEB.....	39
2.10.1. Modelo Cocomo II .....	39
MARCO APLICATIVO.....	42
3.1. INTRODUCCIÓN.....	42
3.2. COMBINANDO SCRUM CON XP .....	43
3.3. PRE-GAME .....	43
3.3.1. Recopilación de requerimientos.....	43
3.3.2. Especificación de requerimientos .....	43
3.3.3. Análisis de Riesgos .....	45
3.4. GAME .....	46
3.4.1. Primera Iteración .....	46
3.4.2. Segunda Iteración .....	48
3.4.3. Tercera Iteración .....	49
3.4.4. Cuarta Iteración .....	50
3.5. MODELADO DEL SISTEMA.....	51
3.5.1. Diagrama de casos de uso .....	51
3.5.1.1. Diagrama de Caso de Uso General.....	52
3.5.2. Descripción de caso de uso.....	52
3.5.2.1. Diagrama de Caso de Uso: Registro de usuario.....	53
3.5.2.2. Diagrama de caso de uso: Administración de equipos electrónicos.....	53
3.5.2.3. Diagrama de caso de uso: Ingreso de equipos electrónicos .....	54
3.5.2.4. Diagrama de caso de uso: Salida de equipos electrónicos .....	55
3.5.2.5. Diagrama de caso de uso: Baja de equipos electrónicos.....	56
3.5.2.6. Descripción de casos de usos .....	57
3.5.2.7. Diagrama de Actividades.....	62
3.5.2.8. Diagrama de Clases .....	66
3.5.2.9. Modelo Físico .....	68
3.5.2.10. Diagrama de Presentación .....	69
3.6. POST-JUEGO .....	73
3.6.1. Seguridad de sistema .....	73
3.6.1.1. Políticas de seguridad en cuanto a usuarios .....	73

3.6.1.2. Políticas de seguridad en cuanto a acceso al sistema .....	73
3.6.1.3. Políticas en cuanto al sistema operativo .....	74
3.6.1.4. Políticas de seguridad en cuanto a backups .....	74
3.6.2. Implantación .....	74
3.7. MODELO DE IMPLEMENTACIÓN DEL PROGRAMA .....	75
3.7.1. Diseño de interfaces .....	75
3.8. PRUEBAS DE CALIDAD ISO 25010 .....	81
3.8.1. Pruebas de http .....	81
3.8.2. Pruebas del navegador .....	82
3.9. METRICAS DE CALIDAD .....	83
3.9.1. Técnica Web ISO 9126 .....	83
3.9.1.1. Usabilidad .....	84
3.9.1.2. Mantenibilidad .....	85
3.9.1.3. Funcionabilidad .....	86
3.9.1.4. Confiabilidad .....	90
3.9.1.5. Portabilidad .....	91
3.9.2. Resultados .....	92
3.10. SEGURIDAD DE SOFTWARE .....	92
3.10.1. Autenticación .....	93
3.10.2. Autorización .....	94
3.10.3. Hash 95	
3.11. EVALUACIÓN DE COSTOS .....	95
3.11.1. Análisis de costos .....	95
3.11.2. Características de hardware requeridas .....	98
3.12. ANÁLISIS DE BENEFICIOS .....	98
CONCLUSIONES Y RECOMENDACIONES .....	99
4.1. CONCLUSIONES .....	99
4.2. RECOMENDACIONES .....	100
BIBLIOGRAFÍA .....	101
REFERENCIAS WEB .....	103
ANEXOS .....	104



## ÍNDICE DE FIGURAS

Fig. 1 Estructura de desarrollo ágil.....	22
Fig. 2 Estructura central de SCRUM .....	23
Fig. 3 Ciclo de SCRUM.....	24
Fig. 4 Seguimiento de Sprint.....	25
Fig. 5 Roles de SCRUM.....	28
Fig. 6 Planificación.....	42
Fig. 7 Diagrama de casos de uso general.....	52
Fig. 8 Diagrama de casos de uso registro de usuario en el sistema.....	53
Fig. 9 Diagrama de casos de administración de equipos electrónicos.....	54
Fig. 10 Diagrama de casos de ingreso de equipos electrónicos.....	55
Fig. 11 Diagrama de casos de salida de equipos electrónicos.....	56
Fig. 12 Diagrama de casos de bajas de equipos electrónicos.....	57
Fig. 13 Diagrama de actividades registro de usuarios en el sistema.....	62
Fig. 14 Diagrama de actividades de administración de equipos electrónicos.....	63
Fig. 15 Diagrama de actividades de ingresos de equipos electrónicos.....	64
Fig. 16 Diagrama de actividades de salidas de equipos electrónicos.....	65
Fig. 17 Diagrama de actividades de bajas de equipos electrónicos.....	66
Fig. 18 Diagrama de clases.....	67
Fig. 19 Modelo físico.....	68
Fig. 20 Diagrama de presentación Ingreso al Sistema.....	69
Fig. 21 Diagrama de presentación de administración.....	70
Fig. 22 Diagrama de presentación de registro de usuario.....	70
Fig. 23 Diagrama de presentación de equipos electrónicos.....	71
Fig. 24 Diagrama de presentación ingresos de equipos electrónicos.....	71
Fig. 25 Diagrama de presentación salida de equipos electrónicos.....	72
Fig. 26 Diagrama de presentación bajas de equipos electrónicos.....	72
Fig. 27 Ingreso al sistema.....	75
Fig. 28 Módulo de administración.....	76
Fig. 29 Registro de artículos.....	76
Fig. 30 Impresión de inventarios generales.....	77

Fig. 31 Registro de ingresos de equipos electrónicos .....	77
Fig. 32 Acta de ingreso de equipos electrónicos .....	78
Fig. 33 Detalle de Ingresos de equipos electrónicos .....	78
Fig. 34 Registro de salida de equipos electrónicos .....	79
Fig. 35 Acta de entrega de equipos electrónicos .....	79
Fig. 36 Modal de selección de artículos de salida .....	80
Fig. 37 Detalle de salida de equipos electrónicos.....	80
Fig. 38 Registro de baja de equipos electrónicos .....	81
Fig. 39 Pruebas de JSON en laravel .....	82
Fig. 40 Código de autenticación o guardias .....	93
Fig. 41 Código de autenticación o proveedores .....	93
Fig. 42 Tabla de base de datos de personas .....	93
Fig. 43 Tabla de base de datos de usuarios.....	94
Fig. 44 Código de autenticación .....	94
Fig. 45 Código de Hash .....	95

## ÍNDICE DE TABLAS

Tabla 1: Enfoques de las pruebas .....	39
Tabla 2: Requerimiento .....	45
Tabla 3: Análisis de riesgo .....	46
Tabla 4: Primera Iteración .....	47
Tabla 5: Segunda Iteración .....	48
Tabla 6: Tercera Iteración .....	49
Tabla 7: Cuarta Iteración.....	51
Tabla 8: Descripción de actores .....	51
Tabla 9: Registro de usuario .....	58
Tabla 10: Administración de equipos electrónicos.....	59
Tabla 11: Ingresos de equipos electrónicos .....	60
Tabla 12: Salida de equipos electrónicos .....	61
Tabla 13: Salida de equipos electrónicos .....	62
Tabla 14. Encuestas de usabilidad del sistema .....	85
Tabla 15. Entradas de funcionalidad según punto función .....	87
Tabla 16. Cuenta total con factor de ponderación medio .....	88
Tabla 17. Ajuste de complejidad del punto fusión .....	89
Tabla 18. Ajuste de complejidad del punto fusión .....	89
Tabla 19. Resultado de evaluación de calidad .....	92
Tabla 20. Conversión de puntos de función para COCOMO II .....	96
Tabla 21. Relación de valores en el modelo COCOMO .....	97

# CAPÍTULO I

## MARCO PRELIMINAR

### 1.1. INTRODUCCIÓN

La evolución de las nuevas tendencias tecnológicas asociadas a la información ha cambiado la vida cotidiana de la humanidad simplificando tareas y resolviendo problemas.

La tecnología responde al deseo y la voluntad que tenemos las personas de transformar nuestro entorno, transformar el mundo que nos rodea buscando nuevas y mejores formas de satisfacer nuestros deseos.

En la actualidad las instituciones públicas y privadas necesitan nuevas tecnologías de información para mantenerse en un mercado cada vez más exigente y competitivo, debiendo adaptarse a las necesidades y requerimientos de sus consumidores, por esta razón deben tener en cuenta ciertos criterios que fortalezcan su competitividad, tales como su organización interna, administrativa y control de sus activos, calidad de sus productos, servicios y atención a los usuarios, criterios que no siempre se toman en cuenta o no se les presta atención requerida.

La administración de inventarios permite que las instituciones logren maximizar el servicio ofrecido en almacenes y minimizar el costo de mantener su inventario, por ello se debe establecer un equilibrio de los inventarios, ya que si los artículos ingresan en grandes cantidades, su control en inventarios se dificultaría y estarían expuestos al deterioro. Por otra parte al mantener cantidades pequeñas puede generar problemas de stock para abastecer la demanda de los usuarios. La gestión de inventarios busca reducir al mínimo posible los niveles de existencia y asegurar la disponibilidad de existencia en el momento justo.

En el presente proyecto titulado “Sistema web de Gestión de Inventarios para almacenes”, caso Caja Nacional de Salud Regional La Paz – Sección Equipos Electrónicos de la Unidad de Sistemas, se desarrollará los procesos más importantes

para la administración tanto como abastecimiento, salidas y bajas de equipos electrónicos donde el sistema brindará un soporte eficaz e inmediato en los procedimientos de información y gestión de inventarios en almacenes.

Para el desarrollo del presente proyecto se utilizará las metodologías ágiles XP y SCRUM combinados, para la gestión del sistema principalmente por su agilidad en el desarrollo de aplicaciones en periodos muy breves de tiempo, iteraciones de análisis, trabajo en equipo y el diseño se apoyará con la metodología UWE, donde se verificará el método de inventario de provisiones y su criterio de evaluación primero en caducar primero en salir para los artículos almacenados. El desarrollo del presente proyecto se usará los lenguajes de programación Laravel Mix, para el resguardo de información en almacenes se usará mariaDB, se evaluará costos, tiempos y métricas de calidad la ISO 9126.

## **1.2. ANTECEDENTES**

### **1.2.1. Antecedentes Institucionales**

La Caja Nacional de Salud en el año 1987 esta institución cambia de nombre de Caja Nacional de Seguridad Social C.N.S.S. a Caja Nacional de Salud C.N.S., el Ministerio de Salud y Deportes junto al Instituto Nacional de Seguros de Salud INASES en Resolución Administrativa aprueba el Estatuto Orgánico de la C.N.S., que actualmente se rige bajo el Código de Seguridad Social y los dictámenes emanados por las autoridades en el transcurso de los años.

La Caja Nacional de Salud, institución descentralizada de derecho público sin fines de lucro, encargada de la gestión, aplicación y ejecución del Régimen de Seguridad Social a corto plazo como ser: Enfermedad, Maternidad, Riesgos Profesionales y Asignaciones Familiares que comprenden los subsidios, natalidad, lactancia y sepelio.

Se tiene el Seguro de Trabajador Dependiente, Rentistas, Seguro de Salud para el Adulto Mayor (SSPAM), Seguro Voluntario, Seguro del Abogado, Niños huérfanos, Niños especiales, Instituto de ceguera, D.S. 20989, Seguro para excombatientes y viudas; la institución tiene la responsabilidad de atender la salud de sus asegurados y

beneficiarios; Esta actividad se realiza con el adecuado conocimiento de sus beneficiarios sobre sus derechos y obligaciones para poder acceder a los derechos de la Seguridad Social.

La C.N.S. (Caja Nacional de Salud) para el logro de sus objetivos, tiene una organización administrativa desconcentrada geográficamente en Administraciones Regionales y Distritales; encargada de la gestión aplicación y ejecución del régimen de Seguridad Social a Corto Plazo: Enfermedad, Maternidad y Riesgos Profesionales, instituidos por el Código de Seguridad Social, su Reglamento, la Ley Financial 924, el Decreto Supremo 21637 y demás disposiciones legales conexas.

#### **1.2.1.1. Misión**

Brindar protección integral de salud en los regímenes de enfermedad, maternidad y riesgos profesionales a la población asegurada, bajo los principios de universalidad, solidaridad, unidad de gestión, economía, optimizando los recursos y ampliando la cobertura.

#### **1.2.1.2. Visión**

Ser líder nacional en la provisión de servicios integrales de salud, con oportunidad, calidad y calidez.

#### **1.2.1.3. Objetivo**

El objetivo principal de administración Regional La Paz es brindar de forma oportuna eficaz y eficiente la presentación de servicios de salud de calidad y calidez a toda su población protegida, optimizando el uso de recursos e insumos buscando ampliar el nivel de cobertura.

### **1.2.2. Antecedentes Internacionales**

- Proyecto de tesis, Diseño de un Sistema de Control de Inventario y Organización de las bodegas de producto terminado de la empresa Ecuaspumas - Lamitex S.A desarrollado por Diego Fernando Quizhpi Campoverde año 2018 de la Universidad

Politécnica Salesiana Sede Cuenca del Ecuador, objetivo de investigación es Diseñar un Sistema de Control de Inventario y Organización de la Bodegas de Producto Terminado de la Empresa ECUAESPUMAS - LAMITEX S.A. Este proyecto apoya con la rentabilidad de la compañía, disminución de ventas perdidas, entrega oportuna de productos y responder ante imprevistos de la oferta y la demanda de activos. [Quizhpi, 2018].

- Tesis, Sistema de gestión de almacenes con identificación automática de captura de datos para un control eficiente del flujo de procesos, diseñado por Miguel Ángel Hernández Trinidad año 2017 de la Instituto Politécnico Nacional de México, con el objetivo de implementar una serie de aplicaciones cliente-servidor, que permitan el control y administración de las operaciones relacionadas con los almacenes de servicio técnico y materia prima de equipos online. Utilizando la metodología de Microsoft Solutions Framework. Por lo cual se desarrolla una plataforma web para que su administración de productos sea confiable donde se relacionan con bodegas de materiales o productos. [Hernández, 2017]

### **1.2.3. Antecedentes Nacionales**

- Proyecto de Grado, Sistema de control de ventas e inventarios para almacenes de aluminios utilizando dispositivos móviles desarrollado por Grover Gutiérrez Vargas año 2015 de la Universidad Mayor de San Andrés, implementado en Técnica de Aluminio, Vidrio y Servicios para TALVISER, su objetivo de investigación es desarrollar un sistema de control de ventas e inventarios para almacenes, utilizando la metodología ágil SCRUM. Donde satisface con los objetivos que quiso llegar alcanzando a cumplir metas automatizando toda la información de activos en almacenes donde se incorpora botones e interfaces de stock e informes. [Gutiérrez, 2015]
- Proyecto de Grado, Sistema de administración y control de almacén e inventario Gobierno Municipal La Paz, desarrollado por Miriam Olga Ortega Mamani el año 2007 de la Universidad Mayor de San Andrés, implementado en Mallasa, objetivo de investigación es Implementar un Sistema de administración y control de almacén que permitirá mejorar el registro de ingreso, salida y existencia de los

materiales en almacenes, utilizando la metodología ágil RUP y UML. Donde la documentación indica que el sistema permitirá cubrir las necesidades para el mejoramiento y seguimiento de entradas y salidas de materiales, haciendo que emita resultados más completos. [Ortega, 2007]

- Proyecto de Grado, Sistema de inventario y control de almacenes desarrollado por Marco Antonio Vicente Magne en el año 2011 de la Universidad Mayor de San Andrés, implementado en AASANA, objetivo de investigación es Implementar el sistema de control de almacenes centralizado para el departamento de almacenes de AASANA, utilizando la metodología ágil RUP, indicando que este sistema permitirá realizar el control para el departamento de almacenes que facilite en manejo de la información rutinario donde esta herramienta viabilizará el fluido de manera rápida y confiable de la información entre los departamentos. [Vicente, 2011]

### **1.3. PLANTEAMIENTO DEL PROBLEMA**

La Caja Nacional de Salud Regional La Paz se gestiona una gran cantidad de información que se genera en el ingreso, entregas, reportes y bajas de equipos electrónicos, cuya información se registra de forma manual, implicando en muchos casos, pérdida de información, pérdida de tiempo en la búsqueda de información, mala actualización de niveles de existencia, mala actualización de disponibilidad de existencia y mala manipulación de datos, el cual tiene como consecuencia problemas de abastecimiento, problemas de información de equipos electrónicos, pérdidas económicas en la depreciación de equipos y como resultado perjudica el desenvolvimiento de las secciones y unidades que tiene la institución y que gozan del servicio de almacenes y por ende la salud de los asegurados.

#### **1.3.1. Problema Principal**

La Caja Nacional de Salud Regional La Paz – Sección de equipos electrónicos, trabaja de manera multidisciplinaria donde participan administradores, almaceneros, personal de diferentes unidades cuyas tareas generales se sintetizan en pedidos de equipos electrónicos requeridos para el buen desempeño de los funcionarios de diferentes



unidades y la salud de todos sus asegurados. La información respectiva del abastecimiento, niveles de existencia, disponibilidad de existencias y entregas se registra de forma manual. Eso causa que no haya un control adecuado de almacenes e inventarios donde se tiene el riesgo de no controlar los niveles de existencia y no asegurar la disponibilidad de existencias en el momento justo.

En este contexto la pregunta que se plantea como problema es la siguiente:

¿De qué forma el sistema web de gestión de inventarios para almacenes sección equipos electrónicos puede coadyuvar en los procesos de gestión y control de equipos electrónicos en la Caja Nacional de Salud Regional La Paz?

### **1.3.2. Problemas secundarios**

- El registro de ingreso de equipos electrónicos es manual, por tanto es moroso registrar en los libros de inventarios la información actual de equipos electrónicos o caso contrario ya no se lo registra perdiendo así los niveles de existencia de equipos electrónicos.
- La actualización de equipos electrónicos es manual, generando problemas en los niveles de existencias actuales de abastecimiento, salida y bajas de equipos electrónicos.
- La información resguardada de documentación de referencia es inadecuada, por pérdidas y desorden de documentación el cual generan problemas en la búsqueda de comprobantes de ventas, facturas y contratos de ciertos equipos electrónicos.
- La búsqueda de equipos electrónicos para la entrega es inadecuada, porque no hay orden de resguardo adecuado, causando demoras en la atención a los funcionarios y afectando así los niveles de existencia.
- La salida de equipos electrónicos es inadecuado, porque al sacar un equipo electrónico no ordenado puede causar que los primeros en caducar se queden obsoletos y se deprecien así afectando económicamente a la empresa.
- Demoras en la verificación de garantías de equipos electrónicos desperfectos, causando que el equipo este sin funcionamiento por mucho tiempo por la búsqueda

de documentos de compras afectando económicamente a la institución y la atención de sus asegurados.

- Retrasos en la elaboración de actas, afectando a los funcionarios de diferentes unidades en la entrega de equipos electrónicos.
- Demoras en la elaboración de inventarios que afecta al administrador de almacenes, en los reportes de estado de almacén.

## **1.4. OBJETIVOS**

### **1.4.1. Objetivo General**

Desarrollar un sistema web de gestión de inventarios en almacenes Sección Equipos Electrónicos para optimizar los procesos de gestión de información de equipos electrónicos de la Caja Nacional de Salud Regional La Paz, asegurando la disponibilidad de pedidos sin dejar que falte o exceda los niveles de existencia.

### **1.4.2. Objetivos Específicos**

- Realizar ingeniería de requerimientos o requisitos para identificar las necesidades del planteamiento del problema a satisfacer para el desarrollo del sistema por las partes interesadas.
- Analizar procesos actuales de la Institución, para identificar modelos y procesos que sirvan de soporte a los problemas a satisfacer.
- Diseñar las especificaciones de requerimientos para la estructura que debe darse al sistema web.
- Evaluar calidad de software usando ISO 9126 para evaluar el desarrollo y métricas de uso del software.

## **1.5. JUSTIFICACIONES**

### **1.5.1. Justificación Técnica**

El proyecto se justifica técnicamente, porque la Institución cuenta con las herramientas informáticas en hardware y software necesario para implementar y mantener el

sistema de gestión de inventarios para almacenes que se propone en el presente proyecto.

Asimismo, se justifica porque para el desarrollo del proyecto se utilizará herramientas tecnológicas de software de última generación, licencia libre y de código abierto.

### **1.5.2. Justificación Económica**

El proyecto se justifica económicamente porque el sistema permitirá a la institución brindar la gestión de inventarios adecuados, donde se usará el tipo de criterio de evaluación de inventarios primero en caducar, primero en salir así para evitar que los equipos electrónicos sean obsoletos y depreciados por temas de garantías., administrar y gestionar los equipos electrónicos para que los pedidos urgentes no falten así para no afectar la atención de sus asegurados. Donde se minimizará el costo, tiempo, sistematizando y gestionando la información de almacenes.

Para el desarrollo del presente proyecto se utilizará software libre de código abierto, publicada bajo la licencia MIT el cual evitará el pago de licencias que permitirá el ahorro de la institución.

### **1.5.3. Justificación Social**

El proyecto se justifica socialmente porque beneficiará al administrador del sistema, donde tendrá acceso a toda la información, gestión y administración de todo el personal de almacén donde tomará decisiones correctas y oportunas acerca de la buena administración del almacén.

Beneficiará a los almaceneros que estarán encargados de interactuar con el sistema, donde será una ayuda para la toma de decisiones, facilitando el uso de la información de una manera más rápida, eficiente y oportuna con lo que se podrá reducir los tiempos de entrega y los costos de operación.

Beneficiará a los funcionarios de diferentes secciones de la institución donde podrán tener información más clara y oportuna de pedidos de equipos electrónicos para que desempeñen su labor de manera más ágil.

Beneficiará a los asegurados, donde la atención que brindan los funcionarios de la institución sea ágil y satisfactoria.

## **1.6. Metodología**

Para la gestión del presente proyecto se utilizará la metodología ágil XP o programación extrema combinando con SCRUM para gestionar el desarrollo de proyectos y desarrollo del software, el cual estas se adaptan a las necesidades del proyecto, en poner principal énfasis en el producto, además de su flexibilidad y principalmente por su agilidad en el desarrollo de aplicaciones en periodos de tiempo con participación de los interesados.

Para el proceso de diseño del proyecto se utilizará la metodología UWE, para aplicaciones Web enfocado sobre el diseño sistemático, la personalización y la generación semiautomática de escenarios, describe una metodología de diseño sistemática, basada en las técnicas que posee UML.

El método de ingeniería de software comprende métodos y técnicas para desarrollar y mantener software de calidad que resuelven problemas de todo tipo.

Se utilizará el criterio de evaluación de inventario primero en caducar, primero en salir, para que los equipos electrónicos sean controlados por el tema de caducidad y depreciación de equipos.

Por lo tanto, la ingeniería de software o método de ingeniería, incluye el análisis previo de la situación, el diseño del proyecto, el desarrollo del software, las pruebas necesarias para confirmar su correcto funcionamiento y la implementación del sistema.

## **1.7. HERRAMIENTAS**

Para el desarrollo del sistema web de gestión de inventarios para almacenes sección equipos electrónicos caso el Caja Nacional de Salud Regional La Paz, se utilizará herramientas de programación de código abierto para su futuro mantenimiento y programación orientadas a la web para que cualquier usuario pueda acceder al

sistema en cualquier navegador de internet, ya que es más fácil alojar el dominio en un servidor o en la nube, para esto utilizaremos lo siguiente:

- Servidor web APACHE, es un software de código abierto, libre de uso y totalmente configurable, es en este momento el más utilizado en la red, ya sea en plataformas Linux o Windows. Un servidor web es un software instalado en el equipo con todas las condiciones necesarias para servir o entregar páginas web que le sean solicitadas por un navegador, asegurando que se muestren y representen todos los elementos necesarios para su correcto funcionamiento y visualización.
- Laravel es un framework PHP para el desarrollo rápido de aplicaciones web, es rápido, relativamente sencillo y capaz de trabajar en la mayoría de entornos de hosting, es un framework de código abierto para desarrollar aplicaciones y servicios web con PHP 5 y PHP 7. Su filosofía es desarrollar código PHP de forma elegante, simple y ordenada.
- Vue.js, es un framework de JavaScript popular para desarrollar aplicaciones modernas y escalables en el lado del cliente.
- MariaDB, es un gestor de base de datos MySQL. El propósito de este sistema es proveer capacidades similares y extendidas en relación a MySQL, con un poco especial en mantener el software de forma libre a través de la licencia GPL. Con aplicaciones muy amplias, puede ser usado por equipos de desarrollo de software, testing de aplicaciones, sitios web de empresas y bloggers, servidores vps o locales y de hosting compartido.

## **1.8. LÍMITES Y ALCANCES**

### **1.8.1. Límites**

El presente proyecto se desarrollará exclusivamente para la Unidad de Sistemas, Sección de Equipos Electrónicos para la Caja Nacional de Salud Regional La Paz.

El sistema está enfocado en la Gestión de inventarios para almacenes y su administración estará circunscrita solo al personal autorizado de la institución.

El sistema estará orientado solo en el registro de equipos electrónicos o material de red, sin embargo es posible ampliar a otras categorías realizando cambios en el software.

### **1.8.2. Alcances**

El Sistema web de gestión de inventarios para almacenes, la institución contará con los siguientes módulos:

- Módulo de administración, donde se automatizará y se administrará toda la información referente a los datos más comunes que interactúan con el sistema como ser: ingreso al sistema según su rol, personal de la institución, empresas, secciones y unidades de la institución.
- Módulo de artículos, donde se automatizará los niveles y disponibilidad de existencias para tener el control de información de inventarios actualizados en el momento de ingreso y salida de equipos electrónicos
- Módulo de registro de compras, donde se registrará toda la información de compras de equipos electrónicos y se automatizará el ingreso de equipos electrónicos para tener todos los datos registrados y detallados en el sistema de cada equipo en caso de que se requiera buscar un ingreso.
- Módulos de registro de salidas, donde se registrará toda la información de salida de equipos electrónicos para automatizar la información de salida y hacer que la búsqueda sea más fácil en caso de que se requiera buscar la información de un equipo electrónico.
- Módulos de registro de bajas, automatizar baja de un equipo electrónico para agilizar los trámites de castigo del equipo y cambiar por uno nuevo para el funcionario que lo requiera.
- Generar reportes finales, inventarios y actas para facilitar el trabajo que realiza el almacenero en el momento de ingreso, salida y baja de equipos electrónicos.
- Generación de estadísticas de compras y salidas de equipos electrónicos, para verificar las compras anuales totales y salidas de equipos electrónicos para la próxima inversión de compras de equipos electrónicos.

