

# UNIVERSIDAD PÚBLICA DE EL ALTO

## CARRERA INGENIERÍA DE SISTEMAS



### PROYECTO DE GRADO

## “SISTEMA DE INFORMACIÓN PARA EL CONTROL DE INVENTARIO EN ALMACÉN”

CASO: (E.S.F.M.T.H.E.A.) “ESCUELA SUPERIOR DE FORMACIÓN DE MAESTROS TECNOLÓGICO HUMANÍSTICO EL ALTO”

Para optar al título de Licenciatura en Ingeniería de Sistemas  
Mención: Informática y Telecomunicaciones

**Postulante:** Nolia Chamairo Espejo  
**Tutor Metodológico:** MSc. Marisol Arguedas Balladares  
**Tutor Especialista:** Lic. Edwin Mamani Viscarra  
**Tutor Revisor:** Lic. Freddy Salgueiro Trujillo

EL ALTO – BOLIVIA  
2020

“.....para comenzar un proyecto hace falta valentía y para culminar un proyecto hace falta perseverancia y amor.....”

**DEDICO ESTE PROYECTO:**

A la memoria de mi padre, quien me  
enseñó que el mejor conocimiento  
que se puede tener es el que se  
aprende por sí mismo.

A Dios, por ser la luz incondicional  
que guía mi camino.

***Nolia Chamairo Espejo***

## Agradecimiento

A Dios, quien como mi gran guía espiritual estuvo presente en cada paso de mi vida, protegiéndome, bendiciéndome y dándome fuerzas para continuar con mis metas trazadas sin desfallecer.

A mis padres Juanito Chamairo Santander y Jesusa Espejo Condori, por toda la confianza que depositó en mi persona, quien me inculcó los buenos valores y me preparó para poder enfrentar todas las vicisitudes de esta vida. La fuerza y la fe de mis padres me dieron una nueva apreciación del significado y la importancia de la vida, la amistad, el sacrificio y la voluntad de seguir avanzando. Su ejemplo me mantuvo soñando cuando quise rendirme.

Dicen que la mejor herencia que nos pueden dejar los padres son los estudios, sin embargo, no creo que sea el único legado del cual yo particularmente me siento muy agradecida, mis padres me han permitido trazar mi camino y avanzar por este camino de senderos escarpados, por eso y por mucho más mis agradecimientos especiales va para ellos.

A mis hermanos, por estar siempre pendientes, por brindarme su apoyo incondicional y creer en mí.

A mis tutores quienes confiaron en mí y me *guiaron* con paciencia, encausando mi trabajo con sus conocimientos y apoyo a través de cada una de las etapas de este proyecto para alcanzar los resultados que buscaba, con el firme propósito de conseguir un producto en beneficio de la sociedad.

A mis amigos Juan Calos Yujra Quispe, Liset Sanga Cucho, Luis Jhonny Tapia y todas aquellas personas especiales que de alguna manera me acompañaron en esta etapa, aportando a mi formación tanto profesional y como ser humano, quienes confiaron en mi capacidad.

Mi agradecimiento especial a la Universidad Pública de El Alto, institución que me cobijo durante los años de estudio que cursé en ella, a sus autoridades, docentes y personal administrativo, quienes día a día aportan con su esfuerzo para la formación de nuevos profesionales en beneficio de la sociedad

# INDICE GENERAL

Dedicatoria.....	i
Agradecimiento.....	ii
INDICE GENERAL.....	iii
INDICE DE FIGURAS.....	xi
INDICE DE TABLAS.....	xiii
<b>1. MARCO PRELIMINAR.....</b>	<b>1</b>
1.1 INTRODUCCIÓN.....	1
1.2 ANTECEDENTES.....	2
1.2.1 Antecedentes de la Institución.....	2
1.2.2 Antecedentes Internacionales.....	3
1.2.3 Antecedentes Nacionales.....	4
1.3 PLANTEAMIENTO DEL PROBLEMA.....	4
1.3.1 Problema Principal.....	4
1.3.2 Problemas Secundarios.....	4
1.4. OBJETIVO.....	5
1.4.1 Objetivo General.....	5
1.4.2 Objetivos Específicos.....	6
1.5 JUSTIFICACIONES.....	6
1.5.1 Justificación Técnica.....	6
1.5.2 Justificación Económica.....	7
1.5.3 Justificación Social.....	7
1.6 METODOLOGÍAS.....	8
1.6.1 Metodología UWE.....	8
1.6.2 Normas de Seguridad.....	8
1.7 HERRAMIENTAS.....	9
1.7.1 Lenguajes de Programación.....	9
1.8. LÍMITES Y ALCANCES.....	11
1.8.1 Límites.....	11
1.8.2. Alcances.....	12
1.9 APORTES.....	13
<b>2 MARCO TEORICO.....</b>	<b>15</b>
2.1 SISTEMA.....	15
2.2 SISTEMA DE INFORMACION.....	16