- Generar control de información de garantías para equipos electrónicos en mal estado para facilitar al almacenero en la búsqueda de ingresos y hacer el cambio oportuno de la misma.

## **1.9. APORTES**

### **1.9.1. Aporte teórico**

El presente proyecto tendrá aporte de información útil de diseño del software utilizando la herramienta web UWE, para el desarrollo, presentación, retroalimentación de software se usará la metodología XP y para el trabajo en equipo se usará la metodología ágil SCRUM.

### **1.9.2. Aporte práctico**

Para la institución, con la implementación del Sistema de gestión de inventarios para almacenes, sección de equipos electrónicos de la Caja Nacional de Salud Regional La Paz, la institución contará con una aplicación para facilitar el trabajo y desempeño en almacenes así evitando el retardo en procesos de búsqueda, entrada, salida y bajas de equipos electrónicos, automatizando la información y tener el control de los equipos electrónicos para la toma de decisiones.

El almacenero, contará con un nuevo sistema que le ayudará a simplificar tareas y redundancia de datos de información, ayudando de esta manera a que todos los equipos electrónicos que se tienen en los almacenes sean administrados de forma ágil y oportuna.

Los equipos electrónicos, la información estará automatizado para saber la cantidad total en almacenes para la toma de decisiones donde se contará con la agilización de información acerca de su garantía y datos importantes que se desee saber de las mismas.

Los funcionarios solicitantes, dispondrán de información oportuna acerca de pedidos de equipos electrónicos y así ayudar en la mejor atención hacia sus asegurados.

## **CAPÍTULO II**

### **MARCO TEÓRICO**

#### **2.1. INTRODUCCIÓN**

La base de toda empresa es la compra o venta de servicios, ahí la importancia del manejo de inventario por parte de la misma. Este manejo contable permite a la empresa mantener el control oportuno de los productos.

Los inventarios comprenden, además de las materias primas, productos en proceso y productos terminados o mercancías para la venta, bienes reparados para la prestación de servicios.

El inventario constituye las partidas del activo corriente que están listas para la venta, es decir, toda aquella mercancía que posee una empresa en el almacén valorada al costo de adquisición, para la venta o actividades productivas. Por medio del siguiente trabajo se darán a conocer algunos conceptos básicos de todo lo relacionado a los inventarios en una empresa, métodos, sistema y control.

#### **2.2. INVENTARIOS Y CONTROL DE INVENTARIOS**

El control de inventarios busca mantener disponible los productos que requieren para la empresa y para los clientes, por lo que implica la coordinación de las áreas de compras, manufactura distribución. [Zapata, 2014]

Los inventarios son acumulaciones de materias primas, provisiones, componentes, trabajo en proceso y productos terminados que aparecen en numerosos puntos a largo del canal de producción y de logística de una empresa [Ballou, 2005]

Lo mencionado hace referencia a que existen diferentes productos que deben estar disponibles, en las empresas, asegurar su funcionamiento, por lo tanto es importante realizar controles para cada uno de estos elementos o productos, según su clasificación.



### **2.2.1. Tipos de inventarios**

Los inventarios se clasifican de acuerdo al estado de los insumos requeridos en el proceso de producción las cuales son:

- **Inventarios de Materias Primas**, materiales utilizados para elaborar los componentes de productos determinados.
- **Inventarios de provisiones**, son todos los productos que la empresa obtiene a partir de sus proveedores y con los cuales se obtienen productos de mayor valor agregado para los clientes.
- **Inventarios de Componentes**, son todos aquellos elementos que hacen parte de alguna maquinaria, proceso o inmueble que se requiere para el correcto funcionamiento de la empresa.
- **Inventarios de Productos en proceso**, los que se efectúan un trabajo o que se encuentran esperando entre una operación y otra.
- **Inventarios de Productos terminados**, los que se encuentran listos para ser embarcados a un cliente de acuerdo a un pedido. [Zapata, 2014]

### **2.2.2. Criterios de evaluación de inventarios**

Existen varios métodos de evaluación de inventarios, los más usuales y los que dieron mayor resultado son los siguientes:

- Método de Primera Entradas, Primeras Salidas(FIFO)
- Método de Ultimas Entradas, Primeras Salidas(LIFO)
- Método de Primero Caducar, Primeras Salidas(FEFO)

#### **2.2.2.1. Primera Entradas, Primeras Salidas(FIFO)**

Se basa en las corrientes de las cantidades físicas, esto supone que los productos deben salir en el mismo orden en que fueron comprados, es decir, los primeros artículos comprados son los primeros que deben salir. El método es utilizado especialmente cuando los artículos se deterioran, pasan de moda o se vuelven inservibles.

### **2.2.2.2. Últimas Entradas, Primeras Salidas(LIFO)**

Supone que los materiales salgan en sentido contrario al que se reciben, es decir que los últimos artículos en entrar a almacenes son los primeros en salir. Bajo este supuesto el material existente en almacén al final de cada periodo pasaría hacer el primero en salir en la siguiente etapa.

### **2.2.2.3. Primero Caducar, Primeras Salidas(FEFO)**

Supone que los materiales con fechas de caducidad más cercanas tendrían que ser los primeros en salir.

## **2.3. CONCEPTOS BÁSICOS**

### **2.3.1. Almacén**

Es el sitio o lugar destinado a guardar, proteger, custodiar y despachar toda clase de materiales y/o artículos. La palabra Almacén proviene de Almagacen, vocablo árabe que significa "Tesoro", por tal significación se identifica el almacén y sus mercancías como un tesoro muy valioso, apreciado que se debe guardar, custodiar y cuidar para el futuro.

La formulación de una política de inventario para un departamento de almacén depende de la información respecto a tiempos de entrega, disponibilidades de materiales, tendencias en los precios y materiales de compras, son las mejores fuentes de información.

Esta función controla físicamente y mantiene todos los artículos inventariados, se deben establecer resguardo físicos adecuados para proteger los artículos de algún daño de uso innecesario debido a procedimientos de rotación de inventarios defectuosos y robos. Los registros de deben mantener, lo cual facilitan la localización inmediata de los artículos.

### **2.3.2. Definición de almacenista**

- Es aquella persona capaz de guardar, proteger, custodiar y despachar toda clase de materiales y/o artículos.
- Tiene que contar, medir y pesar la mercancía.
- Comparar el resultado del conteo con los documentos respectivos, a la hora cuando se despacha, cuando se recibe mercancía y observar la factura.
- Debe utilizar equipos móviles en las descarga (Seguridad Industrial).
- El Almacenista debe evitar la indisciplina.

### **2.3.3. Función de los almacenes**

- Mantienen las materias primas a cubierto de incendios, robos y deterioros.
- Permitir a las personas autorizadas el acceso a los materiales almacenados.
- Mantiene informado al departamento de compras, sobre las existencias reales de materia prima.
- Lleva en forma minuciosa controles sobre las materias primas, entradas y salidas
- Vigila que no se agoten los materiales, máximos y mínimos.

### **2.3.4. Función de las existencias**

- Garantiza el abastecimiento e invalida los efectos de:
  - Retraso en el abastecimiento de materiales.
  - Abastecimiento parcial
- Compra o producción en lotes económicos.
- Rapidez y eficacia en atención a las necesidades.

### **2.3.5. Diseño de almacenes**

El diseño de almacenes es una actividad especializada:

- El uso de planos arquitectónicos permite visualizar los almacenes, lo cual facilita su diseño.
- Los almacenes deben ser diseñados en términos de metros cúbicos y no de metros cuadrados, ya que el espacio vertical puede ser convenientemente utilizado de manera de aumentar al máximo el volumen de almacenamiento sin aumentar la superficie requerida.
- El almacén debe planificarse de manera que el ambiente de trabajo resulte agradable y se facilite el crecimiento futuro.
- La buena iluminación agiliza la localización de materiales, evita robos y reduce accidentes.
- La pérdida de los inventarios puede ser fatal para una empresa. Un buen sistema de detección y extinción de incendios protege los inventarios.

### **2.3.6. Movimientos de almacenes**

El movimiento de materiales es el proceso que estos siguen desde su llegada a los almacenes hasta su despacho.

- Los materiales deben tener localizaciones físicas específicas que permitan las actividades normales de almacén, como son la entrada y salida de inventario y la realización de inventarios físicos, para permitir un adecuado movimiento de materiales es aconsejable:
- Planificar, siempre que sea posible, un flujo de materiales en línea recta (recepción, almacenamiento, despacho).
- Separar las áreas de recepción y despacho de materiales.
- Considerar áreas separadas físicamente para materiales dañados, reparables o no, que esperan ser enviados a reparar o que acaban de ser reparados.
- Ubicar los materiales que requieren condiciones especiales de almacenamiento en áreas especialmente acondicionadas para tal fin.

- Diseñar los pasillos de circulación lo suficientemente anchos para permitir la circulación de montacargas y otros vehículos. Todos los pasillos secundarios deben fluir a un pasillo principal.
- Indicar claramente el sentido y las velocidades máximas de circulación en los pasillos, a fin de reducir el riesgo de accidentes. Un buen sistema de señalización es una excelente inversión.
- Establecer áreas especialmente protegidas para materiales valiosos.
- Construir las plataformas de descarga a la altura de los vehículos de transporte típicos.
- Las puertas de acceso y salida de los almacenes deben ser fácilmente manipulables por los operadores de los vehículos de movimiento de materiales.
- Disponer de suficiente espacio en el área de recepción de materiales para el control de calidad.

### **2.3.7. Seguridad en almacenes**

La seguridad en almacenes es un aspecto fundamental que permite trabajar con menos riesgo y más eficacia siempre y cuando se respeten unas reglas básicas de seguridad. Estas normas de seguridad deberían cumplirse siempre en el lugar de trabajo.

Algunas reglas simples que ayudan a reducir este problema son las siguientes:

- Diseñar con la seguridad en mente: asegurarse de que las áreas de almacenes estén cercadas o protegidas, de manera de minimizar la presencia de intrusos, iluminar adecuadamente las áreas de almacenes, no permitir la entrada de usuarios y empleados que no tengan que ver con el movimiento de materiales o inventario de los mismos al área de almacenes.
- Crear controles de acceso: contratar o conformar un cuerpo de vigilancia, verificar los vehículos que entren y salgan, realizar inspecciones no anunciadas de los casilleros, escritorios y vehículos del personal.
- Hacer inventarios periódicos o permanentes para verificar las pérdidas de material.

- Involucrar al personal en el control de los materiales: concienciarlo con charlas y concursos sobre el problema, aceptar sugerencias, informarles sobre los resultados de los inventarios, en lo que a pérdidas se refiere y sobre todo, hacerle sentirse parte importante de la organización.

Desafortunadamente es usual que sean los mismos empleados o aún los clientes quienes lleven a cabo el robo hormiga, otro factor que lleva al aumento de costos por falta de control del inventario. [BALLOU, 2014]

### **2.3.8. Almacenamiento de materiales**

Existen distintas posibilidades, según las características de los materiales que van a ser almacenados. La más empleada es la estantería, que puede ser obtenida en variedad de tamaños: cerradas, con entrepaños para la colocación de materiales de volumen intermedio, con vigas horizontales para paletas, con vigas inclinadas para elementos cilíndricos, solo con laterales para perfiles y elementos longitudinales, etc.

La ubicación física de los materiales en los almacenes debe ser establecida de manera que permita la localización rápida y sin errores de los materiales.

## **2.4. METODOLOGÍA PARA EL DESARROLLO DEL SOFTWARE**

### **2.4.1. Metodología XP**

La Programación Extrema, propone solo un conjunto de prácticas técnicas aplicadas de manera simultánea. Pretenden enfatizar los efectos positivos de un proyecto de desarrollo de software. Estos valores son:

#### **2.4.1.1. Comunicación**

En XP, todo es trabajado en equipo, desde el relevamiento y análisis hasta el código fuente desarrollado. Todo se conversa cara a cara, procurando hallar soluciones en conjunto a los problemas que puedan surgir.

#### **2.4.1.2. Simplicidad**

Se pretende desarrollar sólo lo necesario y no perder tiempo en detalles que no sean requeridos en el momento.

#### **2.4.1.3. Retroalimentación**

El objetivo de XP es entregar lo necesario al cliente, en el menor tiempo posible. A cambio, demanda al cliente, una retroalimentación continua, a fin de conocer sus requerimientos e implementar los cambios tan pronto como sea posible.

#### **2.4.1.4. Respeto**

El equipo respeta la idoneidad del cliente como tal y el cliente, a la vez, respeta la idoneidad del equipo.

#### **2.4.1.5. Coraje**

Se dice que en XP un equipo debe tener el valor para decir la verdad sobre el avance del proyecto y las estimaciones del mismo, planificando el éxito en vez de buscar excusas sobre los errores. [BAHIT, 2012]

#### **2.4.1.6. Roles XP**

**Programador:** El programador escribe las pruebas unitarias y produce el código del sistema. Debe existir una comunicación y coordinación adecuada entre los programadores y otros miembros del equipo.

**Cliente:** El cliente escribe las historias de usuario y las pruebas funcionales para validar su implementación. Además, asigna la prioridad a las historias de usuario y decide cuáles se implementan en cada iteración centrándose en aportar mayor valor al negocio. El cliente es sólo uno dentro del proyecto pero puede corresponder a un interlocutor que está representando a varias personas que se verán afectadas por el sistema.

**Encargado de pruebas (Tester):** El encargado de pruebas ayuda al cliente a escribir las pruebas funcionales. Ejecuta las pruebas regularmente, difunde los resultados en el equipo y es responsable de las herramientas de soporte para pruebas.

**Encargado de seguimiento (Tracker):** El encargado de seguimiento proporciona realimentación al equipo en el proceso XP. Su responsabilidad es verificar el grado de acierto entre las estimaciones realizadas y el tiempo real dedicado, comunicando los resultados para mejorar futuras estimaciones. También realiza el seguimiento del progreso de cada iteración y evalúa si los objetivos son alcanzables con las restricciones de tiempo y recursos presentes. Determina cuándo es necesario realizar algún cambio para lograr los objetivos de cada iteración.

**Entrenador (Coach):** Es responsable del proceso global. Es necesario que conozca a fondo el proceso XP para proveer guías a los miembros del equipo de forma que se apliquen las prácticas XP y se siga el proceso correctamente.

**Consultor:** Es un miembro externo del equipo con un conocimiento específico en algún tema necesario para el proyecto. Guía al equipo para resolver un problema específico.

**Gestor (Big boss):** Es el vínculo entre clientes y programadores, ayuda a que el equipo trabaje efectivamente creando las condiciones adecuadas. Su labor esencial es de coordinación. [LETERLIER Y PENADÉS, 2005]

#### **2.4.2. Metodología SCRUM**

SCRUM es una metodología ágil de gestión de proyectos cuyo objetivo primordial es elevar al máximo la productividad de un equipo. Reduce al máximo la burocracia y actividades no orientadas a producir software que funcione y produce resultados en periodos muy breves de tiempo.

Más bien delega completamente en el equipo la responsabilidad de decidir la mejor manera de trabajar para ser lo más productivos posible.



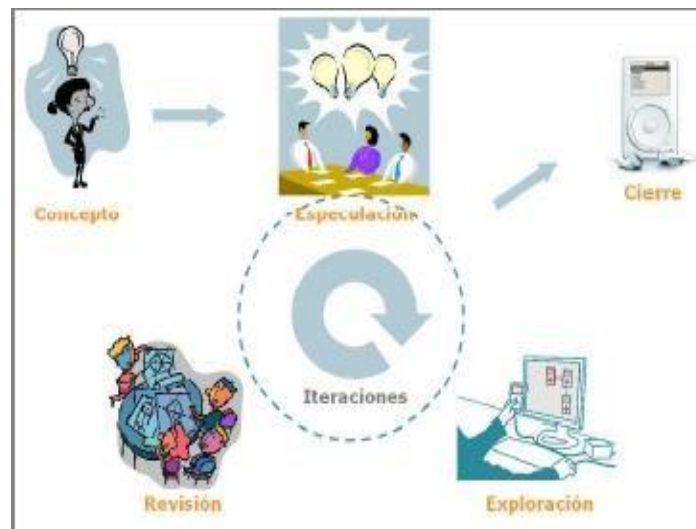
SCRUM es una metodología ágil, y como tal:

- Es un modo de desarrollo de carácter adaptable más que predictivo.
- Orientado a las personas más que a los procesos.
- Emplea la estructura de desarrollo ágil incremental basada en iteraciones y revisiones.

#### 2.4.2.1. Características de SCRUM

Sus principales características se pueden resumir en dos:

La primera comienza con la visión general del producto, especificando y dando detalle a las funcionalidades o partes que tienen mayor prioridad de desarrollo y que pueden llevarse a cabo en un periodo de tiempo breve normalmente de 30 días.



**Fig. 1** Estructura de desarrollo ágil

**Fuente:** [Palacio, 2008]

Cada uno de estos periodos de desarrollo es una iteración que finaliza con la producción de un incremento operativo del producto.

La segunda característica importante son las iteraciones que son la base del desarrollo ágil, y SCRUM gestiona su evolución a través de reuniones breves diarias en las que

todo el equipo revisa el trabajo realizado el día anterior y el previsto para el día siguiente. [PALACIO, 2008].



**Fig. 2** Estructura central de SCRUM

Fuente: [PALACIO, 2008]

#### 2.4.2.2. Ciclo de vida

Está compuesto de cuatro fases: planeación, arquitectura, desarrollo y cierre que se clasifica en esta metodología como: Pre-Game, Game y Post Game

La descripción de estas fases es la siguiente:

**Pre-Game (Antes del juego):** Antes de llevar a cabo el desarrollo del proyecto, se especifica lo que se va a realizar en las iteraciones, además de la prioridad con la que se hará, esta fase consta de dos puntos destacables.

- Planeación: En la cual se define un nuevo entregable basado en el backlog del producto, el cual ya se conoce, junto con una estimación de su horario y costo. Todos los miembros del equipo incluyendo el cliente contribuyen a la creación de una lista de características del sistema, análisis y conceptualización del sistema.
- Arquitectura: En el que se diseñan como los requerimientos del backlog del producto serán puestos en ejecución. Esta fase incluye la revisión de la arquitectura del sistema y diseño de alto nivel.

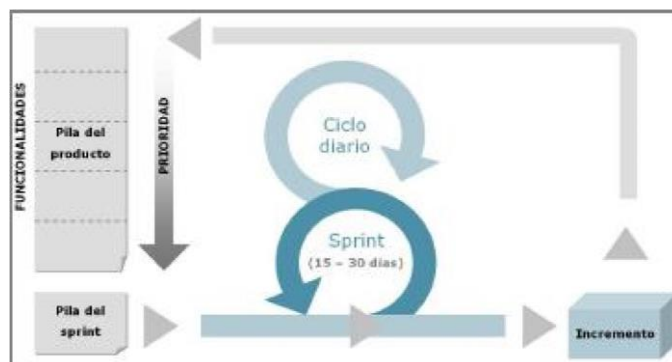
**Game (Juego):** Una vez realizada la especificación correspondiente, se lleva a cabo la elaboración del proyecto con un continuo seguimiento a cargo del mismo grupo, en cada iteración se realizan las tareas [Palacios, 2008].

- Planeación del Sprint: Realizado antes de realizar el Sprint en el cual se lleva a cabo dos reuniones consecutivas, en la primera se refina y se prioriza nuevamente el backlog del producto además de elegir las metas de la iteración; en la segunda reunión se considera como alcanzar los requerimientos y crear el backlog del sprint.
- Desarrollo del Sprint: El Sprint es el desarrollo de la nueva funcionalidad para el producto organizado generalmente en iteraciones de 30 días en el cual se genera la siguiente documentación: Backlog del sprint con las actividades realizadas, los responsables y la duración de la actividad.

**Revisión del Sprint:** Realizada al final de cada iteración en la cual se lleva a cabo una revisión en donde se presenta la nueva funcionalidad del producto e información de las funciones, diseño, ventajas, inconvenientes y esfuerzos del equipo.

**Post-Game (Después del juego):** Realizado después de haber culminado todas las iteraciones solo resta una revisión final denominado cierre:

- Cierre: En esta etapa se realiza la preparación operacional incluyendo la documentación final necesaria para la presentación, se presenta un demo del producto y se toman en cuenta lluvia de ideas para la retroalimentación a futuro.



**Fig. 3** Ciclo de SCRUM  
**Fuente:** [PALACIOS, 2008]

Los elementos que conforman el desarrollo SCRUM son:

### Las reuniones

- **Planificación de sprint:** Jornada de trabajo previa al inicio de cada sprint en la que se determina cual va a ser el trabajo y los objetivos que se deben cumplir en esa iteración.
- **Reunión diaria:** Breve revisión del equipo del trabajo realizado hasta la fecha y el pre revisión para el día siguiente.
- **Revisión de sprint:** análisis y revisión del incremento generado [PALACIOS, 2008].



**Fig. 4** Seguimiento de Sprint  
Fuente: [PALACIO, 2008]

### Los elementos

#### Pila de producto (Blacklog del producto)

Es una lista de requisitos de usuario que se origina con la visión inicial del producto y va creciendo y evolucionando durante el desarrollo, las características principales de esta pila son:

- Priorizar de acuerdo a la importancia que le da el propietario del producto.
- Todos pueden incluir y aportar elementos.
- Debe ser accesible para todos los miembros del equipo del proyecto.
- Responsable directo es el propietario del producto

## **Pila de sprint (Blacklog del sprint)**

Se define Sprint como una iteración que dura alrededor de 30 días la pila de sprint es una lista de trabajos que debe realizar el equipo durante esta iteración para generar el incremento previsto, cuya característica son:

- Debe contener las funcionalidades que se van a realizar durante el Sprint.
- El equipo debe de estar comprometido a realizar dichas funciones.
- Se debe asignar tareas a los distintos miembros del equipo del proyecto.
- Se debe hacer una estimación de cada funcionalidad.

## **Incremento**

Es el resultado de cada Sprint, cuyas características principales son:

- Es parte del producto desarrollado en un Sprint.
- Debe estar en condiciones de ser usado.
- Es una funcionalidad.

Los dos primeros forman los requisitos del sistema que se va a desarrollar, y el tercero es valor que se le entrega al cliente al final de cada sprint. El incremento, como valor real para el cliente, no se trata de un prototipo, o de módulos o subrutinas a falta de pruebas o integración, sino de una parte del producto final, completamente operativa que podría entregarse tal cual al cliente.

### **2.4.2.3. Roles de SCRUM**

En SCRUM, el equipo se focaliza en construir software de calidad. La gestión de un proyecto SCRUM se centra en definir cuáles son las características que debe tener el producto a construir y en vencer cualquier obstáculo que pudiera entorpecer la tarea del equipo de desarrollo.

El equipo SCRUM está formado por los siguientes roles:

- **SCRUM máster (Líder del proyecto):** Responsable del proceso SCRUM, de cumplir la meta y resolver los problemas. Así como también de asegurarse que el proyecto se lleve a cabo de acuerdo a las prácticas.  
Interactúa con el cliente y el equipo. Coordina los encuentros diarios, y se encarga de eliminar eventuales obstáculos. Debe ser miembro del equipo y trabaja a la par.
- **ProductOwner (Dueño del producto):** Representa a todos los interesados en el producto final. Sus áreas de responsabilidades son:
  - Financiación del proyecto
  - Requisitos de sistema
  - Retorno de la inversión del proyecto
  - Lanzamientos de proyecto

Es el responsable oficial del proyecto, gestión, control y visibilidad de la lista de acumulación o lista de retraso del producto.

## Equipo

Responsable de transformar el Backlog de la iteración en un incremento de la funcionalidad del software. Tiene autoridad para reorganizarse y definir las acciones necesarias o sugerir remoción de impedimentos.

- Auto-gestionado
- Auto-organizado
- Multi-funcional

Grupo de profesionales con los conocimientos técnicos necesarios y que desarrollan el proyecto de manera conjunta llevando a cabo historias a las que se comprometen al inicio de cada sprint.



**Fig. 5 Roles de SCRUM**  
**Fuente:** [PALACIOS, 2008]

## Valores

SCRUM es una metodología muy simple en su composición, sin embargo sus fundamentos teóricos y los valores en los que se fundamentan tienen implicaciones que van más allá de la simplicidad de sus componentes.

SCRUM se basa en la sinceridad, transparencia y compromiso de las personas para llevar adelante el proyecto.

Los valores de SCRUM y del manifiesto ágil son el pegamento que une a las personas en las reuniones y a través de los documentos y les permite cumplir con sus compromisos día a día, sprint a sprint hasta el éxito del proyecto.

- **Compromiso:** Estar dispuesto para comprometerse a una meta. La metodología la da a las personas la autoridad que necesitan para cumplir con sus compromisos.
- **Enfoque:** Haz tu trabajo. Enfoca todos tus esfuerzos y habilidades para trabajar en lo que te comprometiste a hacer. No te preocupes por nada más. Alguien lo hará por ti.
- **Transparencia / Honestidad:** SCRUM mantiene todo acerca del proyecto visible a todos.

- **Respeto:** Los individuos estamos formados por nuestros orígenes y nuestras experiencias. Es importante respetar las diferentes a las personas del equipo y sus formas de pensar.
- **Coraje:** Tener el coraje para comprometerse, actuar, ser honesto y esperar respeto [PALACIOS, 2008].

### 2.4.3. Metodología UWE

UWE es un proceso del desarrollo para aplicaciones WEB enfocado sobre el diseño sistemático, la personalización y la generación semiautomática de escenarios que guíen el proceso de desarrollo de una aplicación WEB. UWE describe una metodología de diseño sistemático, basada en la técnica de UML.

Es una herramienta que nos permitirá modelar aplicaciones WEB, utilizada en la ingeniería WEB, prestando especial atención en sistematización y personalización. UWE es una propuesta basada en el proceso unificado UML pero adoptamos a la WEB. En requisitos separa las fases de captura, definición y validación. Hace además una clasificación y un tratamiento especial dependiendo del carácter de cada requisito.

En el marco de UWE es necesario la definición de un perfil UML, basados en estereotipos con este perfil se logra la asociación de una semántica distinta a los diagramas de UML puro, con el propósito de acoplar el UML a un dominio específico, en este caso, las aplicaciones WEB. Entre los principales modelos de UWE podemos citar: el modelo lógico conceptual, modelo de navegación, modelo de presentación, visualización de escenarios Web y la interacción temporal, entre los diagramas: diagramas de estado, secuencia, colaboración, actividad.

UWE define vistas especiales representadas gráficamente por diagramas UML. Además UWE no limita el número de vistas posibles de una aplicación UML proporciona mecanismos de extensión basados en estereotipos. Estos mecanismos de extensión son los que UWE utiliza para definir estereotipos que son los que finalmente se utilizaran en las vistas especiales para el modelado de aplicaciones Web.



De esta manera, se obtiene una notación UML adecuada en un dominio en específico a la cual se le conoce como perfil UML.

#### **2.4.3.1. Fases de la metodología UWE**

UWE cubre todo el ciclo de vida de este tipo de aplicaciones centrando además su atención en aplicaciones personalizadas o adaptativas.

**Las fases o etapas a utilizar son:**

**1) Captura, análisis y especificación de requisitos:** En simples palabras y básicamente, durante esta fase, se adquieren, reúnen y especifican las características funcionales y no funcionales que deberá cumplir la aplicación WEB.

Trata de diferente forma las necesidades de la información, las necesidades de adaptación y las de interfaz de usuario, así como algunos requisitos adicionales. Centra el trabajo en el estudio de los casos de uso, la generación de los glosarios y el prototipo de la interfaz de usuario.

**2) Diseño del sistema:** Se basa en la especificación de requerimientos producido por el análisis de los requerimientos, el diseño define como estos requisitos se cumplirán la estructura que debe darse a la aplicación WEB.

- Diagrama de casos de usos
- Diagrama conceptual
- Diagrama físico
- Diagrama de clases
- Modelo navegación
- Modelo de presentación

**3) Codificación del software:** Durante esta etapa se realizan las tareas que comúnmente se conocen como programación; que consiste, esencialmente, en llevar a código fuente, en el lenguaje de programación elegido, todo lo diseñado en la fase anterior.

4) **Pruebas:** Las pruebas se utilizan para asegurar el correcto funcionamiento de secciones de código.

5) **La instalación o fase de Implementación:** Es el proceso por el cual los programas desarrollados son transferidos apropiadamente al computador destino, inicializados y eventualmente configurados; todo ello con el propósito de ser ya utilizados por el usuario final. Esto incluye la implementación de la arquitectura, de la estructura del hiperespacio, del modelo de usuario, de la interfaz de usuario, de los mecanismos adaptativos y las tareas referentes a la integración de todas estas implementaciones.

6) **El Mantenimiento:** Es el proceso de control, mejora y optimización del software ya desarrollado e instalado, que también incluye depuración de errores y defectos que puedan haberse filtrado en la fase de pruebas de control. [NIEVES Y MENENDEZ, 2014 ]

## 2.5. INGENIERÍA WEB

La ingeniería WEB es un área que abarca procesos, técnicas y modelos orientados a los entornos WEB.

La ingeniería WEB es la aplicación de metodologías sistemáticas, disciplinadas y cuantificables al desarrollo eficiente.

### Áreas de investigación

Las principales áreas de investigación del grupo son:

**Ingeniería del software:** El objetivo es industrializar el proceso de desarrollo de software para reducir costos y mejorar la calidad de los productos software. Para ello se centra las líneas de productos software y el desarrollo de software dirigido por modelos como paradigmas de desarrollo que proporcionan numerosos beneficios respecto al desarrollo de software tradicional. Ambos enfoques utilizados conjuntamente permiten:

- Reducir los costos del ciclo de vida del software.

- Rápida adaptación del sistema a nuevas tecnologías.
- Reducción del tiempo de desarrollo de nuevos sistemas.
- Mejora de la calidad del software permitiendo una fácil evolución del mismo.

Esta área investiga las tecnologías WEB semántica y las estructuras de datos desde el prisma de los sistemas abiertos. Estas tecnologías posibilitan estructuras de datos enriquecidas que se utilizan para:

- Facilitar la búsqueda de contenidos y servicios.
- Crear agentes inteligentes para la automatización de procesos.
- Extraer, procesar y almacenar conocimiento.
- Integrar sistemas heterogéneos.
- Componer sistemas complejos.
- Filtrar información.
- Posibilitar el proceso semántico por parte de las máquinas.
- Estandarizar y flexibilizar vocabularios.

En esta área se investiga en el diseño, desarrollo de plataformas interoperables, basadas en servicios web y el cumplimiento de estándares, la construcción de arquitecturas, que posibilitan la integración, multiplataforma y multidispositivo.

Integración de herramientas y simulación distribuida, el objetivo es integrar herramientas para facilitar el desarrollo de sistemas ciberfísicos y su validación temprana mediante la simulación distribuida. Para ello, utilizamos los siguientes estándares:

- Open Software Lifecycle Collaboration (OSLC): para la integración de herramientas del ciclo de desarrollo de sistemas.
- Functional Mockups Interfaces (FMI): para la simulación de diferentes aspectos de un sistema en la fase de validación temprana.

### **2.5.1. Sistema WEB**

Sistema WEB o también conocidos como aplicaciones WEB son aquellos que están creados o instalados no sobre una plataforma o sistemas operativos Windows o Linux. Sino que se alojan en un servidor en Internet o sobre una intranet. Su aspecto es muy similar a páginas WEB que se ven normalmente, pero en realidad los sistemas WEB tienen funcionalidades muy potentes que brindan respuestas a casos particulares.

Los sistemas WEB se pueden utilizar en cualquier navegador WEB como: chrome, firefox, Internet Explore, opera etc. Sin importar el sistema operativo. Para realizar las aplicaciones WEB no es necesario instalarlas en cada computadora ya que los usuarios se conectan a un servidor donde se aloja el sistema.

Las aplicaciones WEB trabajan con bases de datos que permiten procesar y mostrar información de forma dinámica para el usuario. Los sistemas desarrollados en plataformas WEB, tienen marcadas diferencias con otros tipos de sistemas, lo que lo hacen muy beneficiosos tanto para las empresas que lo utilizan, como para los usuarios que lo operan en el sistema.

Este tipo de diferencias se ven reflejada en los costos, en la rapidez de obtención de la información, en la optimización de las tareas por parte de los usuarios y en alcanzar una gestión estable.

## **2.6. HERRAMIENTAS**

### **2.6.1. Servidor WEB apache**

Es un software de código abierto, libre de uso y totalmente configurable, es en este momento el más utilizado en la red, ya sea en plataformas Linux o Windows. Un servidor WEB es un software instalado en el equipo con todas las condiciones necesarias para servir o entregar páginas web que le sean solicitadas por un navegador, asegurando que se muestren y representen todos los elementos necesarios para su correcto funcionamiento y visualización.

### **2.6.2. Framework Laravel**

Es un framework de PHP para el desarrollo rápido de aplicaciones WEB. Su filosofía es desarrollar código PHP de forma elegante y simple, evitando el código espagueti. [laravel.com]

### **2.6.3. Framework Vuejs**

Vue.js, es un framework de JavaScript popular para desarrollar aplicaciones modernas y escalables en el lado del cliente. [vuejs.org]

### **2.6.4. Base de datos MariaDB**

Es un sistema gestor de bases de datos (SGBD), es decir, un conjunto de programas que permiten modificar, almacenar, y extraer información de una base de datos.

MariaDB surge a raíz de la compra, de la compañía desarrolladora de otro SGBD llamado MySQL, por la empresa Sun Microsystems. El desarrollador original, decide tomar el código fuente original de MySQL y genera un derivado con mejoras y cambios a los que llama MariaDB. Permitiendo así la existencia de una versión de este producto con licencia General Public License. [nerion.es]

## **2.7. CALIDAD DEL SOFTWARE ISO 9126**

La calidad de software está definida como la concordancia con los requisitos funcionales y de rendimiento precisamente establecidos, los estándares de desarrollo explícitamente documentados y con las características implícitas que se espera de todo aquel software profesionalmente. [PRESSMAN, 2002]

Entre las conclusiones más importantes a la que llega a cerca de calidad del software a partir de la norma ISO están:

- Los requisitos del software son la base de las medidas de calidad. La falta de concordancia con los requisitos es una falla de calidad.

- Las metodologías definen un conjunto de criterios de desarrollo que guían la forma en se aplica la ingeniería del software. Si no se sigue ninguna metodología siempre existirá la falta de calidad.
- Existen algunos requisitos implícitos o expectativas que a menudo no se mencionan, o se insinúan de forma incompleta que puede implícita una falta de calidad.

A partir de todo esto, la meta no es siempre conseguir la calidad perfecta, pero si la calidad necesaria y suficiente como para cumplir los requisitos o necesidades de los usuarios finales. También es muy importante conseguir de un sistema siempre varia de uno a otro.

Para poder medir la calidad, es necesario identificar aspectos importantes del sistema que contribuyen a la eficacia global, con el fin de relacionar la visión externa del usuario con la interna del desarrollador, uno de los modelos que ayuda a determinar estos es la norma ISO 9126 que considera seis características muy importantes como es la funcionalidad, confiabilidad, usabilidad, eficiencia, factibilidad de mantenimiento y la portabilidad. Esta norma mide la calidad de un sistema desde la visión del usuario y no así desde la del producto.

### **2.7.1. Confiabilidad**

En términos estadísticos se define como la probidad de operación libre de fallos del sistema en un entorno determinado y durante el tiempo este factor es esencial para la evaluación de la calidad del software. [PRESSMAN, 2002]

El nivel de confiabilidad del software está estrictamente relacionado con la cantidad de errores que lanza el sistema durante el tiempo de ejecución y no en tiempo real. La confiabilidad es expresada en un escala de 0 al 1, en el que el sistema será muy confiable si el valor obtenido se acerca a 1 y viceversa, es decir que a menor error mayor será la confiabilidad ante el usuario.

### **2.7.2. Facilidad de Mantenimiento**

La facilidad de mantenimiento de un sistema es la facilidad con la que se puede localizar y corregir un error o errores dentro del sistema

Se centra más en el cambio que va asociado a la corrección de errores durante la creación del sistema o mantenimiento correctivo, dicho en otras palabras se concentra más a las adaptaciones requeridas a medida que evoluciona e entorno del software o mantenimiento adaptivo, esto como consecuencia de cambios en las políticas de la empresa, a la adaptación de nuevos sistemas operativos, y al descubrimiento de funciones adicionales que van a producir beneficios más allá de sus requisitos funcionales originales o mantenimiento perpetuo y finalmente el cambio del producto pensado en mejoras futuras o ingeniería del software.

### **2.7.3. Usabilidad**

La usabilidad se define como el esfuerzo necesario para comprender el software en general, reparar los datos de entrada e interpretar salidas. Es el intento por medir lo amigable que puede ser un programa con el usuario final.

### **2.7.4. Eficiencia**

La eficiencia se define como el grado en el que el software emplea óptimamente los recursos del sistema, según lo indican los sub atributos siguientes: comportamiento del tiempo y de los recursos.

### **2.7.5. Portabilidad**

La portabilidad se define como la facilidad con la que el software puede llevarse de un ambiente a otro según lo indican los siguientes atributos: adaptable, conformidad y sostenible. [PRESSMAN, 2002]

## **2.8. SEGURIDAD DE SOFTWARE**

En la actualidad la seguridad en la informática va adquiriendo mayor importancia por el incremento del volumen de datos que se hallan en computadoras de empresas tanto privadas como públicas, en este tipo de sistemas para un usuario experto es fácil tener acceso a información confidencial. Por esta razón se dio la necesidad de establecer un mecanismo de seguridad mediante el uso de autorización, autenticación y la encriptación o cifrado de la información de tal manera que aquellos que lo intercepten sin ningún permiso los datos sean ilegibles a plena vista,

### **2.8.1. autenticación**

Se encarga de identificar a los usuarios que inician sesión desde cualquier punto determinado de una red ya sean estas privadas o públicas. Para el presente proyecto se realiza un servicio de autenticación mediante un usuario y contraseña.

### **2.8.2. Autorización**

Se encarga de proporcionar al usuario a la red o aplicación de acuerdo con los derechos o al nivel que le corresponda, se implementan grupos de usuarios a los cuales se les da ciertos criterios privilegios de acceso dependiendo de la función que realicen dentro la empresa.

### **2.8.3. Cifrado**

El cifrado es una forma de transferir dicha información mediante la red, de tal forma que esta sea ilegible a usuarios no admitidos, los cuales sean capas de interceptar en el transcurso de una lugar a otro.

### **2.8.4. Encriptación**

Es una forma más segura por el cual se puede mandar datos, tales como la contraseña o el mismo nombre de usuario al servidor mediante la red para su previa verificación y si autorización de acceso al sistema al cual desea ingresar.



### 2.8.5. Seguridad de base de datos

Se refiere a medidas de protección de la privacidad digital que se aplican para evitar el acceso no autorizado a los datos, los cuales pueden encontrarse en ordenadores , base de datos, sitios web, etc.

### 2.9. PRUEBAS DE CALIDAD NORMA ISO/IEC 25010

En la actualidad los productos de software son una parte importante en nuestras actividades. Se debe tener en cuenta que los productos de software son creados, desarrollados e implementados por seres humanos y por ende en cualquiera de sus etapas de creación se puede presentar una equivocación, al generarse una equivocación se puede llevar a un defecto, como la mala digitalización, distracción al codificar entre otras.

Con el fin de garantizar el aseguramiento de calidad se han creado una serie de pruebas, como se observa en la siguiente tabla:

<b>Nombre de la prueba</b>	<b>Descripción</b>
Pruebas unitarias	Se focaliza en ejecutar cada módulo, lo que provee un mejor modo de manejar la integración de las unidades en componentes mayores
Pruebas de integración	Identificar errores introducidos por la combinación de programas probados anteriormente
Pruebas del sistema	Asegurar la apropiada navegación dentro del sistema, ingreso de datos, procesamiento y recuperación,
Pruebas de desempeño	Las pruebas de desempeño miden tiempos de respuestas, índices de procesamiento de transacciones y otros requisitos sensibles al tiempo

Pruebas de carga	Verificar el tiempo de respuesta del sistema para transacciones o casos de uso de negocios, bajo diferentes condiciones de carga
Pruebas de estilo	Comprobar que la aplicación sigue los estándares de estilos propios del cliente
Pruebas de integridad de datos y base de datos	Asegurar que los métodos de acceso y procesos funcionan adecuadamente y sin ocasionar corrupción de datos
Pruebas de seguridad y control de acceso	Nivel de seguridad de la aplicación, verifica que un actor solo pueda acceder a las funciones y datos que su usuario tiene permitido.

**Tabla 1:** Enfoques de las pruebas  
**Fuente:** [MERA, MIRANDA y CUARAN, 2017]

## 2.10. ESTUDIO DE COSTOS DEL SISTEMA WEB

### 2.10.1. Modelo Cocomo II

COCOMO II es un modelo que permite estimar el costo, el esfuerzo y el tiempo cuando se planifica una nueva actividad de desarrollo software, y está asociado a los ciclos de vida modernos. Fue desarrollado a partir de COCOMO, incluyendo actualizaciones y nuevas extensiones más adecuadas a los requerimientos de los ingenieros software.

Está construido para satisfacer aquellas necesidades expresadas por los estimadores software, como por ejemplo el apoyo a la planificación de proyectos, la previsión de personal de proyecto, re planificación, seguimiento, negociaciones de contrato o la evaluación del diseño.

COCOMO II proporciona una familia de modelos de estimación muy detallados, y que tiene muy en cuenta el tipo información disponible. Este conjunto de modelos se divide en tres:

- Modelo de composición de aplicaciones, para proyectos de construcción de interfaces gráficas de usuario.
- Modelo de diseño preliminar, utilizado para obtener estimaciones sobre el coste de un proyecto, antes de que esté determinada por completo su arquitectura.
- Modelo post-arquitectura, que es el más detallado, y debe ser usado una vez determinada la arquitectura al completo.

El usar un modelo u otro depende del nivel de detalle del proyecto, de la fidelidad requerida de las estimaciones, y de la definición de los requerimientos y de los detalles de la arquitectura. De una manera general, podemos determinar en qué momentos es más adecuado la utilización de un modelo u otro:

- El modelo de composición de aplicaciones se aplicará sobre todo en las primeras fases o ciclos en espiral.
- El modelo de diseño preliminar es útil para las siguientes fases o ciclos espirales, en los que se incluye la exploración de arquitecturas alternativas o estratégicas de desarrollo incremental.
- Una vez que el proyecto está listo para desarrollar y sostener un sistema especializado, el sub modelo de Post-arquitectura proporciona información más precisa de los controladores de costo de entradas y permite cálculos de coste más exactos.

Los modelos usan la siguiente ecuación:

$$PM = 2.45 * EAF * \text{Tamaño B}$$

Donde EAF o factor de ajuste de esfuerzo es el producto de siete multiplicadores de esfuerzo para el modelo de desarrollo temprano, y diecisiete para el modelo post arquitectural.

B es un factor que toma valores comprendidos entre 1.01 y 1.26, que viene dado por:

$B = 1.01 + \sum w_i$ , siendo  $\sum w_i$  la suma de cinco factores que causan efecto en la economía del proyecto, y que son:

- Existencia de precedentes al proyecto.
- Flexibilidad del desarrollo.
- Resolución de riesgos.
- Cohesión del equipo.
- Madurez del proceso.

## CAPÍTULO III

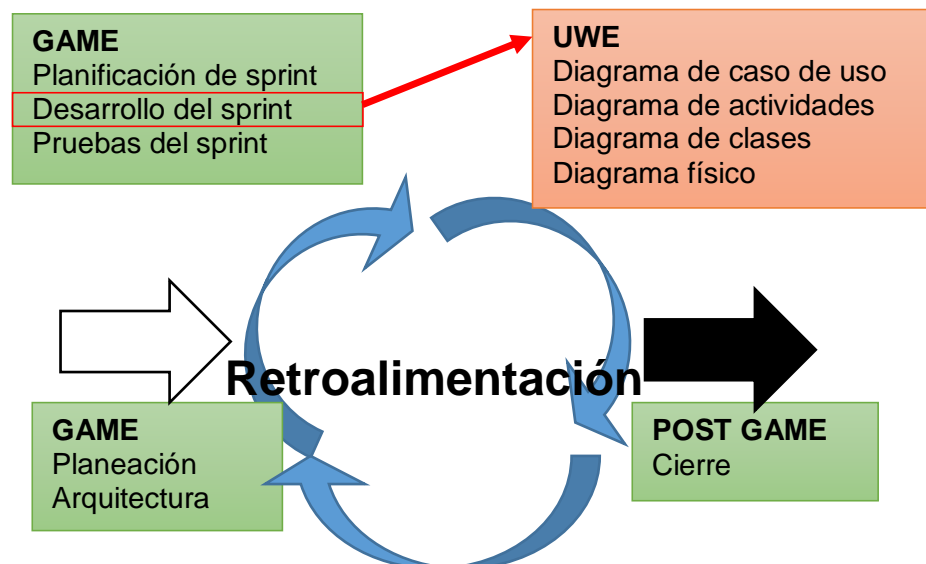
### MARCO APLICATIVO

#### 3.1. INTRODUCCIÓN

En el presente capítulo tiene como finalidad describir el análisis, diseño, gestión, evaluación de métricas de calidad y costo estimado del proyecto, dado que XP y SCRUM son metodologías ágiles que requiere de trabajo organizado y en equipo, limitando el tiempo, porque la gestión no se basa en el seguimiento de un plan, sino en la adaptación continua a las circunstancias de la evolución del proyecto.

Para la gestión, desarrollo y retroalimentación del proyecto se usarán las metodologías ágiles XP y SCRUM para limitar el tiempo y trabajo en equipo, ya que las dos metodologías utilizan un modelo de proceso incremental, trabajo en equipo y para las etapas de diseño para cada iteración se complementará la metodología UWE.

Al comenzar cada iteración sprint se determinará que partes se van a construir para el desarrollo del sistema y la cantidad de trabajo que abordará en cada iteración.



**Fig. 6** Planificación  
Fuente: [Elaboración propia]

## 3.2. COMBINANDO SCRUM CON XP

En este caso, la base del proyecto será XP en su totalidad, respetando sus cinco valores y las 12 practicas técnicas sin alteraciones.

Esta base, será complementada, por los artefactos y ceremonias de SCRUM, adaptados a los valores de XP:

- Organización por Sprints
- Planificación, revisión y retrospectiva
- Estimación por Planning Poker, T-Shirt Sizing.

Los roles, solo quedarán repartidos entre Cliente y Equipo. La figura del SCRUM Master queda exceptuada en esta combinación. [BAHIT, 2012]

## 3.3. PRE-GAME

### 3.3.1. Recopilación de requerimientos

Se presenta el Blacklog del producto, que contiene los requerimientos y características finales del sistema.

**Las reuniones o trabajo en equipo**, se generan el “Sprint Blacklog” o lista de tareas que se van a realizar, y que en ella se determinara los requerimientos del sistema.

**Los requisitos del sistema**, es parte de la visión del resultado que se desea obtener, y evolucionar durante el desarrollo.

**Los elementos (producto Blacklog)**, requisitos del sistema es parte de la visión del resultado que se desea obtener y que evoluciona durante el desarrollo, y es el inventario de la característica que el propietario del producto desea obtener, ordenando por orden de prioridad.

### 3.3.2. Especificación de requerimientos

Los requerimientos se obtuvieron a través de las entrevistas con los usuarios interesados de la institución, como resultado, se presenta a continuación los

requerimientos funcionales que debe cumplir el sistema a desarrollar. Así mismo el control interno es parte de los procesos básicos de planificación, ejecución, supervisión y está incorporado en ellas.

<b>ID</b>	<b>DESCRIPCIÓN</b>	<b>MÓDULO</b>	<b>PRIORIDAD</b>	<b>ESTADO</b>
R1	Base de datos independiente único para el sistema de Gestión de inventarios para almacenes.	Plataforma	Alta	Terminado
R2	Diseño de registros o formularios impresos a aplicar para una información fiable.	Plataforma	Media	Terminado
R3	Interfaz web fácil de manejar.	Comunicación	Media	Terminado
R4	Control y acceso seguro y diferenciado a usuarios.	Plataforma	Alta	Terminado
R5	Automatización de los procesos de registro que interactuara con el sistema.	Plataforma	Alta	Terminado
R6	Registro adecuado de compra de equipos electrónicos.	Contenido	Media	Terminado
R7	Desarrollo de una interfaz web donde se obtiene información y actas de ingreso.	Contenido	Media	Terminado
R8	Registro adecuado de salida de equipos electrónicos.	Plataforma	Alta	Terminado
R9	Desarrollo de una interfaz web donde se obtiene o información y actas de salidas	Contenido	Media	Terminado
R10	Control de niveles de existencia	Plataforma	Alta	Terminado
R11	Control de Stocks de almacén	Contenido	Media	Terminado
R12	Búsqueda de registros almacenados mediante filtros.	Plataforma	Alta	Terminado

R13	Control de garantías de equipos electrónicos.	Plataforma	Alta	Terminado
R14	Pruebas de software	plataforma	Media	Terminado

**Tabla 2:** Requerimiento  
**Fuente:** [Elaboración propia]

### 3.3.3. Análisis de Riesgos

Un riesgo es la probabilidad de que ocurra algo adverso, existen tres tipos de riesgos:

- **Riesgo de proyecto.-** es el que afecta a la calendarización o recursos del proyecto.
- **Riesgos de producto.-** afectan a la calidad o rendimiento del software que se está desarrollando.
- **Riesgo de negocio.-** afecta a la organización que desarrolla o suministra el software.

RIESGO	TIPO	DESCRIPCIÓN	PROB.	EFECTO	ESTRATÉGIA
No se cumplen las fechas establecidas en el cronograma.	Proyecto	Es probable que en la fecha del diagrama de Gantt no se cumplan al pie de la letra.	Alta	Tolerable	Realizar un segundo cronograma que sea más flexible.
Cambio en los requerimientos del cliente.	Proyecto Producto	Riesgos de que haya cambios en la empresa.	Moderada	Tolerable	Realizar una revisión constante a los requerimientos.



No se cumple con los plazos de entrega del producto.	Producto		Moderada	Serio	Agilizar los procesos de desarrollo del producto. Realizar una correcta planificación considerando el tiempo y los alcances del proyecto.
--	----------	--	----------	-------	---

**Tabla 3:** Análisis de riesgo  
**Fuente:** [Elaboración propia]

### 3.4. GAME

Durante esta etapa del proyecto se desarrollaron cuatro iteraciones de retroalimentación con el usuario, cada una de ellas corresponde a un elemento del sistema de gestión de inventarios. A continuación se desglosan las actividades realizadas a cada una de estas etapas.

Las estrategias que se usan para el desarrollo de cada iteración fue construir en el principio los modelos de la metodología UWE y posteriormente implementarlos utilizando como elemento central la Base de Datos, la solución por la que se optó para la persistencia de objetos fue hacer corresponder cada clase de modelo conceptual con una tabla de base de datos en donde las filas representan las instancias de los objetos y las columnas a los atributos de la clase. Las clases y sus métodos fueron implementados en el framework LARAVEL, finalmente se desarrollaron la página WEB en base a los modelos de presentación y de navegación.

#### 3.4.1. Primera Iteración

Durante la primera iteración se desarrollaron los elementos necesarios para el desarrollo del sistema de gestión de inventarios Registro, Control, y Seguimiento de los procesos. Las actividades realizadas durante esta iteración se observa en la tabla que constituye el Blacklog del Sprint.

		<b>SPRINT</b>	<b>INICIO</b>	<b>DURACIÓN</b>
		<b>1</b>	<b>02-09-2019</b>	<b>30 Días</b>
<b>ID</b>	<b>TAREA</b>	<b>TIPO</b>	<b>DIAS DE TRABAJO</b>	<b>ESTADO</b>
1.1	Realizar la planificación de la iteración.	Planificación	3	Terminado
1.2	Analizar los requerimientos del Blacklog del producto.	Planificación	3	Terminado
1.3	Analizar los requerimientos de la iteración de casos de uso.	Diseño	2	Terminado
1.4	Construir diagrama de actividades.	Diseño	2	Terminado
1.5	Construir diagrama de clases.	Diseño	2	Terminado
1.6	Construir diagrama físico.	Diseño	2	Terminado
1.7	Diseñar y construir la base de datos del sistema	Desarrollo	3	Terminado
1.8	Construir diagrama de presentación.	Diseño	2	Terminado
1.9	Desarrollar el módulo de administración.	Desarrollo	4	Terminado
1.10	Desarrollar el módulo de ingresos de equipos electrónicos.	Desarrollo	2	Terminado
1.11	Desarrollar el módulo de salida de equipos.	Desarrollo	2	Terminado
1.12	Probar los módulos elaborados en la iteración con los clientes.	Revisión	3	Terminado

**Tabla 4:** Primera Iteración  
**Fuente:** [Elaboración Propia]

Funcionalidades correspondientes al incremento de la iteración son:

- Base de datos independientes del sistema.
- Página de ingreso con control de acceso a los usuarios.
- Módulo de administración del sistema.
- Módulo de ingreso de equipos electrónicos.
- Módulo de salida de equipos electrónicos

### 3.4.2. Segunda Iteración

En la segunda iteración se desarrollaron los módulos de ingreso de equipos electrónicos y administración.

		SPRINT	INICIO	DURACIÓN
		2	25-11-19	30 días
ID	TAREA	TIPO	DIAS DE TRABAJO	ESTADO
2.1	Realizar la planificación de la iteración.	Planificación	3	Terminado
2.2	Analizar los requerimientos del Blacklog del producto.	Planificación	2	Terminado
2.3	Analizar los requerimientos de la iteración de casos de uso.	Diseño	2	Terminado
2.4	Construir diagrama de presentación.	Diseño	2	
2.5	Complementar el módulo de administración que interactuara con el ingreso de equipos electrónicos.	Desarrollo	4	Terminado
2.6	Complementar en registro de ingreso de equipos electrónicos el año de garantía en meses.	Desarrollo	3	Terminado
2.7	Desarrollar detalle de ingresos de equipos electrónicos en el módulo de ingresos.	Desarrollo	4	Terminado
2.8	Complementar el módulo de salidas de equipos electrónicos.	Desarrollo	4	Terminado
2.9	Desarrollar detalle de salidas de equipos electrónicos en el módulo de salidas para tener su estado.	Desarrollo	3	Terminado
2.10	Probar los módulos elaborados en la iteración con los clientes.	Revisión	3	Terminado

**Tabla 5:** Segunda Iteración  
**Fuente:** [Elaboración Propia]

En la segunda iteración se desarrollaron las siguientes funcionalidades para el sistema:

- Módulo de administración, centros y oficinas.
- Módulo de ingreso, detalle ingresos y manejo de garantías de equipos electrónicos en meses.
- Módulo de salidas, detalle salidas y estado de equipos electrónicos.
- Desarrollo el control de entradas y salidas de equipos electrónicos.

### 3.4.3. Tercera Iteración

En la tercera iteración se desarrolló el módulo de artículos.

		<b>SPRINT</b>	<b>INICIO</b>	<b>DURACIÓN</b>
		<b>3</b>	<b>25-11-2019</b>	<b>31 días</b>
<b>ID</b>	<b>TAREA</b>	<b>TIPO</b>	<b>DIAS DE TRABAJO</b>	<b>ESTADO</b>
3.1	Realizar la planificación de la iteración.	Planificación	3	Terminado
3.2	Analizar los requerimientos del Blacklog del producto.	Planificación	3	Terminado
3.3	Analizar los requerimientos de la iteración de casos de uso.	Diseño	2	Terminado
3.4	Complementar diagrama de actividades.	Diseño	2	Terminado
3.5	Complementar diagrama de clases.	Diseño	2	Terminado
3.6	Complementar diagrama físico.	Diseño	2	Terminado
3.7	Construir diagrama de presentación.	Diseño	4	Terminado
3.8	Desarrollo de actas de ingresos e informes.	Desarrollo	5	Terminado
3.9	Desarrollo de estadísticas.	Desarrollo	5	Terminado
3.10	Probar los módulos elaborados en la iteración con los clientes.	Revisión	3	Terminado

**Tabla 6:** Tercera Iteración  
**Fuente:** [Elaboración Propia]

En la tercera iteración se desarrollaron las siguientes funcionalidades para el sistema.

- Desarrollo las actas e informes de entregas.
- Desarrollo estadísticas.
- Diseño artículos para el manejo de inventarios.

#### 3.4.4. Cuarta Iteración

Finalmente durante la cuarta iteración se desarrolló los sistemas para dar de castigo a equipos electrónicos. Las actividades durante esta iteración se las puede observar en el siguiente Blacklog.

		SPRINT	INICIO	DURACIÓN
		4	07-01-2020	22 días
ID	TAREA	TIPO	DIAS DE TRABAJO	ESTADO
4.1	Realizar la planificación de la iteración.	Planificación	2	Terminado
4.2	Analizar los requerimientos del Blacklog del producto.	Planificación	2	Terminado
4.3	Analizar los requerimientos de la iteración de casos de uso.	Diseño	2	Terminado
4.4	Complementar diagrama de actividades.	Diseño	1	Terminado
4.5	Complementar diagrama de clases.	Diseño	1	Terminado
4.6	Complementar diagrama físico.	Diseño	1	Terminado
4.7	Construir diagrama de presentación.	Diseño	2	Terminado
4.8	Desarrollo de bajas de equipos electrónicos y equipos defectuosos.	Desarrollo	5	Terminado
4.9	Desarrollo de informe de bajas.	Desarrollo	3	Terminado

4.10	Probar las funcionalidades elaboradas durante la iteración con los clientes.	Revisión	3	Terminado
------	--	----------	---	-----------

**Tabla 7:** Cuarta Iteración  
**Fuente:** [Elaboración Propia]

Durante la cuarta iteración y última se desarrollaron las siguientes funcionalidades para el sistema:

- Búsquedas.
- Módulo de bajas.
- Impresión de informe de baja.

### 3.5. MODELADO DEL SISTEMA

#### 3.5.1. Diagrama de casos de uso

Es un modelo del sistema que contiene actores, casos de uso y sus respectivas relaciones.

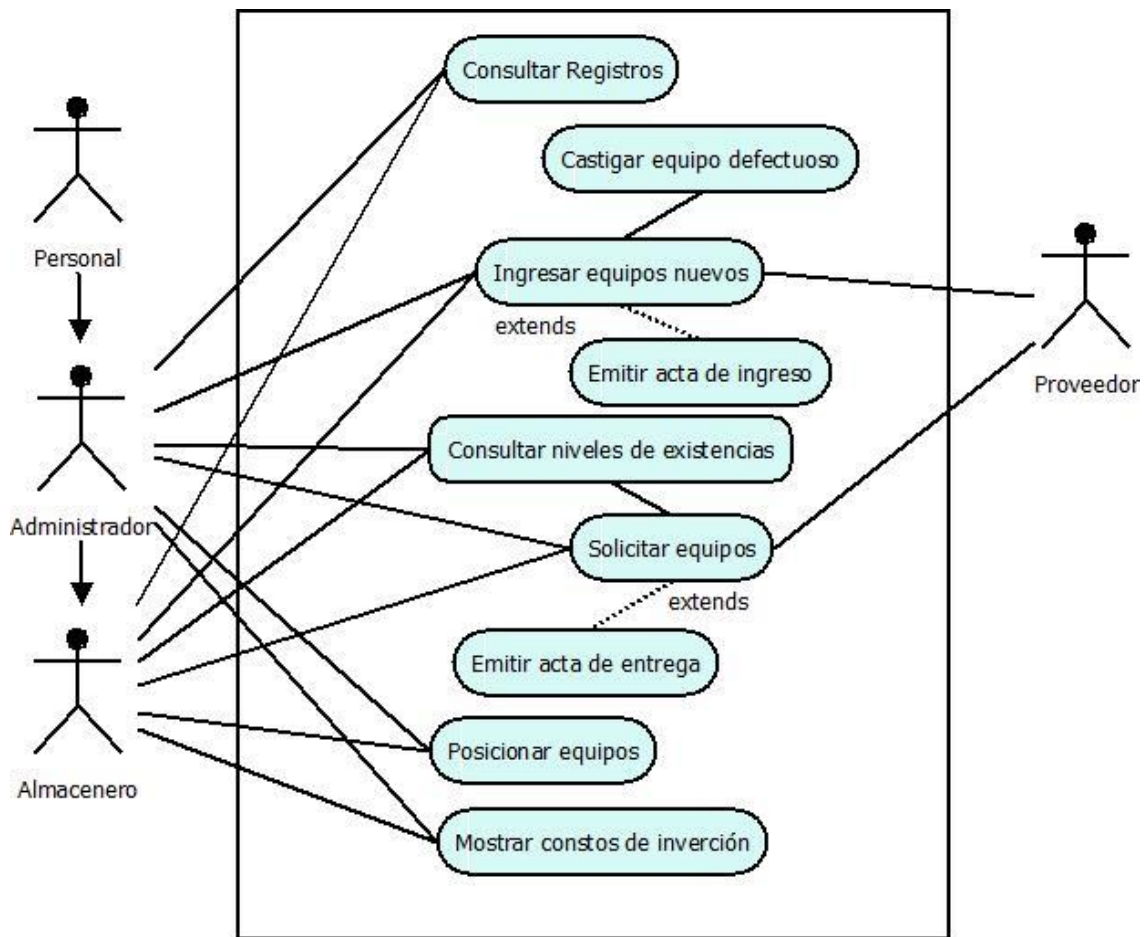
A continuación se describen las características de los actores identificados en el manejo e implementación del sistema.

<b>ACTORES</b>	<b>DESCRIPCIÓN</b>
Administrador	Es la persona que dentro del sistema web a desarrollarse tendrá todos los privilegios sobre el manejo y administración de la información. Toma conciencia del informe enviado al sistema en cada etapa y realiza la previa evaluación de los recursos.
Almacenero	Es la persona encargada de realizar la compra y salida de equipos de computación de la institución. Toma conciencia del informe enviado al sistema en cada etapa y realiza la previa evaluación de los recursos.
Proveedor	Es la empresa encargada de equipos electrónicos a la institución.
Centros	Son las secciones que tiene la institución el cual será distribuidos los equipos electrónicos.

**Tabla 8:** Descripción de actores  
**Fuente:** [Elaboración Propia]

### 3.5.1.1. Diagrama de Caso de Uso General

Este modelo se presenta mediante diagramas de Casos de Uso el cual se piensan en cuáles serán los principales funcionalidades que el software debe permitir llevar a cabo, mientras que los actores representan los roles o papeles que actúan recíprocamente con los procesos.



**Fig. 7** Diagrama de casos de uso general

Fuente: [Elaboración Propia]

### 3.5.2. Descripción de caso de uso

En este diagrama se representan la interacción de los diferentes actores con sistema.

### 3.5.2.1. Diagrama de Caso de Uso: Registro de usuario

Para el registro de usuarios al sistema el actor principal es el administrador, se encarga de llenar un formulario de datos del nuevo usuario, dar un rol, habilitar o en algún caso deshabilitar al usuario como se muestra en la siguiente figura.



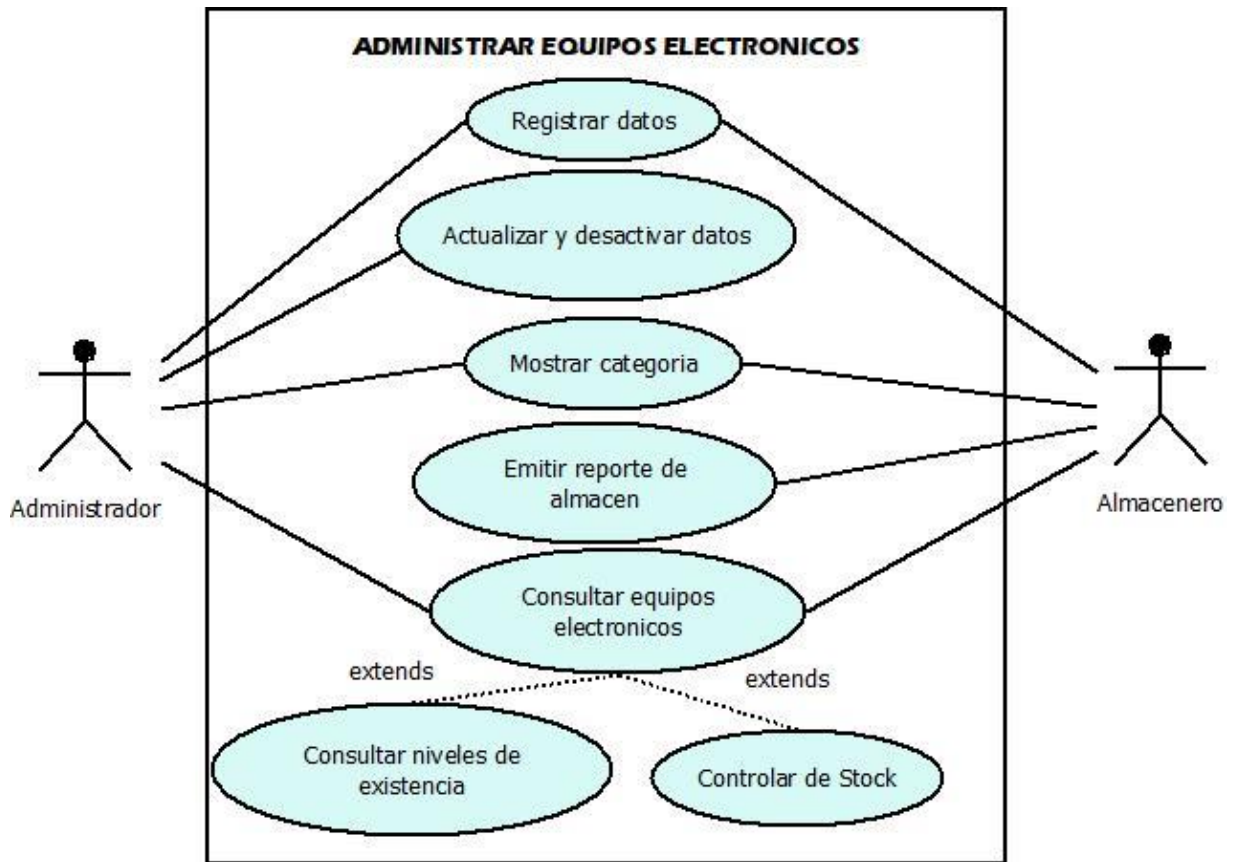
**Fig. 8** Diagrama de casos de uso registro de usuario en el sistema

Fuente: [Elaboración Propia]

### 3.5.2.2. Diagrama de caso de uso: Administración de equipos electrónicos

Para la actualización, desactivación y activación solo los actores que pueden realizar esa acción son administradores, donde se observara control de stock y niveles de existencia de cada equipo electrónico.

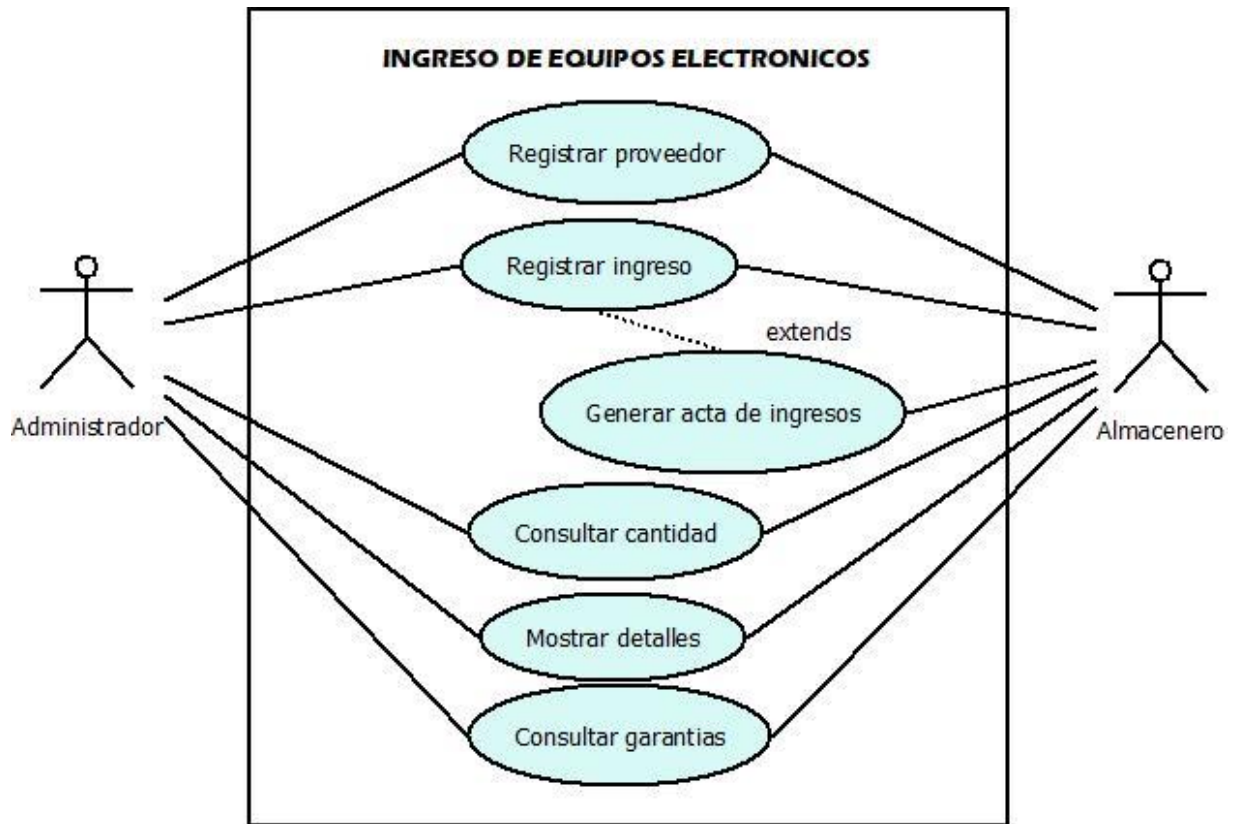




**Fig. 9** Diagrama de casos de administración de equipos electrónicos  
**Fuente:** [Elaboración Propia]

### 3.5.2.3. Diagrama de caso de uso: Ingreso de equipos electrónicos

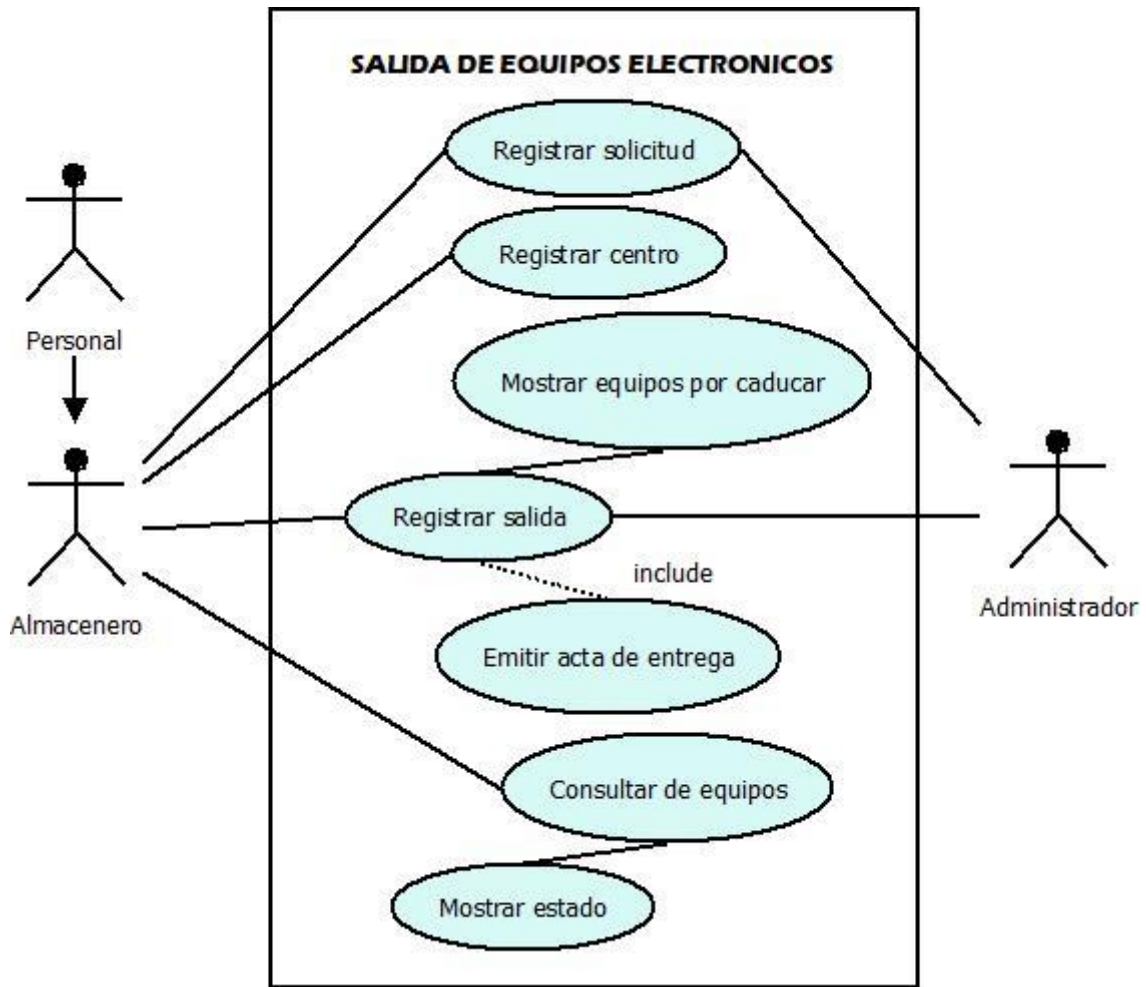
En el módulo de ingreso de equipos electrónicos se registrara la garantía en meses, para su nuevo registro se emitirá el acta de ingreso detallando toda su información y su respectivo caso de uso esta descrita en la siguiente figura.



**Fig. 10** Diagrama de casos de ingreso de equipos electrónicos  
**Fuente:** [Elaboración Propia]

#### 3.5.2.4. Diagrama de caso de uso: Salida de equipos electrónicos

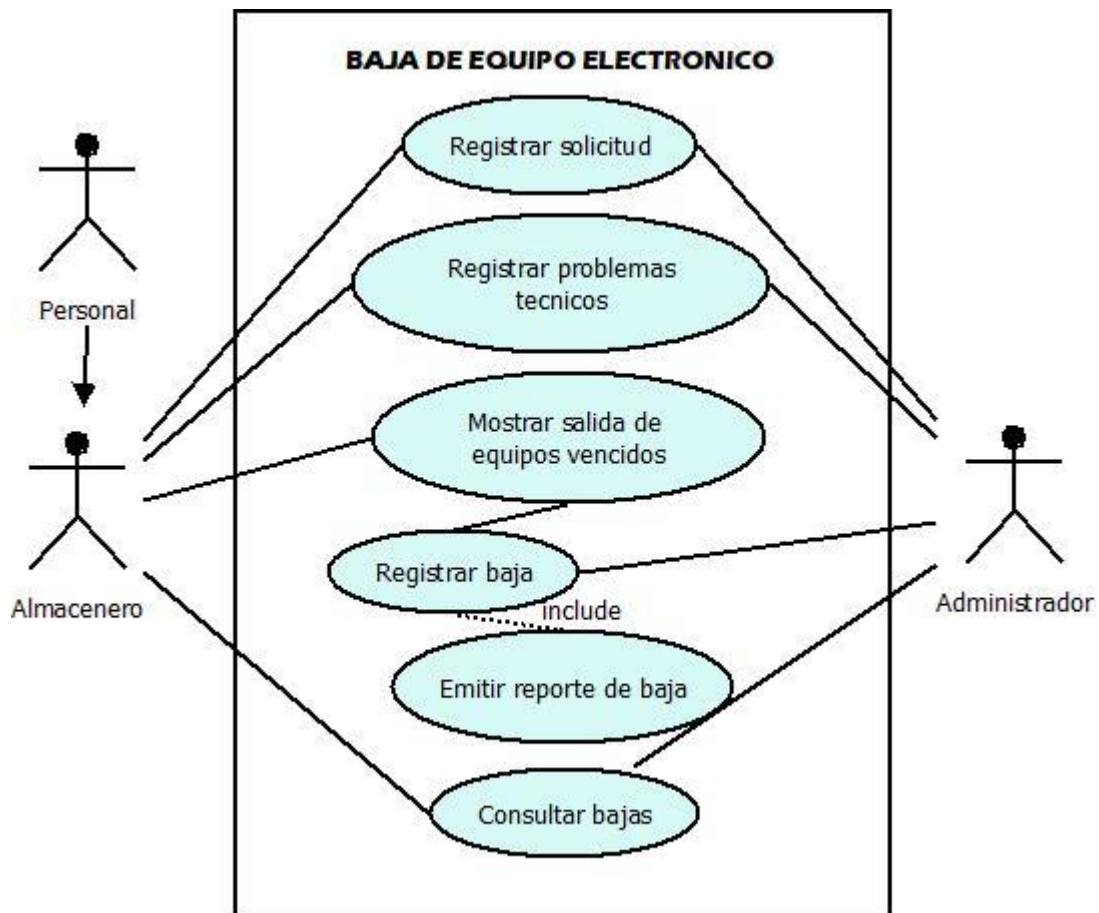
En el módulo de salida de equipos electrónicos, se registrará la sección de la institución y el serial del equipo para controlar en el futuro si sigue vigente su garantía, al registrar una nueva salida emitirá su acta de entrega del equipo.



**Fig. 11** Diagrama de casos de salida de equipos electrónicos  
**Fuente:** [Elaboración Propia]

### 3.5.2.5. Diagrama de caso de uso: Baja de equipos electrónicos

En el módulo de baja de equipos electrónicos, se registrara al equipo que ya no cuenta con garantía y posteriormente se emitirá el informe de baja.



**Fig. 12** Diagrama de casos de bajas de equipos electrónicos

Fuente: [Elaboración Propia]

### 3.5.2.6. Descripción de casos de usos

En esta sección se tiene la descripción de los casos de uso.

CASO DE USO	REGISTRO DE USUARIO
ACTOR	Administrador, almacenero
PRE-CONDICIÓN	El administrador debe estar autenticado en el sistema y el almacenero debe ser registrado por el administrador con usuario contraseña y cargo.
ESCENARIO BÁSICO	El caso de uso comienza cuando el usuario administrador requiere registrar un nuevo almacenero o realizar cambio de cargo de dicho almacenero. 1. El administrador selecciona la opción de registro de usuario.

	<ol style="list-style-type: none"> <li>2. El sistema despliega la interfaz de registro de registro de nuevo usuario.</li> <li>3. El administrador ingresa los datos solicitados.</li> <li>4. El administrador asigna usuario y contraseña del nuevo usuario.</li> <li>5. El administrador asigna el cargo del nuevo usuario.</li> <li>6. El administrador modifica los datos de los usuarios.</li> <li>7. El administrador selecciona guardar o cancelar los datos ingresados.</li> <li>8. Despliega alerta de conformidad.</li> </ol>
<b>ESCENARIOS ITERATIVOS</b>	
ALTERNATIVA - 1	<p>Si en el escenario 8 del escenario básico el sistema encuentra datos que no concuerdan con los requerimientos, entonces el sistema despliega alertas que no fueron introducidos.</p> <p>Si en el escenario 5 del escenario básico el sistema encuentra datos duplicados en usuario, entonces el sistema despliega alertas.</p>
POST-CONDICIÓN	<ol style="list-style-type: none"> <li>1. Los datos de registro del nuevo usuario están almacenados en la Base de Datos.</li> </ol>

**Tabla 9:** Registro de usuario

**Fuente:** [Elaboración Propia]

<b>CASO DE USO</b>	<b>ADMINISTRACIÓN DE EQUIPOS ELECTRÓNICOS</b>
ACTOR	Administrador, almacenero
PRE-CONDICIÓN	El administrador o almacenero debe estar autenticado en el sistema.
ESCENARIO BÁSICO	<p>El caso de uso comienza cuando el administrador o almacenero requiere registrar un nuevo equipo electrónico al sistema.</p> <ol style="list-style-type: none"> <li>1. El usuario selecciona la opción de registro de equipos electrónicos.</li> <li>2. El sistema despliega la interfaz de registro de nuevo equipo electrónico.</li> <li>3. El usuario ingresa los datos del equipo electrónico.</li> <li>4. El usuario selecciona guardar o cancelar los datos ingresados.</li> <li>5. El administrador modifica los datos de los equipos electrónicos.</li> </ol>

	<ol style="list-style-type: none"> <li>6. Despliega alerta de conformidad.</li> <li>7. El sistema muestra interfaz de niveles de existencia de equipos electrónicos.</li> <li>8. El sistema controla Stock de los equipos electrónicos.</li> <li>9. El sistema emitirá reportes de equipos electrónicos almacenados.</li> </ol>
<b>ESCENARIOS ITERATIVOS</b>	
ALTERNATIVA - 1	Si en el escenario 6 del escenario básico el sistema encuentra datos que no concuerdan con los requeridos, entonces el sistema despliega alertas que no fueron introducidos correctamente.
POST-CONDICIÓN	<ol style="list-style-type: none"> <li>1. Los datos ingresados están almacenados en la Base de Datos.</li> <li>2. Automatiza niveles de existencia.</li> <li>3. Automatiza control de Stock.</li> <li>4. Emite reportes de almacenes.</li> </ol>

**Tabla 10:** Administración de equipos electrónicos  
**Fuente:** [Elaboración Propia]

<b>CASO DE USO</b>	<b>INGRESO DE EQUIPOS ELECTRÓNICOS</b>
ACTOR	Administrador, almacenero
PRE-CONDICIÓN	El administrador o almacenero debe estar autenticado en el sistema.
ESCENARIO BÁSICO	<p>El caso de uso comienza cuando el administrador o almacenero requiere registrar un ingreso de equipos electrónicos al sistema.</p> <ol style="list-style-type: none"> <li>1. El usuario selecciona la opción de registro de ingreso de equipos electrónicos.</li> <li>2. El sistema despliega la interfaz de registro de nuevo ingreso de equipos electrónicos.</li> <li>3. El usuario ingresa los datos requeridos de ingreso de equipos electrónicos.</li> <li>4. El usuario registra la empresa que le provee.</li> <li>5. El usuario selecciona guardar o cancelar los datos ingresados.</li> <li>6. Despliega alerta de conformidad.</li> <li>7. El sistema despliega interfaz de garantía de equipos ingresados.</li> </ol>

	<ol style="list-style-type: none"> <li>8. El sistema despliega interfaz de consultas o detalles de equipos electrónicos ingresados.</li> <li>9. El sistema emitirá actas de ingreso de equipos electrónicos ingresados.</li> </ol>
<b>ESCENARIOS ITERATIVOS</b>	
<b>ALTERNATIVA - 1</b>	Si en el escenario 5 del escenario básico el sistema encuentra datos que no concuerdan con los requeridos, entonces el sistema despliega alertas que no fueron introducidos correctamente.
<b>POST-CONDICION</b>	<ol style="list-style-type: none"> <li>1. Los datos ingresados están almacenados en la Base de Datos.</li> <li>2. Automatiza consultas e informaciones.</li> <li>3. Muestra consultas o informaciones de garantías.</li> <li>4. Emite actas de ingreso</li> </ol>

**Tabla 11:** Ingresos de equipos electrónicos

**Fuente:** [Elaboración Propia]

<b>CASO DE USO</b>	<b>SALIDA DE EQUIPOS ELECTRÓNICOS</b>
<b>ACTOR</b>	Administrador, almacenero, personal
<b>PRE-CONDICIÓN</b>	El administrador o almacenero debe estar autenticado en el sistema.
<b>ESCENARIO BÁSICO</b>	<p>El caso de uso comienza cuando el administrador o almacenero requiere registrar una salida de equipos electrónicos al sistema a secciones que tiene la intuición.</p> <ol style="list-style-type: none"> <li>1. El usuario selecciona la opción de registro de salida de equipos electrónicos.</li> <li>2. El sistema despliega la interfaz de registro de nueva salida de equipos electrónicos.</li> <li>3. El usuario ingresa los datos requeridos para la salida de equipos electrónicos.</li> <li>4. El usuario registra la solicitud de salida requerida por el personal de la institución.</li> <li>5. El usuario registra la sección y la unidad que será destinado el equipo electrónico.</li> <li>6. El usuario selecciona guardar o cancelar los datos ingresados.</li> <li>7. Despliega alerta de conformidad.</li> <li>8. El sistema despliega interfaz de referencia o información de garantía de equipos salidos.</li> </ol>

	<p>9. El sistema despliega interfaz de consultas o detalles de equipos electrónicos salidos.</p> <p>10.El sistema emitirá actas de entrega de equipos electrónicos ingresados.</p>
<b>ESCENARIOS ITERATIVOS</b>	
<b>ALTERNATIVA - 1</b>	Si en el escenario 6 del escenario básico el sistema encuentra datos que no concuerdan con los requeridos, entonces el sistema despliega alertas que no fueron introducidos correctamente.
<b>POST-CONDICIÓN</b>	<ol style="list-style-type: none"> <li>1. Los datos ingresados están almacenados en la Base de Datos.</li> <li>2. Automatiza consultas e informaciones.</li> <li>3. Muestra consultas o informaciones de garantías.</li> <li>4. Emite actas entregas.</li> </ol>

**Tabla 12:** Salida de equipos electrónicos

**Fuente:** [Elaboración Propia]

<b>CASO DE USO</b>	<b>BAJAS DE EQUIPOS ELECTRÓNICOS</b>
<b>ACTOR</b>	Administrador, almacenero, personal
<b>PRE-CONDICIÓN</b>	El administrador o almacenero debe estar autenticado en el sistema.
<b>ESCENARIO BASICO</b>	<p>El caso de uso comienza cuando el administrador o almacenero requiere registrar una baja de equipo electrónico dañado al sistema.</p> <ol style="list-style-type: none"> <li>1. El usuario selecciona el equipo que ya no cuenta con garantía y está sin solución de arreglo, registra el equipo dañado.</li> <li>2. El sistema despliega la interfaz de registro de nueva baja de equipo electrónico.</li> <li>3. El usuario ingresa los datos requeridos para la baja.</li> <li>4. El usuario registra la solicitud de soporte técnico.</li> <li>5. El usuario registra los problemas técnicos que tiene el equipo electrónico.</li> <li>6. El usuario selecciona guardar o cancelar los datos ingresados.</li> <li>7. Despliega alerta de conformidad.</li> <li>8. El sistema despliega interfaz de consulta de referencia o información de los equipos eléctricos.</li> <li>9. El sistema emitirá informe de baja.</li> </ol>



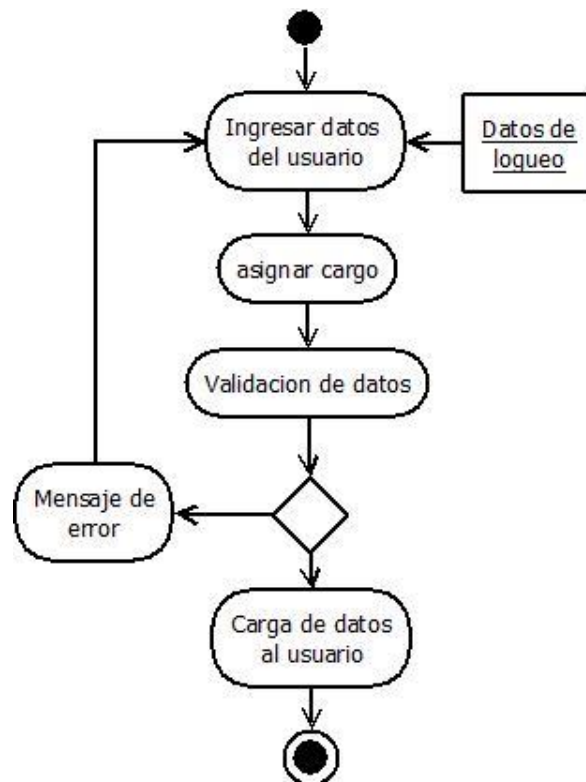
ESCENARIOS ITERATIVOS	
ALTERNATIVA - 1	Si en el escenario 6 del escenario básico el sistema encuentra datos que no concuerdan con los requeridos, entonces el sistema despliega alertas que no fueron introducidos correctamente.
POST-CONDICIÓN	<ol style="list-style-type: none"> <li>1. Los datos ingresados están almacenados en la Base de Datos.</li> <li>2. Automatiza consultas e informaciones.</li> <li>3. Emite informe de baja.</li> </ol>

**Tabla 13:** Salida de equipos electrónicos

**Fuente:** [Elaboración Propia]

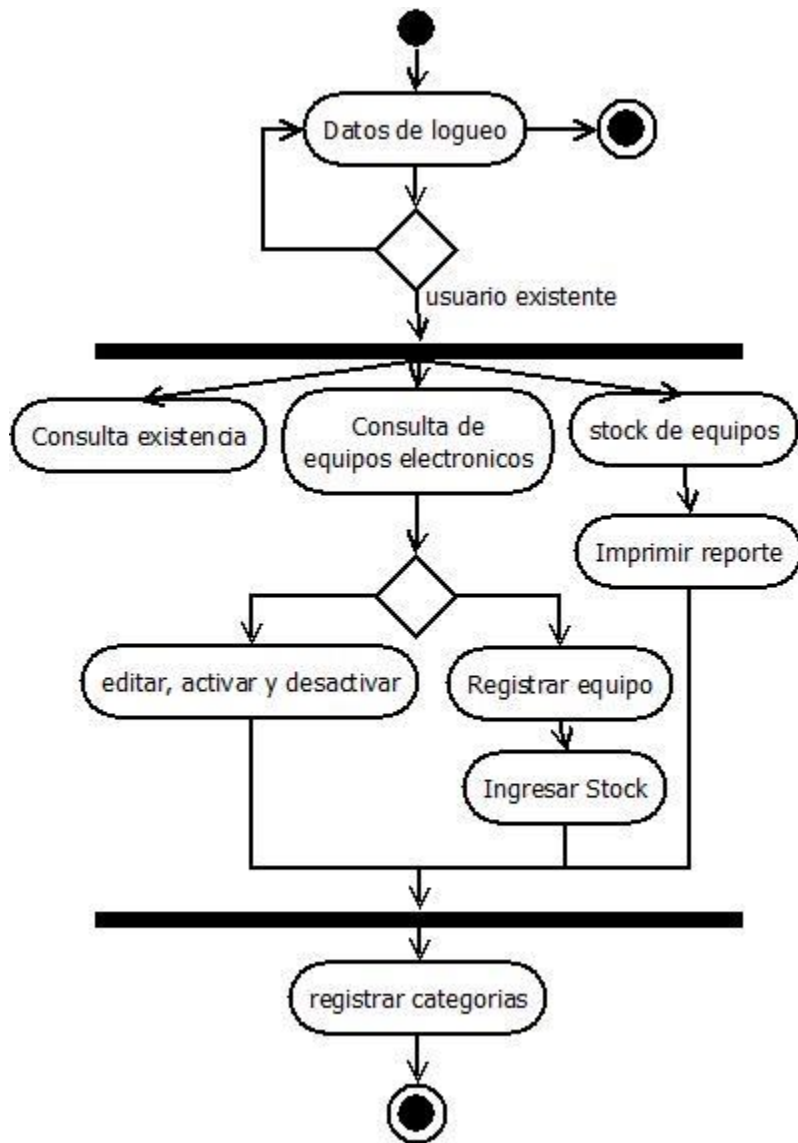
### 3.5.2.7. Diagrama de Actividades

El diagrama de actividades se empleara para definir la secuencia de pasos lógicos que se deben seguir para cumplir con una determinada función en el sistema.

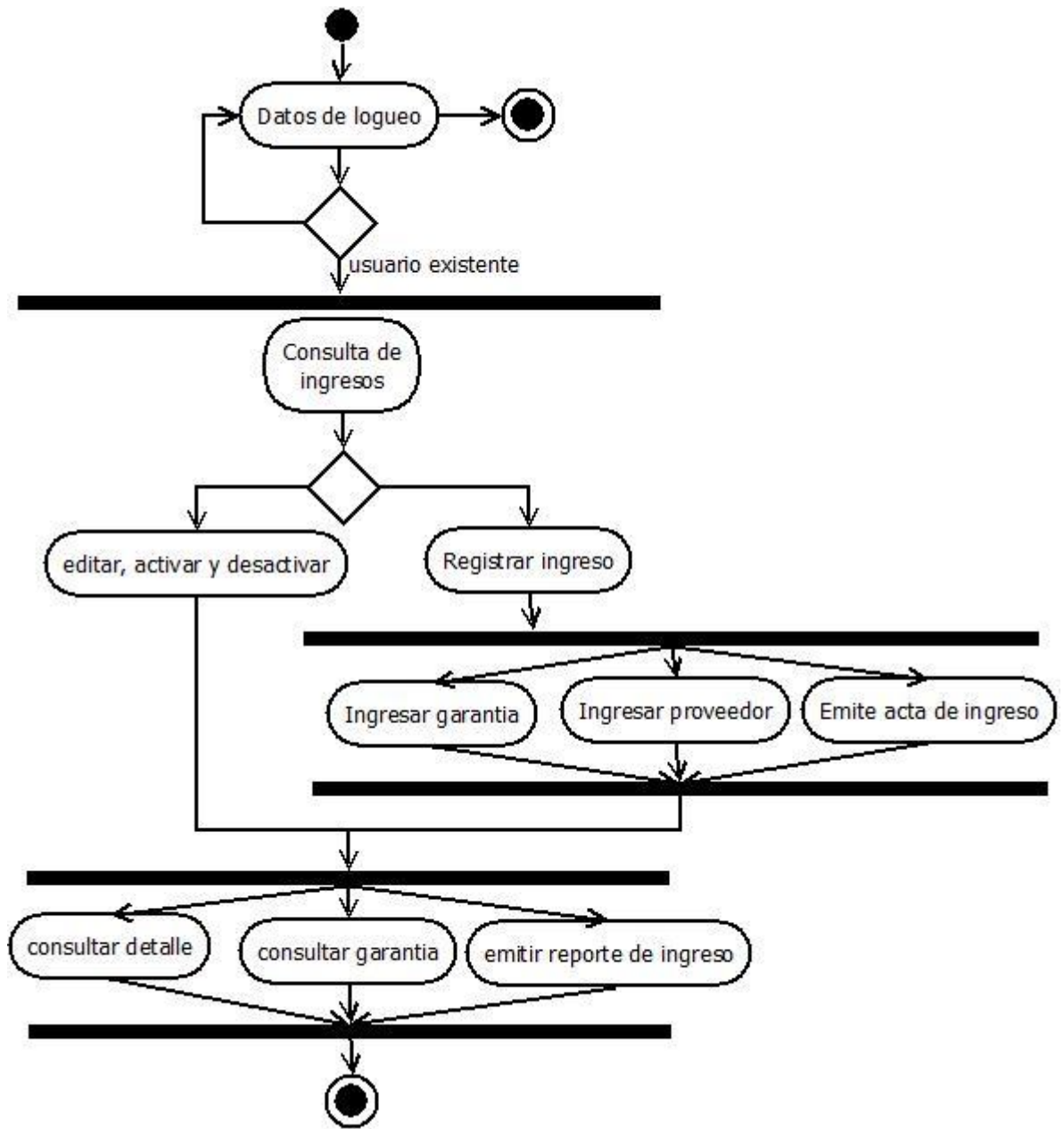


**Fig. 13** Diagrama de actividades registro de usuarios en el sistema

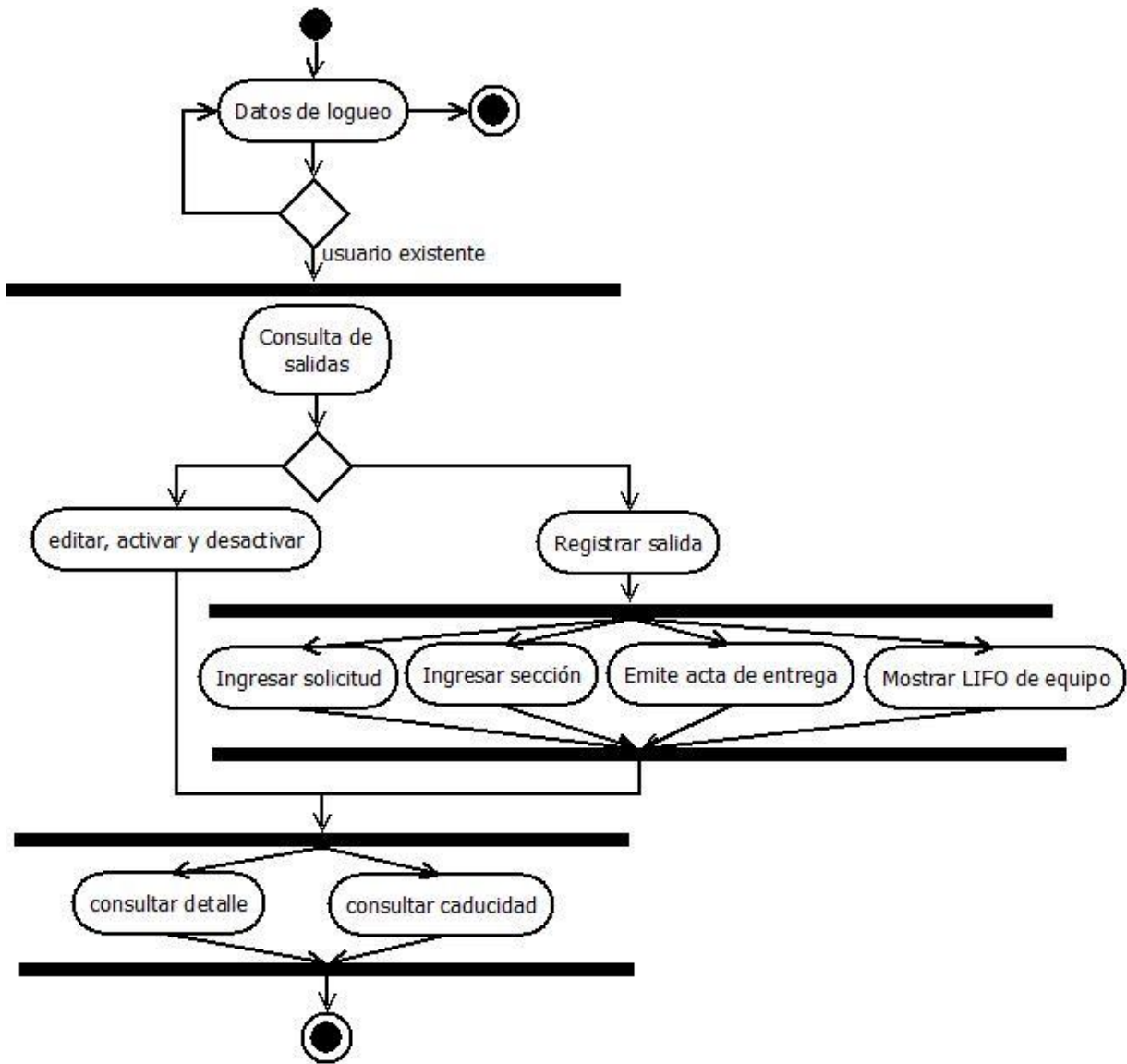
**Fuente:** [Elaboración Propia]



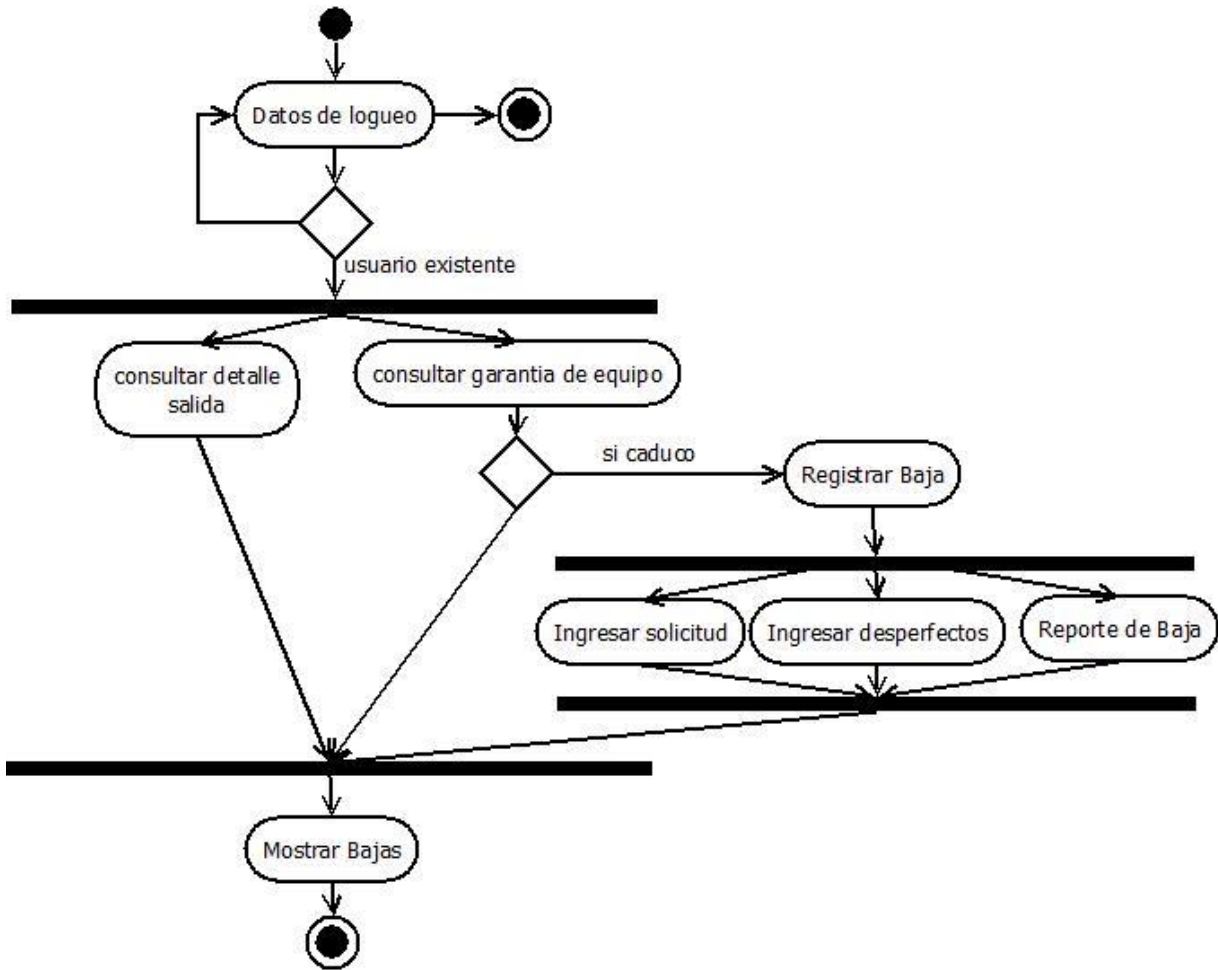
**Fig. 14** Diagrama de actividades de administración de equipos electrónicos  
**Fuente:** [Elaboración Propia]



**Fig. 15** Diagrama de actividades de ingresos de equipos electrónicos  
**Fuente:** [Elaboración Propia]



**Fig. 16** Diagrama de actividades de salidas de equipos electrónicos  
**Fuente:** [Elaboración Propia]

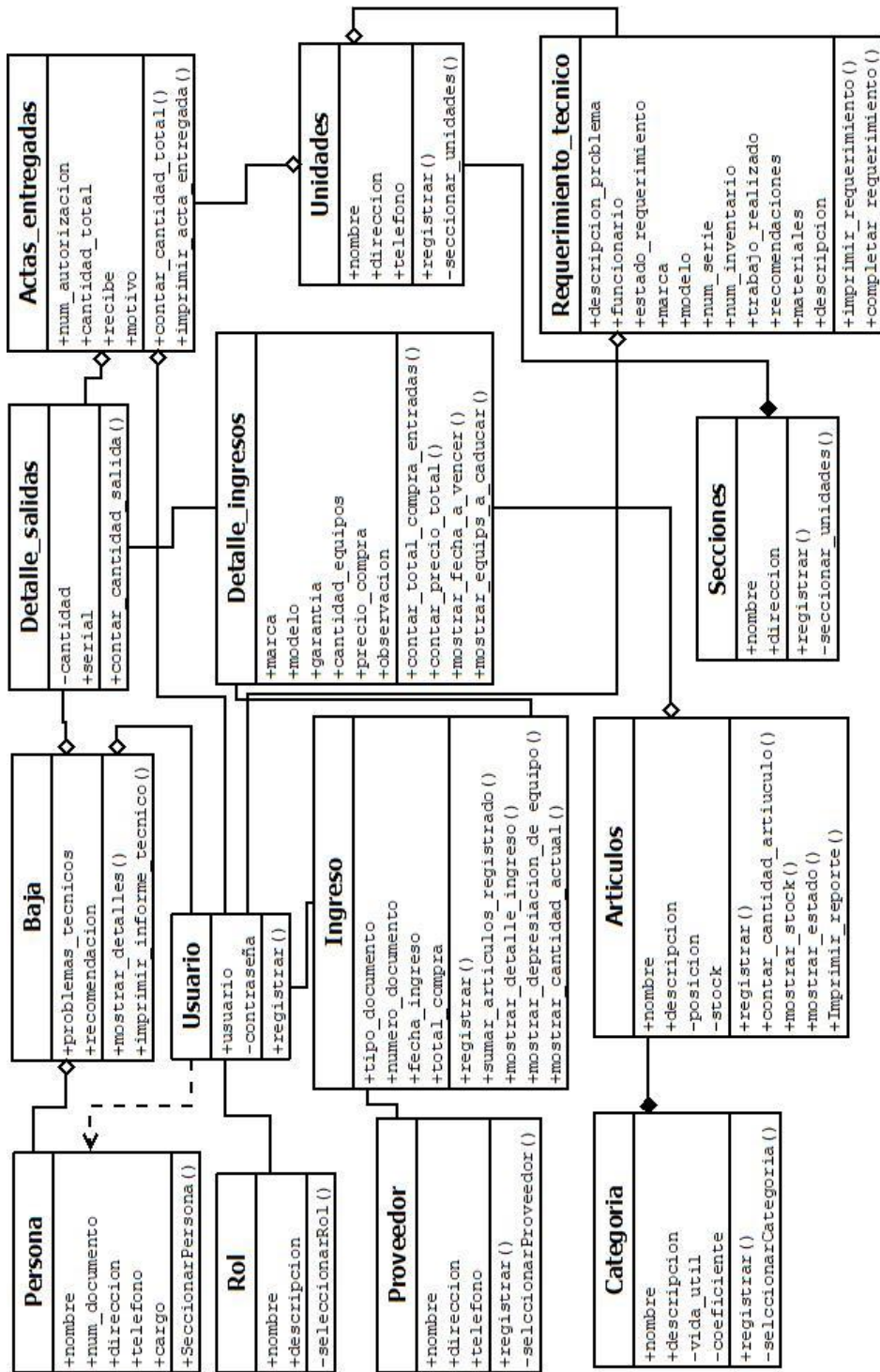


**Fig. 17** Diagrama de actividades de bajas de equipos electrónicos

**Fuente:** [Elaboración Propia]

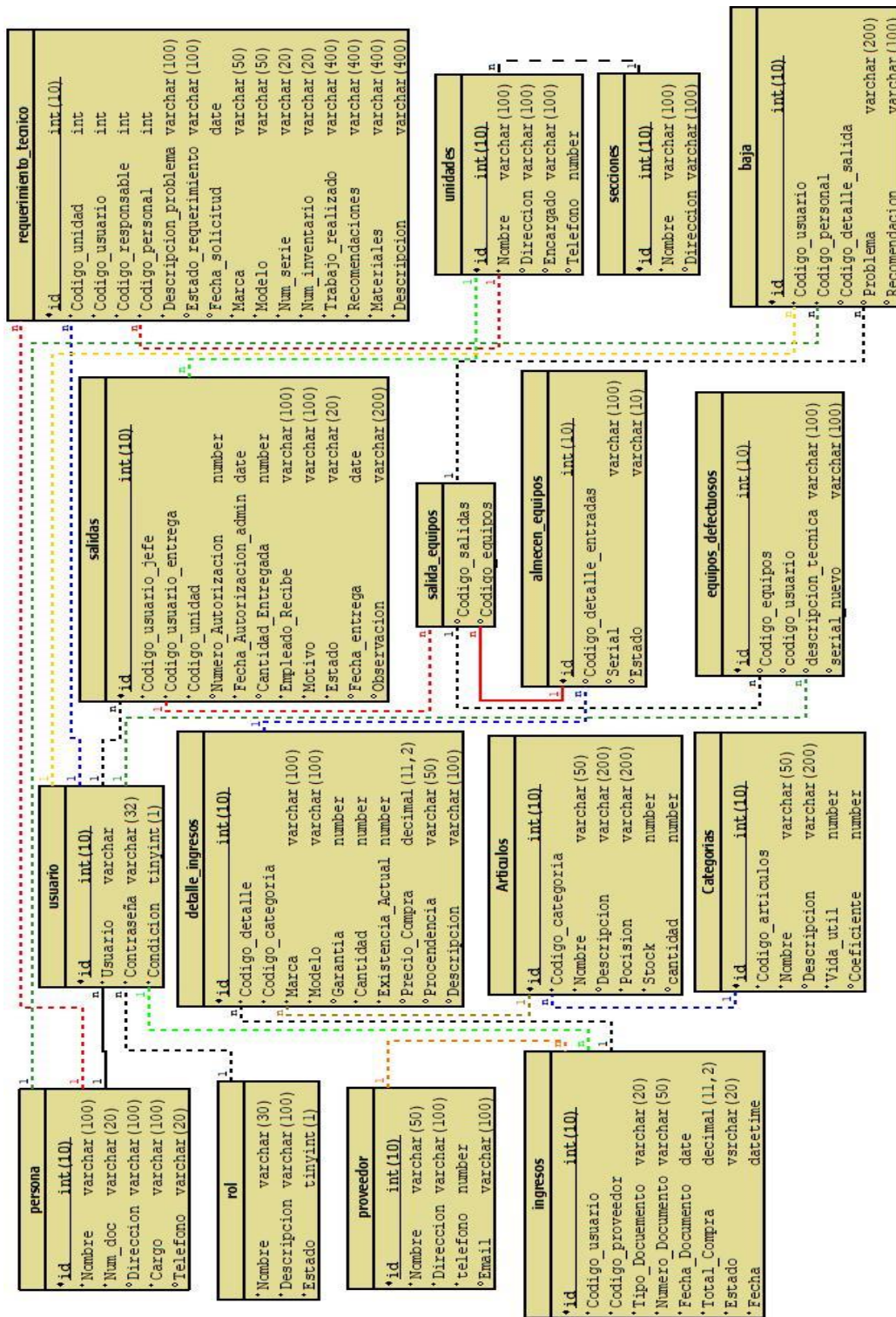
### 3.5.2.8. Diagrama de Clases

Este diagrama especifica cómo se encuentran relacionadas los contenidos del sistema, muestra una colección de elementos de modelado declarativo, tales como clases, tipos y sus contenidos y relaciones. Una clase es un elemento importante dentro del contexto de un sistema, que puede tener información o datos valiosos y realizar acciones que sean necesarias dentro del funcionamiento del sistema.



**Fig. 18** Diagrama de clases  
**Fuente:** [Elaboración Propia]

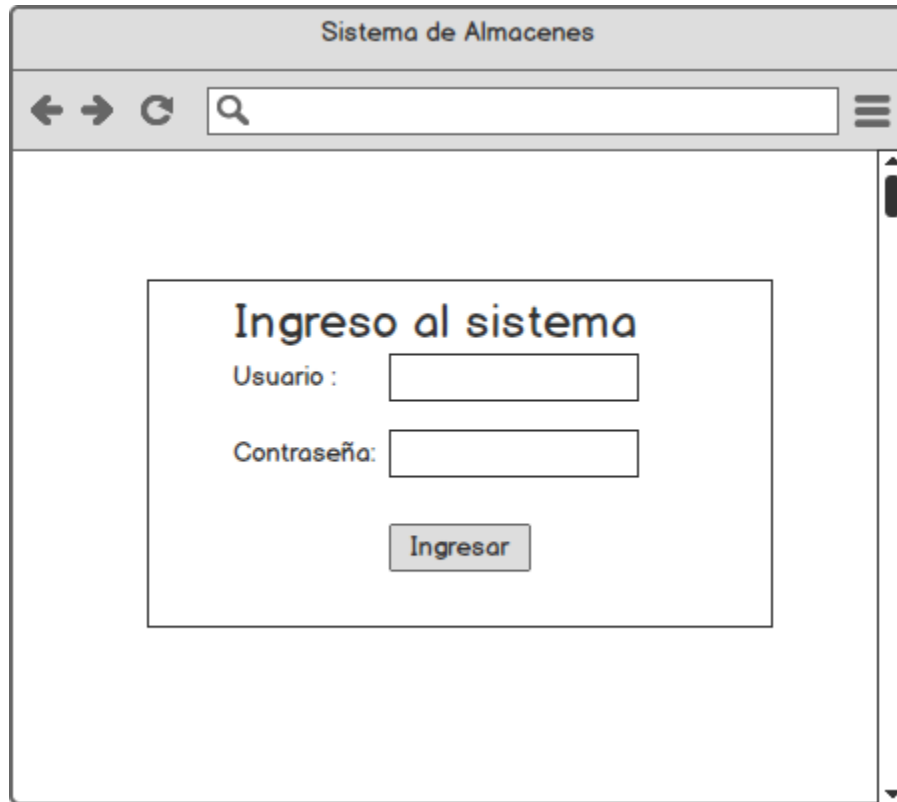
### 3.5.2.9. Modelo Físico



**Fig.19** Modelo físico  
Fuente: [Elaboración Propia]

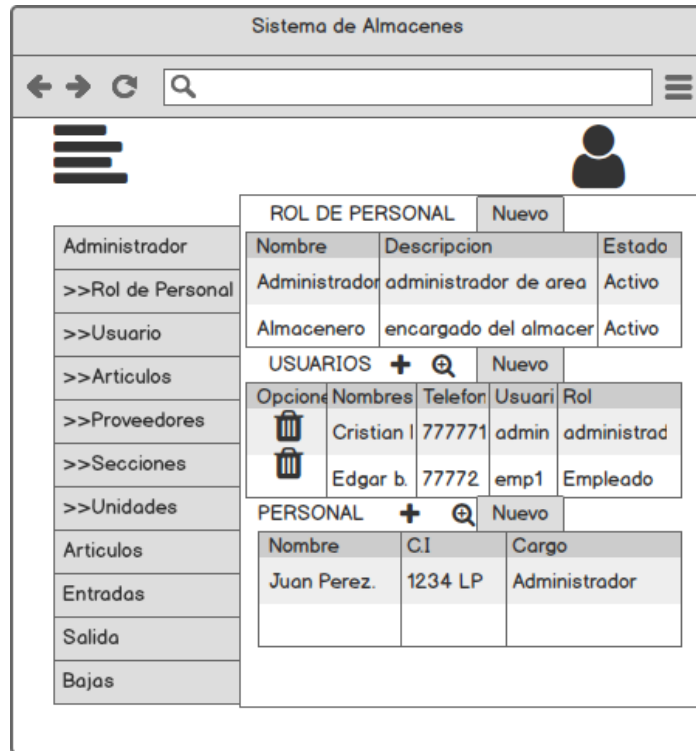
### 3.5.2.10. Diagrama de Presentación

Los diagramas de presentación, que se muestra a continuación, permiten visualizar el resultado final de las interfaces que tendrá el sistema cumpliendo con los requerimientos planteados.



**Fig. 20** Diagrama de presentación Ingreso al Sistema  
**Fuente:** [Elaboración Propia]





**Fig. 21** Diagrama de presentación de administración  
**Fuente:** [Elaboración Propia]



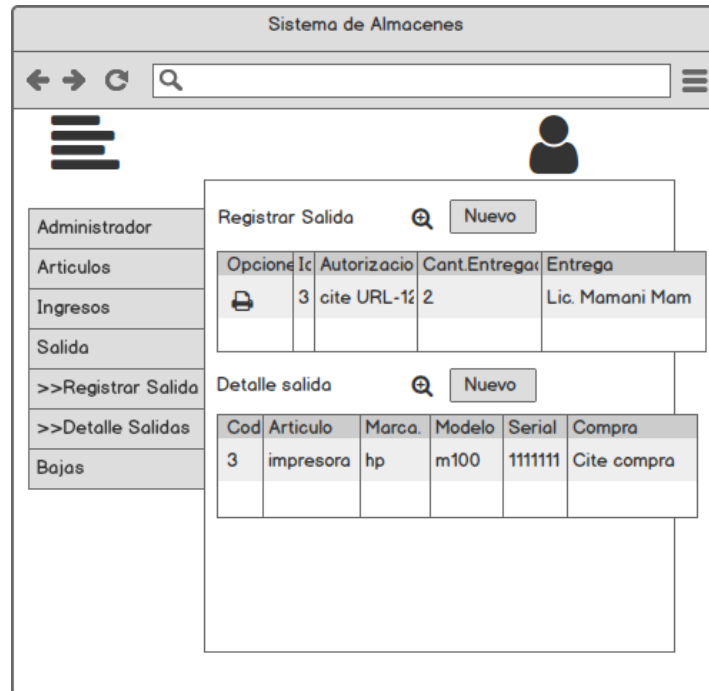
**Fig. 22** Diagrama de presentación de registro de usuario  
**Fuente:** [Elaboración Propia]



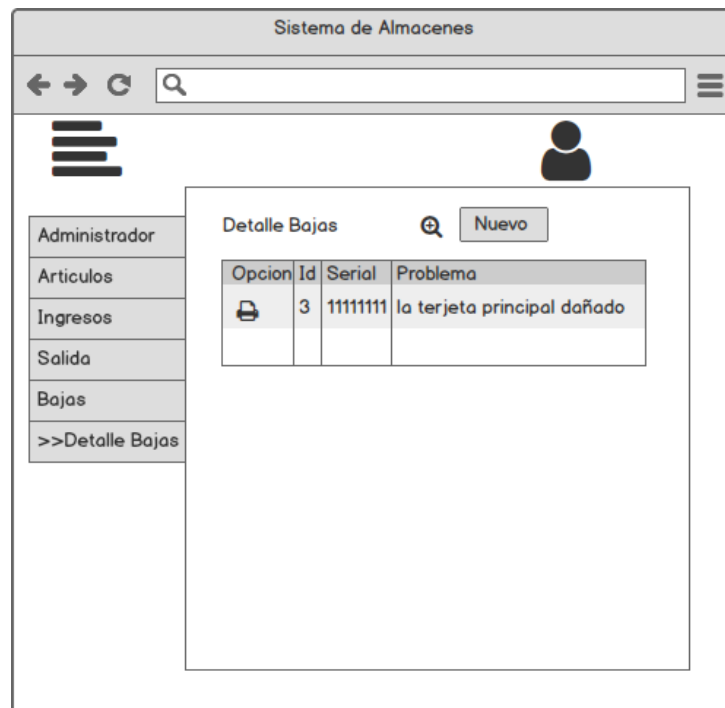
**Fig. 23** Diagrama de presentación de equipos electrónicos  
**Fuente:** [Elaboración Propia]



**Fig. 24** Diagrama de presentación ingresos de equipos electrónicos  
**Fuente:** [Elaboración Propia]



**Fig. 25** Diagrama de presentación salida de equipos electrónicos  
**Fuente:** [Elaboración Propia]



**Fig. 26** Diagrama de presentación bajas de equipos electrónicos  
**Fuente:** [Elaboración Propia]

## **3.6. POST-JUEGO**

Durante esta última etapa se trabajó en las políticas de seguridad para el sistema. Se obtuvieron matrices se realizó el diseño de ayuda para los usuarios.

### **3.6.1. Seguridad de sistema**

#### **3.6.1.1. Políticas de seguridad en cuanto a usuarios**

Súper Administrador, puede acceder a la información de toda la base de datos, instanciar, modificar, administrar la información y las secciones de todos los usuarios y encargados de las copias de seguridad del sistema.

Administrador, responsable de administrar el sistema tanto como modificación y eliminación de datos, es el encargado de asignar usuarios, roles de ingreso al sistema, además es el encargado de verificación de niveles de existencia y stock de equipos electrónicos.

Almacenero, que solo registra el ingreso, salida y bajas de equipos electrónicos para luego imprimir informes y entregar a quien corresponde para su correcto respaldo de inventarios, además es el encargado de verificación de niveles de existencia y stock de equipos electrónicos.

#### **3.6.1.2. Políticas de seguridad en cuanto a acceso al sistema**

El sistema verifica la autenticidad del usuario y su contraseña.

Los usuarios deben de guardar de forma segura tanto su contraseña como su nombre de usuario, en caso de olvido se debe de solicitar al administrador su olvido de contraseña para que le asigne otro.

El acceso a la base de datos es restringido para usuarios no autorizados mediante contraseña y nombre de usuario en el servidor.

### **3.6.1.3. Políticas en cuanto al sistema operativo**

El acceso al sistema deberá estar restringido con una contraseña creada por el administrador de sistema operativo.

Si el sistema operativo es Windows debe de tener instalado un antivirus y este debe estar actualizado.

### **3.6.1.4. Políticas de seguridad en cuanto a backups**

Realizar copias de seguridad a la base de datos periódicamente.

Los dispositivos de almacenamiento que contengan el backup deben estar etiquetados con fecha, hora y el responsable, además de ser guardados en un lugar dentro de la institución.

## **3.6.2. Implantación**

### **Requerimientos técnicos**

Las características mínimas de hardware necesarias para la implantación del presente sistema son:

- Microprocesador Intel Core(TM) i3 de 2.00Hz o superior
- Memoria Ram de 4 Gb o superior
- Disco duro de 80 Gb
- Sistema operativo Debian
- Servidor Apache/2.4
- PHP versión 7.1.10
- Mysql 5.0.12

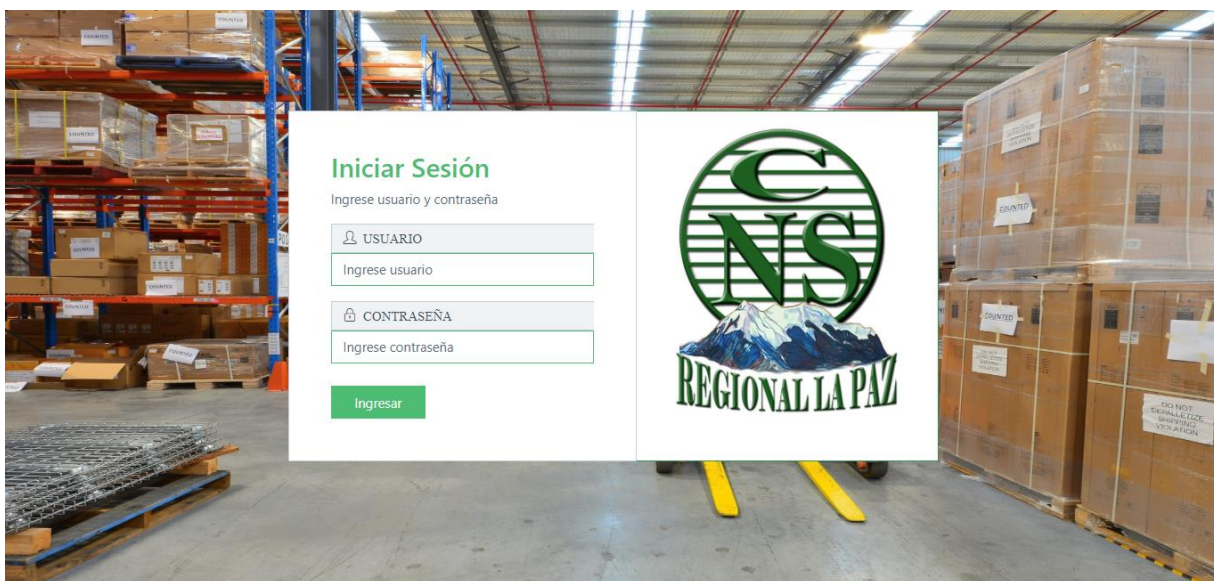
La implantación del sistema implica una labor de capacitación no solo respecto al manejo del sistema sino también al diseño de instrumentos descriptivos aplicando conceptos básicos de base de datos.

La estrategia para implementar esta ayuda fue elaborar un manual de usuario.

### 3.7. MODELO DE IMPLEMENTACIÓN DEL PROGRAMA

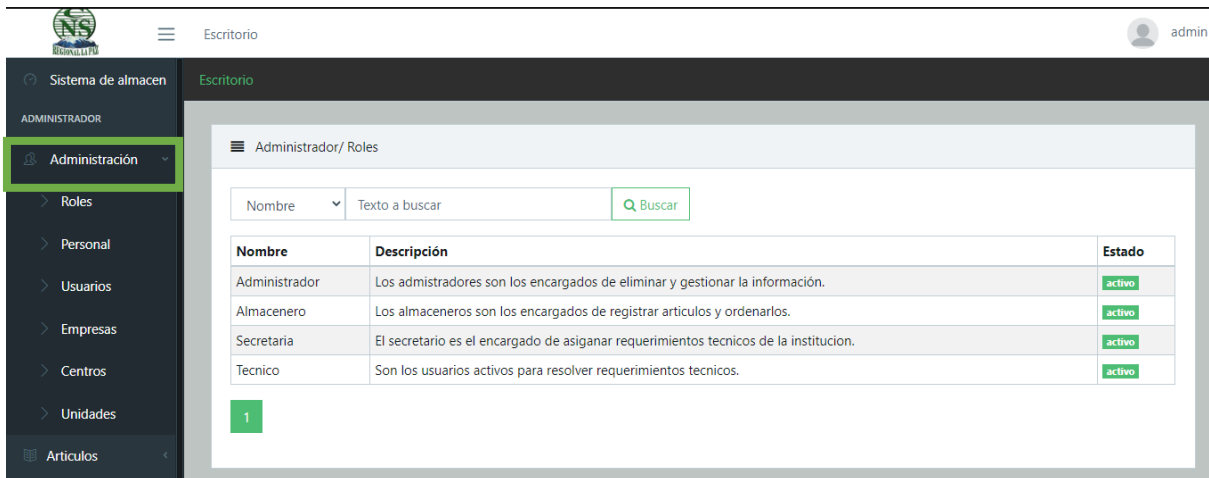
#### 3.7.1. Diseño de interfaces

Las siguientes figuras modelan los diseños de interfaz de usuarios de los diferentes módulos, para el ingreso al sistema si la información introducida en la ventana es correcta ingresará al sistema caso contrario dará error y posteriormente el sistema habilita los menús correspondientes, dependiendo al rol de los usuarios.



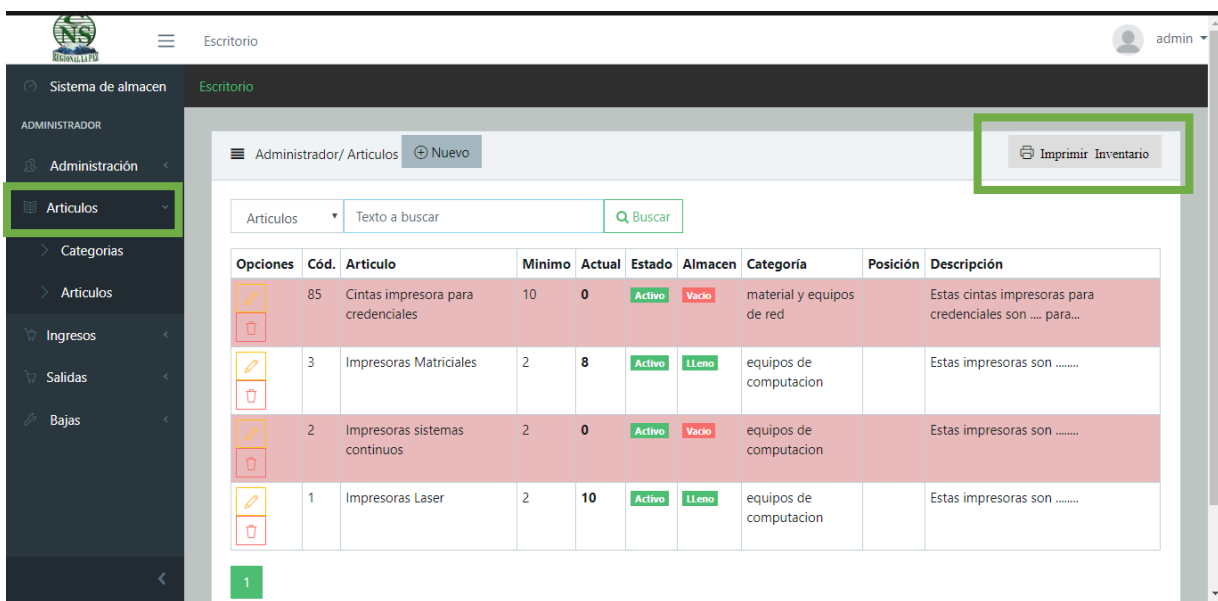
**Fig. 27** Ingreso al sistema  
**Fuente:** [Elaboración Propia]

El módulo de administración es donde mostrará todos los datos e informaciones de referencia el cual interactuará con el sistema.



**Fig. 28** Módulo de administración  
Fuente: [Elaboración Propia]

En el módulo de artículos y botón artículos es donde se hará la gestión de equipos para la verificación de niveles de existencia y stock de los equipos electrónicos por lo cual mostrará alertas de estados de control de almacén.



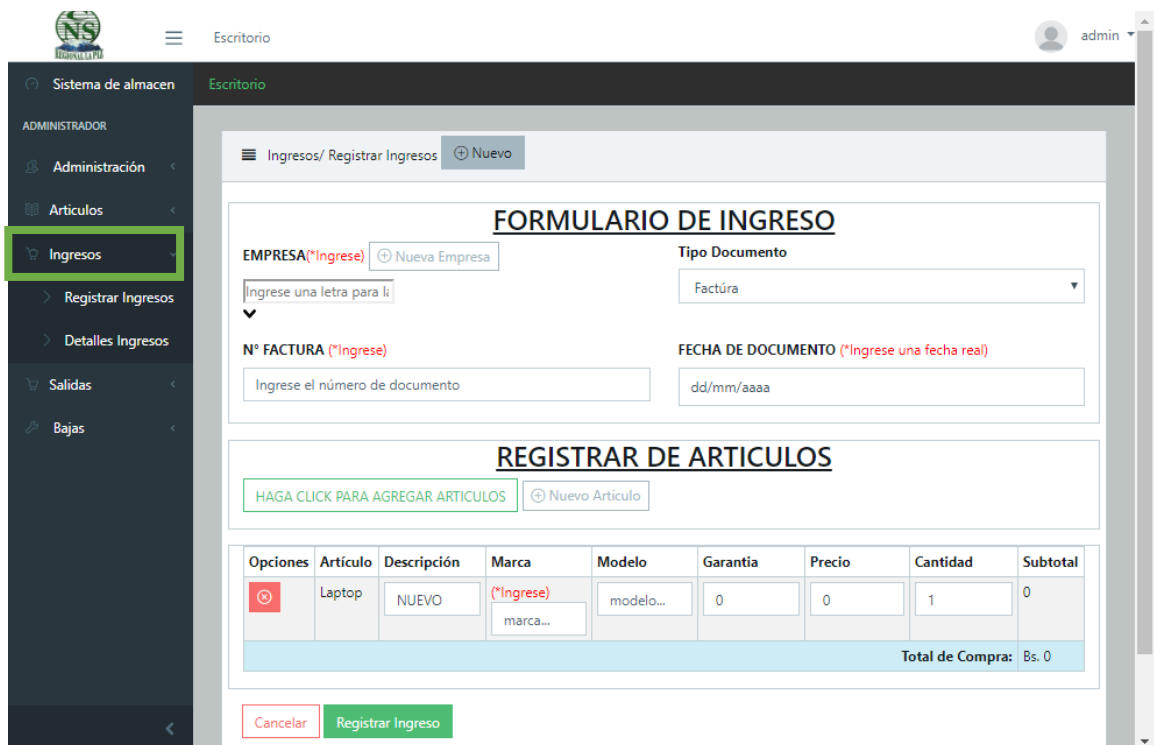
**Fig. 29** Registro de artículos  
Fuente: [Elaboración Propia]

Imprime en PDF los inventarios generales al presionar en el botón imprimir inventario (ver figura 29), donde se mostrará todos los artículos registrados y la cantidad de equipos electrónicos sea mayor a cero.



**Fig. 30 Impresión de inventarios generales**  
**Fuente:** [Elaboración Propia]

En el módulo de ingresos se realiza el registro de ingreso de equipos electrónicos, después de registrar se sumara a artículos, en caso de que el artículo no esté almacenado habrá un botón para agregar artículos. Al registrar un ingreso se emitirá un acta de ingreso para que el encargado de almacenes reporte a activos.



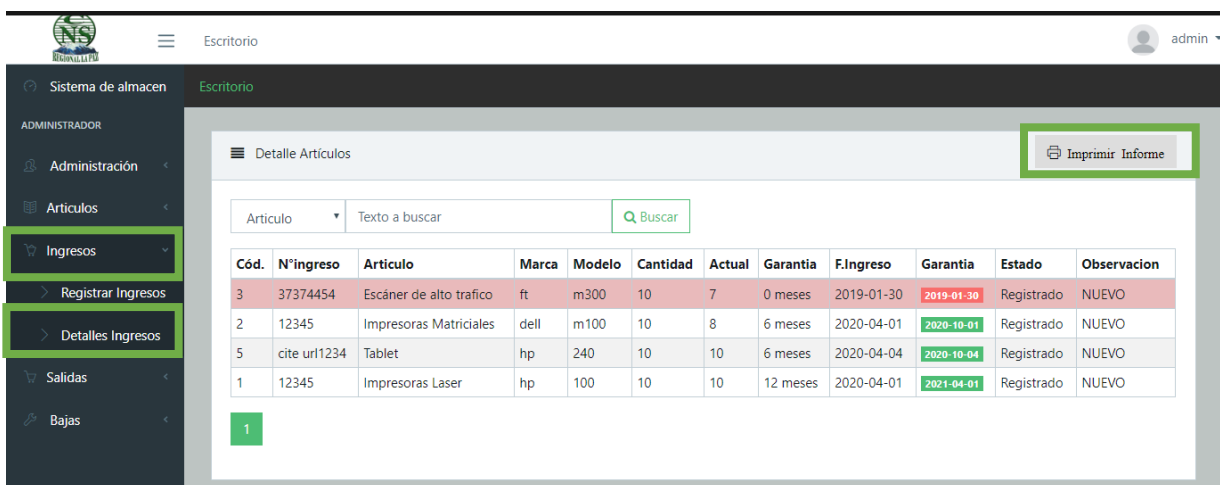
**Fig. 31 Registro de ingresos de equipos electrónicos**  
**Fuente:** [Elaboración Propia]





**Fig. 32** Acta de ingreso de equipos electrónicos  
 Fuente: [Elaboración Propia]

En el módulo de ingreso y botón detalle ingresos, realiza el listado detallado de equipos electrónicos ingresados, donde detalla cada artículo con su garantía, cantidad actual de ingresos, en caso de que la cantidad este en cero se elimina del listado para el informe de equipos electrónicos detallados por compra.



**Fig. 33** Detalle de Ingresos de equipos electrónicos  
 Fuente: [Elaboración Propia]

En el módulo de salidas se registrará los equipos electrónicos que se requieran y se restara de artículos para tener datos exactos en los niveles de existencia de equipos electrónicos, después de registrar el almacenero emite su acta de entrega para su respaldo de salida.

**Fig. 34** Registro de salida de equipos electrónicos  
Fuente: [Elaboración Propia]

ARTICULO	PROCEDENCIA	MARCA	MODELO	SERIAL	GARANTIA
IMPRESORAS MATRICIALES	CHINA	EPSN	M200	22222	2021-01-07
IMPRESORAS MATRICIALES	CHINA	EPSN	M200	1111111	2021-01-07

**Fig. 35** Acta de entrega de equipos electrónicos  
Fuente: [Elaboración Propia]

Al presionar el botón de seccionar artículos de la figura 34, mostrará un modal de todos los equipos electrónicos que estén a punto de caducar para sacar de almacenes por temas de depreciación de equipos.

Opción	Cód.Ingr.	Procedencia	Nombre	Marca	Modelo	Serial	Garantía
<input type="checkbox"/>	1	china	Impresoras Matriciales	epson	m200	666666	2021-01-07
<input type="checkbox"/>	1	china	Impresoras Matriciales	epson	m200	555555	2021-01-07
<input type="checkbox"/>	1	china	Impresoras Matriciales	epson	m200	1111111	2021-01-07
<input type="checkbox"/>	2	china	Impresoras Laser	epson	m100	975311368aab	2021-07-09
<input type="checkbox"/>	2	china	Impresoras Laser	epson	m100	637284ute92	2021-07-09
<input type="checkbox"/>	2	china	Impresoras Laser	epson	m100	135791113aa	2021-07-09
<input type="checkbox"/>	2	china	Impresoras Laser	epson	m100	378573h3k4	2021-07-09
<input type="checkbox"/>	2	china	Impresoras Laser	epson	m100	24681912bb	2021-07-09

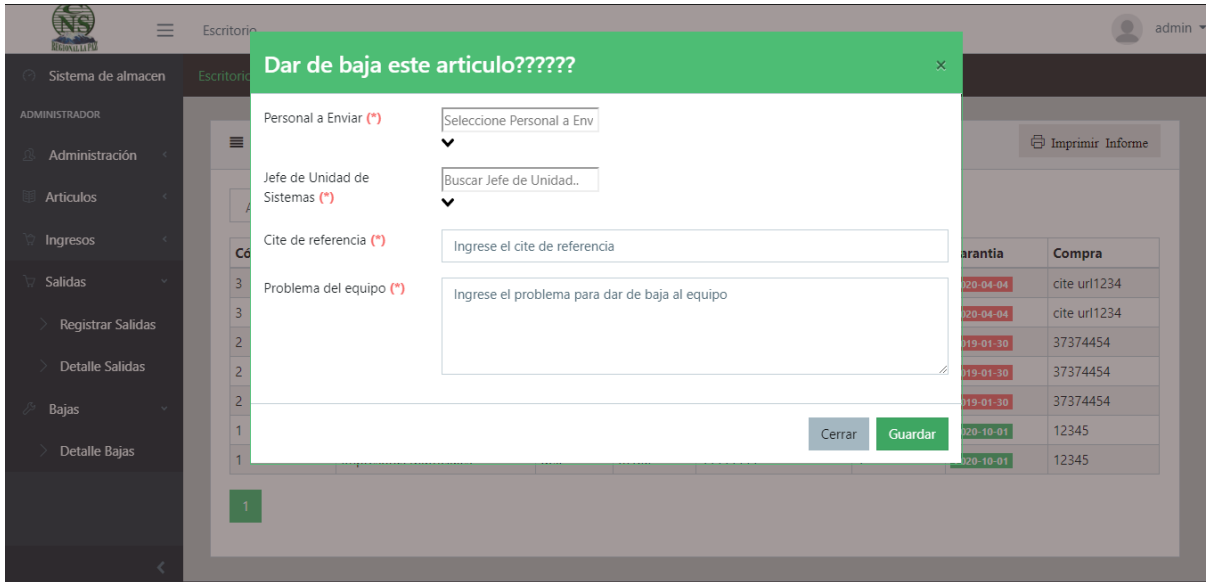
**Fig. 36** Modal de selección de artículos de salida  
Fuente: [Elaboración Propia]

En el módulo de salidas y botón detalle salidas se detalla el listado de todos los equipos electrónicos sacados de almacenes el cual mostrará la garantía de cada equipo entregado.

Cód.Salida	Artículo	Marca	Modelo	Serial	Cantidad	Garantía	Compra
3	Grímpador de red	simsom		244	1	2020-04-04	cite url1234
3	Grímpador de red	simsom		243	1	2020-04-04	cite url1234
2	Escáner de alto trafico	ft	m300	33333333333345	1	2019-01-30	37374454
2	Escáner de alto trafico	ft	m300	2222222233	1	2019-01-30	37374454
2	Escáner de alto trafico	ft	m300	11111111	1	2019-01-30	37374454
1	Impresoras Matriciales	dell	m100	22222222	1	2020-10-01	12345
1	Impresoras Matriciales	dell	m100	11111111	1	2020-10-01	12345

**Fig. 37** Detalle de salida de equipos electrónicos  
Fuente: [Elaboración Propia]

Si en el registro de detalle de salidas, se verifica que la garantía haya vencido y que el equipo no tenga una solución técnica se registra la baja del equipo electrónico, generando una impresión de informe técnico de baja.



**Fig. 38** Registro de baja de equipos electrónicos

Fuente: [Elaboración Propia]

### 3.8. PRUEBAS DE CALIDAD ISO 25010

En la documentación oficial de laravel facilita las pruebas de software usando la ISO 25010

#### 3.8.1. Pruebas de http

Laravel proporciona una API fluida para realizar solicitudes HTTP a su aplicación y examinar la salida. Como también proporciona varios ayudantes para probar las API JSON y sus respuestas como ser: json, get, post, put, patch y delete métodos que se pueden utilizar para emitir solicitudes con diferentes verbos HTTP.

Después de concluir con el software se realizó la prueba de http para protección contra ataques de CCRF y ataques JSON aplicando la prueba de salida.

1. Si introducimos en la url un get de obtención de datos con la extensión persona <http://127.0.0.1:8000/persona> según JSON nos devolverá la siguiente consulta en el navegador.

```

← → 127.0.0.1:8000/persona
{"pagination":{"total":3747,"current_page":1,"per_page":20,"last_page":188,"from":1,"to":20},"personas":{"current_page":1,"data":[{"id":4230,"nombre":"JENNY JOHANA ARMAZA SAAVEDRA","num_doc":"4362351 LP","cargo":"TRAB SOCIAL","direccion":"AV. RAFAEL BALLIVIAN # 1953","telefono":"68023938","email":"jenny_johana@hotmail.com","created_at":null,"updated_at":null},{"id":4229,"nombre":"ROSA LOURDES GUARDIA MIRANDA","num_doc":"2714682 LP","cargo":"TRAB SOCIAL","direccion":"Z/ VILLA SAN ANTONIO # 50","telefono":"71510787","email":"rocioguardiamiranda@gmail.com","created_at":null,"updated_at":null},{"id":4228,"nombre":"MARY CRUZ ZAMBRANA SARMIENTO","num_doc":"4243694 LP","cargo":"TRAB SOCIAL","direccion":"Z/ BAJO LLOJETA # 120","telefono":"65111761","email":"maricruzambrana@gmail.com","created_at":null,"updated_at":null},{"id":4227,"nombre":"WENDY KARINA OLIVARES COSS","num_doc":"4268436 LP","cargo":"TRAB SOCIAL","direccion":"Z/ACHUMANI # 17","telefono":"72521254","email":"wendyolivarescos@gmail.com","created_at":null,"updated_at":null},{"id":4226,"nombre":"GISELA MARIA HIGUERAS VALDIVIA","num_doc":"3479441 LP","cargo":"TRAB SOCIAL","direccion":"Z/ PAMPAHASI # 184","telefono":"72537767","email":"giselahigieras@gmail.com","created_at":null,"updated_at":null},{"id":4225,"nombre":"MARIA ELENA TORREZ SALCEDO","num_doc":"4806845 LP","cargo":"TRAB SOCIAL","direccion":"Z/ ALTO SANTIAGO DE LACAY # 5","telefono":"73271005","email":"elenatorrez@hotmail.com","created_at":null,"updated_at":null},{"id":4224,"nombre":"ALINA OFELIA RAMOS MACIAS","num_doc":"2638371 LP","cargo":"TRAB SOCIAL","direccion":"Z/SOPOCACHI # 130","telefono":"76225571","email":"alinaramos1112@gmail.com","created_at":null,"updated_at":null},{"id":4223,"nombre":"GRETZEL LUCIA TARIFA ARISPE","num_doc":"4882497 LP","cargo":"TRAB SOCIAL","direccion":"Z/ CAJA FERROVIARIA # 2125","telefono":"70596605","email":"irinal3dastan@gmail.com","created_at":null,"updated_at":null},{"id":4222,"nombre":"MARCELO GALINDO TORREZ","num_doc":"9108830 LP","cargo":"PORTERO","direccion":"Z/SANTIAGO II C/ 16 # 1343","telefono":"61171247","email":"marcelogalindotorrez@gmail.com","created_at":null,"updated_at":null},{"id":4221,"nombre":"JAIIME TANCARA SURCI","num_doc":"2679690 LP","cargo":"PORTERO","direccion":"Z/SAN JUAN 24 DE JUNIO # 1394 B","telefono":"73543167","email":"","created_at":null,"updated_at":null},{"id":4220,"nombre":"YONANTAN FELIX PEREZ FLORES","num_doc":"576915 OR","cargo":"PORTERO","direccion":"Z/CALLE BENI # 9","telefono":"73219347","email":"","created_at":null,"updated_at":null},{"id":4219,"nombre":"WILLY TICONA CONDORI","num_doc":"7091369 LP","cargo":"PORTERO","direccion":"RESD. EN OQUETITTI","telefono":"73576967","email":"","created_at":null,"updated_at":null},{"id":4218,"nombre":"JOSE LUIS CONDARCO ALCON","num_doc":"2480202 LP","cargo":"ENCARGADO DE PORTERIA","direccion":"Z/ ALTO SAN PEDRO # 1600","telefono":"75808480","email":"","created_at":null,"updated_at":null}, {"id":4217,"nombre":"RODRIGO ARISMENDI SIRPA","num_doc":"5940252 LP","cargo":"PORTERO","direccion":"Z/VILLA FATIMA # 282","telefono":"74030749","email":"rodrigo_aris34@hotmail.com","created_at":null,"updated_at":null}, {"id":4216,"nombre":"IRMA MANAMACU COCARICO","num_doc":"6785380 LP","cargo":"POLICIA DE SEGURIDAD","direccion":"Z/ VILLA INGENIO C/MONTERO HOYOS NRO 4252","telefono":"71226478","email":"","created_at":null,"updated_at":null}, {"id":4215,"nombre":"MARCELO ANDRES PARY CRUZ","num_doc":"9099834 LP","cargo":"CLASE DE POLICIA","direccion":"Z/ SENKATA C/NUEVA ESPERANZA NRO 3000","telefono":"75081664","email":"","created_at":null,"updated_at":null}, {"id":4214,"nombre":"RENE BENITO CALANI CONDORI","num_doc":"7050591 LP","cargo":"CLASE DE POLICIA","direccion":"Z/ SAN LUIS SEGUNDO C/ OBLITAS NRO 1054","telefono":"71914834","email":"SISPLUBGAVER@GMAIL.COM","created_at":null,"updated_at":null}, {"id":4213,"nombre":"EDWIN CHOQUE CHURQUI","num_doc":"4962013 LP","cargo":"CLASE DE POLICIA","direccion":"Z VILLA NUEVA POTOSI C/ VILLAMIL DE RADA NRO

```

**Fig. 39 Pruebas de JSON en laravel**

**Fuente:** [Elaboración Propia]

El cual al devolver datos esto no debería ocurrir en modo producción por temas de seguridad así que laravel nos ofrece una condición que al momento que llame los datos de personas en la url se re direcciona a la página principal.

**if (!\$request->ajax()) return redirect('/');**

En este fragmento de código, la url no mostrará las extensiones que hagan las peticiones los JSON o JQUERY.

2. contra ataques de CSRF se asignará el siguiente código que brinda laravel para aplicaciones de interfaces seguras.

**<meta name="csrf-token" contenr={{ csrf token()}}>**

### 3.8.2. Pruebas del navegador

Laravel Dusk proporciona una API de prueba y automatización de navegador expresiva y fácil de usar. De forma predeterminada, Dusk no requiere que instale JDK o Selenium en su máquina.

Seguendo la documentación de laravel y el desarrollo de software para prueba de navegador de bajo nivel asociada al código fuente, se realiza pruebas de simular lo que hace un usuario con GoogleDrive.

```

<?php
namespace Tests\Browser;
use App\User;
use Tests\DuskTestCase;
use Laravel\Dusk\Chrome;
use Illuminate\Foundation\Testing\DatabaseMigrations;
class ExampleTest extends DuskTestCase
{
    public function testBasicExample()
    {

        $this->browse(function ($browser) {
            $browser->visit('/')
                ->type('usuario', 'admin')
                ->type('password', 'admin')
                ->click('buton[type="submit"]')
                ->assertPathIs('/contenido');
        });
    }
}

```

Al ejecutar el comando `php artisan dusk` entonces mostrará que el software está corriendo con un tiempo aproximado de 5 a 10s y consumo de memoria de 10.00 a 12.00MB.

### 3.9. METRICAS DE CALIDAD

#### 3.9.1. Técnica Web ISO 9126

El objetivo principal de calidad de software es alcanzar la calidad necesaria para satisfacer las necesidades del cliente. Se evalúan dos ámbitos: el producto final y los procesos. Estos ámbitos son conocidos como modelos de referencia. La norma ISO 9126 que hace referencia al modelo de calidad del producto del software y la ISO 14594 hace referencia a la calidad de evaluación del producto de software.

La calidad según esta norma web ISO 9126 puede ser perdida de acuerdo a los factores:

- Usabilidad

- Eficacia
- Flexibilidad
- Facilidad de uso
- Portabilidad

### 3.9.1.1. Usabilidad

La usabilidad consiste en la evaluación del esfuerzo necesario que el usuario invertirá para usar el sistema, en base a su comprensión y estructurado lógica que el sistema tiene. Esta comprensión por parte de los usuarios con relación al sistema evalúa los siguientes pasos:

- Comprensibilidad
- Facilidad de aprender
- Operatividad

Se realizan encuestas a los usuarios finales sobre el manejo, la comprensión y la factibilidad de aprender el sistema para medir la usabilidad según la siguiente tabla:

PREGUNTAS	RESPUESTAS		PORCENTAJE
	SI	NO	
¿El acceso al sistema es complicado?	0	5	100%
¿Es crítico el rendimiento?	1	4	80%
¿Son complicados los procesos que realiza el sistema?	1	4	80%
¿Es satisfactorio las repuestas que el sistema devuelve?	4	1	80%
¿El sistema tiene interfaces entendibles?	4	1	80%
¿El sistema reduce su tiempo de trabajo?	5	0	100%
¿Es difícil aprender a manejar el sistema?	2	3	60%

¿El sistema satisface las necesidades que usted requiere?	<b>4</b>	<b>1</b>	<b>80%</b>
¿Utiliza el sistema con facilidad?	<b>3</b>	<b>2</b>	<b>60%</b>
¿Considera usted que es una herramienta útil?	<b>5</b>	<b>0</b>	<b>100%</b>
<b>PROMEDIO</b>			<b>82%</b>

**Tabla 14.** Encuestas de usabilidad del sistema

**Fuente:** [Elaboración Propia]

### 3.9.1.2. Mantenibilidad

Para hallar la mantenibilidad del sistema se utiliza el índice de madurez de software (IMS), que proporciona una indicación de la estabilidad de un producto de software, basado en los cambios que ocurren con cada versión del producto.

Se determina la siguiente fórmula para hallar en (IMS).

$$\text{IMS} = [\text{Mt} - (\text{Fc} + \text{Fa} + \text{Fe})] / \text{Mt}$$

Donde:

**Mt:** número de módulos total de la versión actual.

**Fc:** número de módulos de la versión actual que se cambiaron.

**Fa:** número de módulos de la versión actual que se añadieron.

**Fe:** número de módulos de la versión anterior que se eliminaron en la versión actual.

Por lo tanto se aplica al sistema y da el siguiente resultado de índice de madures.

$$\text{IMS} = [5 - (0 + 1 + 0)] / 5$$

$$\text{IMS} = 0.8 * 100\% = 80\%$$

Por tanto el sistema empieza a estabilizarse en un 80%



### 3.9.1.3. Funcionabilidad

La funcionalidad de un sistema no puede ser medido directamente, entonces corresponde derivar, mediante otras medidas directas como *punto función*.

Los puntos función miden el software desde una perspectiva del usuario, dejando de lado los detalles de la codificación.

Es una técnica totalmente independiente de todas las consideraciones del lenguaje y ha sido aplicada en más de 240 lenguajes diferentes. Se supone que FPA evalúa con fiabilidad.

El proceso consta de dos etapas:

**Pasó 1.-** Se identifican las funciones disponibles para el usuario y se organizan en cinco grupos.

**Pasó 2.-** Se ajusta este total de acuerdo con unas características del entorno.

Haremos uso de cinco características de dominios de información y se proporcionan las cuentas en la posición apropiada de la tabla. Los valores de los dominios de la información, y se definen de la siguiente manera.

**Número de entradas de usuario.-** Se cuenta cada entrada de usuario que proporciona diferentes datos orientados a la aplicación. Las entradas deberían diferenciar de las peticiones, las cuales se encuentran de forma separada.

**Número de salidas de usuario.-** En este contexto la salida de usuario se refiere a informes, pantallas, mensajes de error, etc. Los elementos de datos particulares dentro de un informe no se cuentan de forma separada.

**Número de peticiones de usuario.-** Una petición se define como una entrada interactiva que produce la generación de alguna respuesta del software inmediata en forma de salida interactiva. Se cuenta cada petición por separado.

**Número de archivos.-** Se cuenta cada archivo maestro lógico (se refiere a un grupo lógico de datos que puede ser parte de una gran base de datos o un archivo independiente).

**Número de interfaces externos.-** Se cuentan todas interfaces legibles por la máquina que se utilizan para transmitir información a otro sistema. [Pressman, 2005]

De acuerdo a la información y comportamiento del sistema se obtuvieron los siguientes datos:

<b>PARÁMETROS DE ENTRADA</b>	<b>CUENTA</b>
Nro. de entradas	75
Nro. de salidas	50
Nro. de peticiones de usuario.	35
Nro. de archivos	60
Nro. de interfaces externos	20

**Tabla 15.** Entradas de funcionalidad según punto función

**Fuente:** [Elaboración Propia]

Los puntos función se calculan con la ayuda de la anterior tabla considerando los factores de ponderación medio.

<b>PARÁMETROS DE MEDICIÓN</b>	<b>CUENTA</b>	<b>FACTOR DE PONDERACIÓN MEDIO</b>	<b>TOTAL</b>
Nro. de entradas	75	5	375
Nro. de salidas	50	6	300
Nro. de peticiones de usuario.	35	5	175
Nro. de archivos	60	10	600
Nro. de interfaces externos	20	7	140

Cuenta total			<b>1590</b>
--------------	--	--	-------------

**Tabla 16.** Cuenta total con factor de ponderación medio

**Fuente:** [Elaboración Propia]

Calcular la relación para calcular el punto función.

$$PF = CUENTA\ TOTAL * (Grado\ de\ confiabilidad + Tasa\ de\ error * \Sigma F)$$

Donde:

**PF** = medida de confiabilidad.

**CUENTA TOTAL** = la suma del valor de las entradas, salidas, peticiones, interfaces externas y archivos.

**GRADO DE CONFIABILIDAD** = confiabilidad estimada del sistema.

**TASA DE ERROR** = probabilidad subjetiva estimada del dominio de la informaciones este error es del 1%.

**ΣF** = son los valores de ajuste de complejidad que toman los valores de la tabla anterior que dan como respuesta en la siguiente tabla:

<b>FACTORES</b>	<b>SIN IMPORTANCIA</b>	<b>INCREMENTAL</b>	<b>MODERADO</b>	<b>MEDIO</b>	<b>SIGNIFICATIVO</b>	<b>ESENCIAL</b>
<b>FACTOR</b>	<b>0</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>
¿Requiere el sistema copias de seguridad y de información?						0
¿Existen funciones de procesos distribuidos?			0			

¿Se requiere comunicación de datos?					0	
¿Es crítico el rendimiento?			0			
¿El sistema web se ejecuta en el sistema operativo actual?	0					
¿Se requiere una entrada interactiva para el sistema?				0		
¿Se requiere que el sistema tenga entrada de datos con múltiples ventanas?				0		
¿Son complejos las entradas, salidas, los archivos o las peticiones?			0			
¿Es complejo el procesamiento interno del sistema?			0			
¿Se ha diseñado el código para ser reutilizable?				0		
¿Se ha diseñado el sistema para facilitar a los usuarios el trabajo y ayudarlos a encontrar la información?						0
<b>TOTAL ΣFi</b>	<b>30</b>					

**Tabla 17.** Ajuste de complejidad del punto fusión

**Fuente:** [Elaboración Propia]

Sin importancia	0
Incremental	1
Moderado	2
Medio	3
Significativo	4
Esencial	5

**Tabla 18.** Ajuste de complejidad del punto fusión

**Fuente:** [Elaboración Propia]

Con la obtención de los datos anteriores se considera un grado de confiabilidad del 80% calculemos el grado de punto fusión.

$$\text{PF} = \text{CUENTA TOTAL} * (\text{Grado de confiabilidad} + \text{Tasa de error} * \Sigma F)$$

$$\text{PF} = 1590 * (0.8+0.01*30)$$

$$\text{PF} = 1749$$

Si consideramos el máximo valor de ajuste de complejidad como  $\Sigma(fi) = 55$ , se tiene;

$$\text{PF} = 1590 * (0.8+0.01*55)$$

$$\text{PF} = 2147$$

Calcularemos el valor de ajuste de la complicación del punto función. Por lo tanto la funcionalidad es real es:

$$\text{FUNCIONALIDAD} = (1749 / 2147) * 100\%$$

$$\text{FUNCIONALIDAD} = 81\%$$

#### 3.9.1.4. Confiabilidad

Es la probabilidad de operación libre de fallos de un programa en un entorno determinado y durante un tiempo específico.

Observamos el trabajo hasta que se produzca un fallo en el instante t, se halla la probabilidad de falla con una variable aleatoria continua T, en una función eXPonencial la relación es la siguiente:

Probabilidad de hallar una falla:  $P(T \leq t) = F(t)$

Probabilidad de no hallar una falla:  $P(T > t) = 1 - F(t)$

$$\text{Conf}(t) = FC * e^{-\lambda/6*12}$$

Donde:

FC = 0.81; funcionalidad del sistema.

$\lambda = 1$ ; tasa de fallos en 8 ejecuciones dentro de un mes.

Hallamos la confiabilidad del sistema:

$$F(t) = FC * e^{\frac{\lambda}{6} * 12}$$

$$F(t) = 0.81 * e^{\frac{1}{6} * 12}$$

$$F(t) = 0.13$$

La portabilidad de hallar una falla es de un 13% durante los próximos 12 meses.

$$P(t > t) = 1 - F(t)$$

$$P(T > t) = 1 - 0.13$$

$$P(T > t) = 0.87$$

Por tanto se determina que la probabilidad de no hallar una falla es del 87% durante los próximos 12 meses, por lo tanto es una aceptación confiable y aceptable de parte del sistema.

### 3.9.1.5. Portabilidad

La portabilidad se refiere a la habilidad del software de ser transferido de un ambiente a otro, se consideran los siguientes aspectos.

- **Adaptabilidad:** Se evalúa la capacidad de adaptar el software a diferentes ambientes sin la necesidad de aplicarle modificaciones.
- **Facilidad de instalación:** Es el esfuerzo necesario para instalar el software en un ambiente determinado.

- **Conformidad:** Permite evaluar si el sistema se adhiere a estándares o convenciones relativas a portabilidad.
- **Capacidad de reemplazo:** Hace referencia a la oportunidad y el esfuerzo usado en sustituir el software por otro producto con funciones similares.

### 3.9.2. Resultados

El factor de calidad total está directamente relacionado con el grado de satisfacción con el usuario que ingresa al sistema web de inventario, ingresos, salidas y bajas.

A continuación se muestra los resultados de la evaluación de calidad.

CARACTERÍSTICAS	RESULTADOS
Usabilidad	0.82
Funcionabilidad	0.81
Confiabilidad	0.87
Mantenibilidad	0.80
<b>EVALUACIÓN DE CALIDAD TOTAL</b>	<b>0.82</b>

**Tabla 19.** Resultado de evaluación de calidad

**Fuente:** [Elaboración Propia]

El nivel de aceptabilidad satisfactorio, indica que los valores de preferencia se encuentran en el rango 60 – 100.

El nivel de aceptabilidad marginal, indica que los valores de preferencia se encuentra en el rango de 0 -40.

Por los tanto la calidad total está directamente relacionada con el grado de satisfacción del usuario que ingresa al sistema web, la calidad total es 82%.

### 3.10. SEGURIDAD DE SOFTWARE

En la documentación oficial de laravel ya nos facilita el tema de seguridad de software y recomienda que se utilice su propia aplicación de testeado de software el cual hace que el trabajo de desarrollo y testeado sea más fácil de manejar.

### 3.10.1. Autenticación

Laravel hace que implementar la autenticación sea muy sencilla. Las instalaciones de autenticación están compuestas por guardias y proveedores. Los guardias definen como se autentican los usuarios para cada solicitud. Los proveedores definen como se recuperan los usuarios de su almacenamiento persistente. Laravel envía con soporte para recuperar usuarios usando Eloquent y el generador de consultas de base de datos.

Los guardias para la seguridad de autenticación estarían en la ruta `almacen/routes/web.php` el siguiente fragmento de código:

```
Route::group(['middleware'=>['guest']],function(){
    Route::get('/', 'Auth\LoginController@showLoginForm');
    Route::post('/login', 'Auth\LoginController@login')->name('login');
});
```

**Fig. 40** Código de autenticación o guardias

Fuente: [Elaboración Propia]

Y como proveedor o aceptación de autenticación estaría el siguiente código:

```
Route::group(['middleware'=>['auth']],function(){
    .....
});
```

**Fig. 41** Código de autenticación o proveedores

Fuente: [Elaboración Propia]

En la tabla personas cuando se actualice la contraseña encriptada con AES-128 o se haga un `remember_token` de usuario se actualizará inmediatamente la fecha de `updated_at`.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Comentarios	Extra	Acción
<input type="checkbox"/>	1	id	int(10)	UNSIGNED	No	Ninguna		AUTO_INCREMENT	Cambiar  Eliminar  Más
<input type="checkbox"/>	2	nombre	varchar(50)	utf8mb4_unicode_ci	No	Ninguna			Cambiar  Eliminar  Más
<input type="checkbox"/>	3	num_doc	varchar(20)	utf8mb4_unicode_ci	No	Ninguna			Cambiar  Eliminar  Más
<input type="checkbox"/>	4	cargo	varchar(100)	utf8mb4_unicode_ci	No	Ninguna			Cambiar  Eliminar  Más
<input type="checkbox"/>	5	direccion	varchar(100)	utf8mb4_unicode_ci	Sí	NULL			Cambiar  Eliminar  Más
<input type="checkbox"/>	6	telefono	varchar(20)	utf8mb4_unicode_ci	Sí	NULL			Cambiar  Eliminar  Más
<input type="checkbox"/>	7	email	varchar(50)	utf8mb4_unicode_ci	Sí	NULL			Cambiar  Eliminar  Más
<input type="checkbox"/>	8	created_at	timestamp		Sí	NULL			Cambiar  Eliminar  Más
<input type="checkbox"/>	9	updated_at	timestamp		Sí	NULL			Cambiar  Eliminar  Más

**Fig. 42** Tabla de base de datos de personas

Fuente: [Elaboración Propia]



#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Comentarios	Extra	Acción
<input type="checkbox"/>	1 <b>id</b>	int(10)		UNSIGNED	No	Ninguna			
<input type="checkbox"/>	2 <b>usuario</b>	varchar(191)	utf8mb4_unicode_ci		No	Ninguna			
<input type="checkbox"/>	3 <b>password</b>	varchar(191)	utf8mb4_unicode_ci		No	Ninguna			
<input type="checkbox"/>	4 <b>condicion</b>	tinyint(1)			No	1			
<input type="checkbox"/>	5 <b>idrol</b>	int(10)		UNSIGNED	No	Ninguna			
<input type="checkbox"/>	6 <b>remember_token</b>	varchar(100)	utf8mb4_unicode_ci		Sí	NULL			

**Fig. 43** Tabla de base de datos de usuarios  
Fuente: [Elaboración Propia]

### 3.10.2. Autorización

Laravel proporciona un enfoque de autorización simple basado en el cierre, mientras que las políticas, como los controladores, agrupas su lógica en torno a un modelo o recurso en partículas. Ejemplo:

```
Route::group(['middleware'=>['auth']],function(){
    Route::post('/logout', 'Auth\LoginController@logout')->name('logout');
    Route::get('/main', function () {
        return view('contenido/contenido');
    })->name('main');
    Route::group(['middleware' => ['Almacenero']], function () {
        Route::get('/categoria', 'CategoriaController@index');
        Route::post('/categoria/registrar', 'CategoriaController@store');
        Route::get('/entrada', 'IngresoController@index');
        Route::post('/entrada/registrar', 'IngresoController@store');
        Route::get('/entrada/obtenerCabecera', 'IngresoController@obtenerCabecera');
        Route::get('/entrada/obtenerDetalles', 'IngresoController@obtenerDetalles');
        Route::get('/entrada/listarDetalles', 'IngresoController@listarDetalles');
    });
});
```

**Fig. 44** Código de autenticación  
Fuente: [Elaboración Propia]

### 3.10.3. Hash

Hash Bcrypt o es una excelente opción para hacer hash de contraseñas porque su factor de trabajo es ejecutable, lo que significa que el tiempo lleva generar un hash puede incrementarse a medida que aumenta la potencia del hardware. Ejemplo:

```
public function store(Request $request)
{
    if (!$request->ajax()) return redirect('/');

    try{
        DB::beginTransaction();

        $user = new User();
        $user->id = $persona->id;
        $user->idrol = $request->idrol;
        $user->usuario = $request->usuario;
        $user->password = bcrypt( $request->password);
        $user->condicion = '1';
        $user->save();

        DB::commit();
    } catch (Exception $e){
        DB::rollBack();
    }
}
```

**Fig. 45** Código de Hash  
Fuente: [Elaboración Propia]

## 3.11. EVALUACIÓN DE COSTOS

El método de costos se basa en la razón de un proyecto en particular. Se considera que un proyecto es atractivo, cuando los beneficios derivados de su implementación exceden a su costo COCOMO II.

### 3.11.1. Análisis de costos

Para la determinación del costo del software desarrollado, se usa el modelo constructivo costo COCOMO II.

Obteniendo en la anterior sección que **PF = 1749**

Conversión de punto función a KLDC

Ahora se convierte los puntos función a miles de líneas de código, tomando en cuenta la siguiente tabla.

LENGUAJE	NIVEL	FACTOR LDC/PFF
C	2.5	128
Java	5	64
Visual Basic	7	46
ASP	9	36
PHP	6	53
Visual C++	9.5	34

**Tabla 20.** Conversión de puntos de función para COCOMO II

**Fuente:** [Elaboración Propia]

$$\text{LDC} = \text{PF} * \text{Factor LDC/PF}$$

$$\text{LDC} = 1749 * 53$$

$$\text{LDC} = 92697$$

Las líneas de código son 96407 de las cuales se estima que un 50% del código es reutilizable, entonces el total de LDC es:

$$\text{KLDC} = (\text{total LDC} - \text{LDC reutilizable}) / 1000$$

$$\text{KLDC} = (92697 - 46348.5) / 1000 = 46.35$$

Por tanto existen 46.35 líneas de código distribuidos para el proyecto.

Ahora se aplican las fórmulas básicas de esfuerzo, tiempo calendario y personal requerido.

Las ecuaciones de COCOMO básico tiene la siguiente forma:

$$E = ab(KLDC)^{bb}$$

$$D = Cb(E)^{db}$$

Donde:

E: esfuerzo aplicando en personas por mes.

D: tiempo de desarrollo en meses cronológicos.

KLDC: número estimado de líneas de código distribuidas en miles.

PROYECTO DE SOFTWARE	ab	bb	Cb	db
Orgánico	2.4	1.05	2.5	0.38
Semi-acoplado	3.0	1.12	2.5	0.35
Empotrado	3.6	1.20	2.5	0.32

**Tabla 21.** Relación de valores en el modelo COCOMO

**Fuente:** [Elaboración Propia]

Reemplazando los datos en la ecuación se tiene:

$$E = 2.4 (46.35)^{1.05}$$

$$D = 2.5(134.76)^{0.38}$$

$$E = 134.76 \text{ programadores / mes}$$

$$D = 16 \text{ meses}$$

El personal requerido, en este caso el número de programadores se obtiene con la siguiente fórmula:

$$\text{Número de programadores} = E/D = 134.76 / 16 = 8 \text{ programadores}$$

El salario de un programador puede oscilar entre los 100\$us, cifra que es tomada en cuenta para la estimación siguiente:

$$\text{Costo de software desarrollado} = \text{Núm. de programadores} * \text{Salario de un programador}$$

$$\text{Costo del software desarrollado por persona} = 8 * 100\$us = 800\$us$$

$$\text{Costo de estimación software desarrollado} = 800 * 8 = 6400\$us$$

### **3.11.2. Características de hardware requeridas**

En las características mínimas que se requieren para la ejecución e implementación del sistema son:

- Un servidor Intel de 2 de memoria RAM con 50gb de disco duro y procesador de 2 Hz.
- Sistema operativo Windows o Linux
- Si en caso que fuera un sistema operativo Windows un servidor XAMPP o LAMPP que ya tiene instaladas php7.1 y mysql5.0.
- Si en caso que fuera Linux instalar el servidor apache, php7.1 y mysql5.0.

### **3.12. ANÁLISIS DE BENEFICIOS**

Los beneficios del presente proyecto son aquellos que tienen un impacto importante para la Caja Nacional de Salud Regional La Paz.

- Mejora en la atención a solicitudes.
- Facilita la consulta y acceso a la documentación de los equipos electrónicos.
- Aumenta la transparencia organizativa y control de los equipos electrónicos.
- Contempla la incorporación de los recursos digitales asociadas a documentos y descripción de equipos en distintos niveles de descripción.
- Ahorra tiempo y esfuerzo en el registro, salida y reportes de equipos electrónicos.

Al contar con un administrador del sistema que en este caso es un miembro del personal o el responsable de área de sistemas se da mayor seguridad a todos los usuarios en lo que se refiere a la confidencialidad, integridad y la disponibilidad creando roles y permisos para navegar por el sistema.

## CAPÍTULO IV

### CONCLUSIONES Y RECOMENDACIONES

#### 4.1. CONCLUSIONES

Una vez concluido con todos los puntos propuestos para el desarrollo del proyecto Sistema Web de Gestión de Inventarios para almacenes, tomando en cuenta la problemática y los objetivos planteados se puede afirmar que se han logrado los objetivos trazados llegando a las siguientes conclusiones:

- El objetivo principal, se concluyó con el desarrollo del sistema web de gestión de inventarios para almacenes de manera satisfactoria tanto para los usuarios del sistema como para la institución, el mismo ayudara a optimizar los procesos de control y seguimiento de información que se realiza a los equipos electrónicos.
- Se realizó la ingeniería de requerimientos utilizando entrevistas a los usuarios que requieren del sistema, se realizó la correspondiente encuesta para la funcionalidad satisfactoria del sistema y análisis de información de manera satisfactoria obteniendo un diseño del sistema óptimo para el desarrollo del software.
- El diseño del sistema fue desarrollado con la metodología UWE, para el desarrollo, iteraciones y reuniones con la parte interesada se utilizó las metodologías SCRUM y XP.
- Acceder a la información de almacenes de manera segura y confiable por el uso de contraseñas y rol asignados a cada usuario.
- El sistema permite obtener el control de equipos electrónicos en niveles de existencia y disponibilidad de equipos.
- El sistema emite advertencias para procesos de abastecimiento en el stock del almacén.
- El sistema emite reportes de inventarios de almacén.
- Permite realizar el registro de ingreso de equipos electrónicos y toda su información de compras, contratación, factura e información de seriales de cada equipo, emitiendo un acta de ingreso para el reporte de activos.
- Adecuado control de garantías y su depreciación de equipos electrónicos.

- El sistema permite mostrar un kardex de control de salida y disponibilidad de cada compra de equipos electrónicos.
- El sistema emite reportes de inventarios de cada compra detallado de activos fijos.
- Permite realizar el registro de salida de equipos electrónicos, mostrando desacuerdo a su caducidad y emitiendo un acta de salida detallando la información más relevante de la misma.
- El sistema muestra el proceso de garantías vigentes de cada equipo electrónico salido para su devolución o baja del equipo.
- Permite la búsqueda de cada equipo que salió detallando la compra.
- La interfaz del usuario tiene acceso a pantallas comprensibles y de fácil manejo.

Es importante señalar que el software desarrollado, actualmente se encuentra en etapa de producción y para los almaceneros como administradores de la institución se constituye en una herramienta que brinda un óptimo control y seguimiento de todos los procesos de los cuales se planteó en este presente proyecto, brindando también una información integra, precisa y oportuna.

#### **4.2. RECOMENDACIONES**

Si bien el sistema cumple con los requerimientos de la institución. Se puede mencionar las siguientes recomendaciones:

- Los almaceneros que realizan el registro de datos deben tener en cuenta la importancia de todos y cada uno de los campos que se define el sistema de gestión de inventarios así también de conocimiento básico sobre la base de datos a fin de dar más valor a los instrumentos descriptivos que se generan.
- A los administradores del sistema, dar pleno cumplimiento a las políticas de seguridad de la información adoptadas al proyecto, sobre todo lo referente a los backups y la seguridad física.
- La información nueva introducida al sistema deberá ser previamente revisados, para que los reportes emitidos por el sistema sean fiables para los usuarios que lo requieran.

## BIBLIOGRAFÍA

Rafael Lapiedra Alcami, Carlos Devece Carañana, Joaquin Guiral Herrano (2010). Introducción a la gestión de sistemas de información en la empresa. 1° ed. Castellón de la Plana: Publicaciones de la Universidad Jaume 1.

Diego Fernando Quizhpi Campoverde (2018). Diseño de un sistema de control de inventario y organización de las bodegas de producto terminado de la empresa ECUAESPUMAS-LAMITEX S.A Cuenca-Ecuador, Universidad Politécnica Salesiana Sede Cuenca.

Miguel Ángel Hernández Trinidad (2017). Sistema de gestión de almacenes con identificación automática de captura de datos, para un control eficiente del flujo de procesos, México, Instituto Politécnico Nacional.

Grover Gutiérrez Vargas (2015). Sistema de control de ventas para almacenes de aluminios utilizando dispositivos móviles CASO: Técnica de aluminio, vidrio y servicios (TALVISER) La Paz, Universidad Mayor de San Andrés, Facultad de Ciencias Puras y Naturales.

Miriam Olga Ortega Mamani (2007). Sistema de administración y control de almacén e inventario GMLP CASO: Mallasa La Paz, Universidad Mayor de San Andrés, Facultad de Ciencias Puras y Naturales.

Marco Antonio Vicente Magne (2011). Sistema de inventario y control de almacenes CASO: AASANA La Paz, Universidad Mayor de San Andrés, Facultad de Ciencias Puras y Naturales.

Diego Fernando Quizhpi Campoverde (2018). Diseño de control de inventario y organización de las bodegas de producto terminado CASO: Empresa Ecuaespumas-Lamitex S.A. Cuenca – Ecuador. Universidad Politécnica salesiana sede cuenca.

Pressman Roger (2005), Ingeniería de Software un efecto práctico 7ma edición, Edición Mc-Graw Hill. España.



Siniestra G. Polanco L. (1993) Contabilidad, sistemas de información para las organizaciones 2da edición, Editorial Mc-Graw Hill, Bogotá - Colombia.

Julián Andrés Zapata Cortez. (2014) Fundamentos de la gestión de inventarios, Centro Editorial Esumer, Medellín - Colombia.

Ballou, R.H. (2004), Logística. Administración de la cadena de suministros. Quinta edición.

Citlali Nieves Guerrero, Juan Ucan Pech y Víctor Menéndez Domínguez. (2014) UWE en Sistema de Recomendación de Objetos de Aprendizaje. Aplicando Ingeniería Web: Un Método en Caso de Estudio de la Universidad Autónoma de Yucatán Facultad de Matemáticas, México.

Eugenia Bahit (2012), SCRUM y Extreme Programming para programadores. SafeCreative, Buenos Aires, Argentina.

Patricio Letelier y M. Carmen Pañades (2005), Metodologías ágiles para el desarrollo del software extreme programming (XP). Laboratorio de sistemas de información. Universidad Politécnica de Valeria.

Mari Yicel Miranda Gómez, Sammy Cuaran Rosas y Julián Andrés Mera Paz (2017), Análisis Sistemático de Información de la norma ISO 25010 como base para la implementación en un laboratorio de testeo de software, Universidad Cooperativa de Colombia sede Popayán.

## REFERENCIAS WEB

Documentación y guía de desarrollo oficial de laravel

<http://www.laravel.com/document/5.5> [acceso: abril de 2019]

Desarrollo oficial de gestor de base de datos Mysql

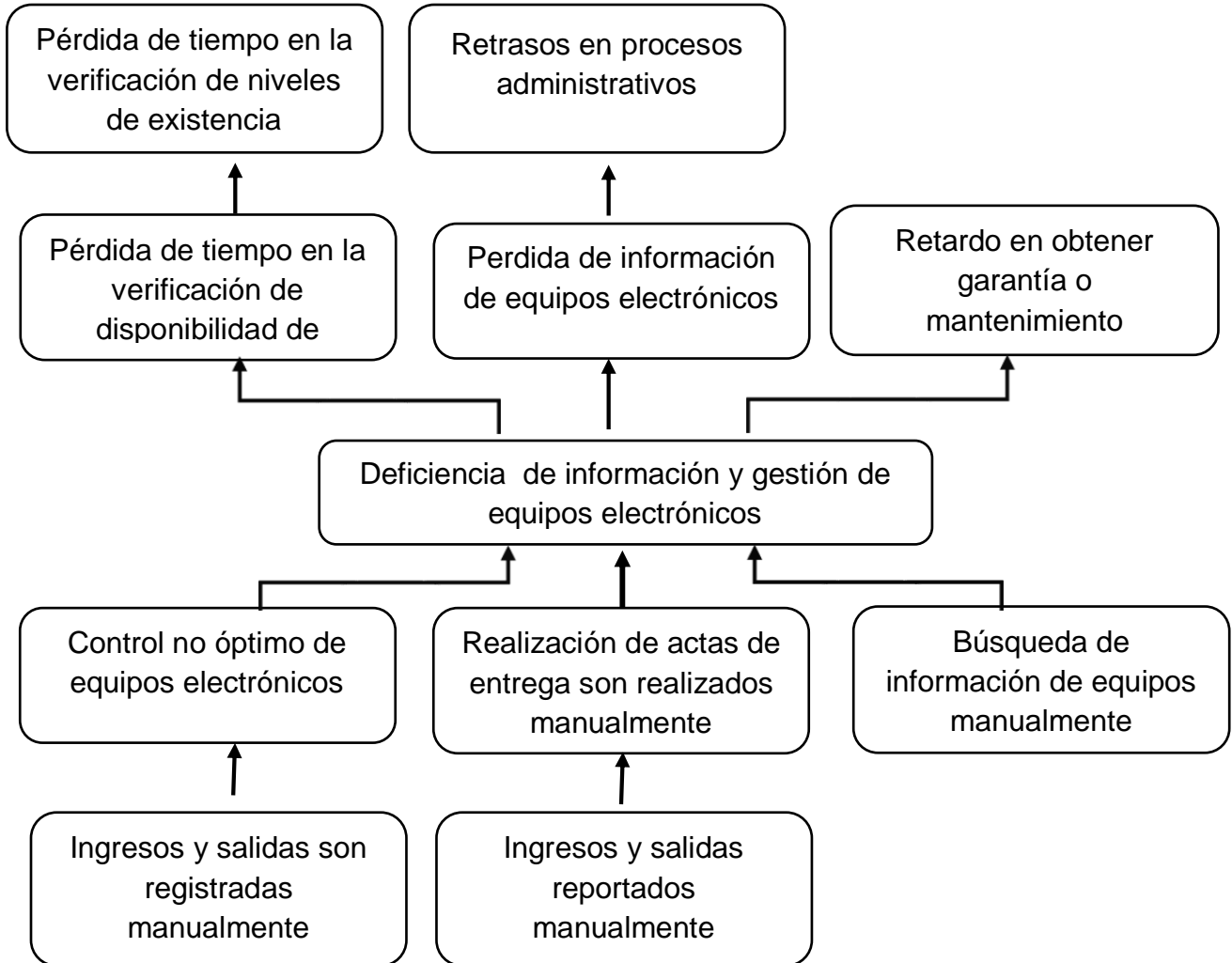
<http://www.nerion.es> [acceso: abril de 2019]

Documentación y guía de desarrollo oficial de vuejs

<http://www.lvuejs.org/document> [acceso: junio de 2019]

# **ANEXOS**

## ARBOL DE PROBLEMAS



## ARBOL DE OBJETIVOS