2.3 CONTROL .....	16
2.4. INVENTARIO.....	17
2.4.1 Funciones del control de inventarios .....	17
2.4.1.1 Las principales funciones son:.....	17
2.4.1.2 Método de Inventario PEPS.....	17
2.5 ALMACEN .....	20
2.5.1 Funciones del Almacén.....	20
2.6 INGENIERIA DE REQUERIMIENTOS .....	21
2.6.1 Requerimientos Funcionales.....	21
2.6.2. Requerimientos No Funcionales.....	21
2.7 INGENIERIA DE SOFTWARE .....	23
2.8 INGENIERIA WEB.....	25
2.8.1 Características de la Ingeniería Web.....	26
2.9 METODOLOGIA DE INGENIERIA WEB .....	26
2.9.1. Metodología UWE .....	27
2.9.2. Características de la Metodología UWE.....	27
2.9.3. Modelo de la Metodología De UWE.....	27
2.9.3.1. Modelo de Casos de USO.....	27
2.9.3.2. Modelo de Contenido.....	28
2.9.3.3. Modelo de Navegación.....	29
2.9.3.4. Modelo de Presentación.....	30
2.10. MODELO DE IMPLEMENTACION.....	30
2.10.1 Fases de la Metodología UWE .....	31
2.11 HERRAMIENTAS.....	32
2.11.1 Servidor Apache.....	32
2.11.2 Base De Datos .....	33
2.11.2.1 Gestor de Base de Datos MYSQL .....	33
2.11.3 Lenguaje De Programación PHP.....	34
2.11.4 Herramientas de Diseño .....	34
2.11.4.1 HTML .....	34
2.11.4.2 Javascript .....	34
2.11.4.3 CSS3.....	35
2.11.5 Framework Angular.....	35
2.11.6 Framework Materialize.....	35

2.12 PRUEBAS DE SOFTWARE.....	36
2.12.1 Diseño de casos de prueba: .....	37
2.12.1.1 Pruebas de Caja Blanca.....	38
2.12.1.2 Pruebas de Caja Negra .....	39
2.13 METRICAS DE CALIDAD .....	41
2.13.1 ISO 9000 .....	42
2.13.2. Estándares ISO/IEC 9126.....	43
2.13.2.1 Norma de Evaluación ISO/IEC 9126 .....	44
2.14 SEGURIDAD INFORMATICA .....	55
2.14.1 Estándar ISO/IEC 27000.....	55
2.14.2 ISO 27002 .....	55
2.15 METODOS DE ESTIMACION DE COSTOS DE SOFTWARE .....	56
2.15.1 Modelo COCOMO .....	57
2.15.2 Modelo de Estimación .....	58
<b>3. MARCO APLICATIVO .....</b>	<b>63</b>
3.1. ANÁLISIS DEL SISTEMA ACTUAL.....	63
3.2 OBTENCIÓN DE REQUISITOS.....	64
3.2.1 Identificación de los Actores .....	64
3.3 INGENIERÍA DE REQUERIMIENTOS .....	66
3.3.1 Requerimientos Funcionales.....	67
3.3.2 Requerimientos No Funcionales.....	68
3.4 APLICACION DEL MODELO UWE .....	68
3.4.1 Modelo De Casos De Uso.....	68
3.4.1.1 Identificación de los Actores .....	68
3.4.2 Diagrama de casos de uso .....	69
3.4.2.1. Diagrama de casos de uso General del Sistema.....	69
3.4.2.2. Diagrama de Caso de Uso: Administración de Usuarios: Funcionarios	70
3.4.2.3. Diagrama de Caso de Uso: Administración de Usuarios: Área .....	72
3.4.2.4 Diagrama de Caso de Uso: Administración de Usuarios: Cargo .....	73
3.2.4.5 Diagrama de Caso de Uso: Proveedores.....	74
3.2.4.6. Diagrama de Caso de Uso: Control de Materiales .....	75
3.2.4.7. Diagrama de Caso de Uso: Entrada y Salida de Materiales .....	76
3.2.4.8. Diagrama de Caso de Uso: Reportes.....	77

3.2.4.9 Diagrama de Caso de Uso: Proveedores.....	78
3.4.3 Modelo De Contenido.....	78
3.4.4. Modelo De Navegación .....	78
3.4.4.1. Modelo de Navegación Administrador (Jefe de Almacén).....	79
3.4.4.2 Modelo de Navegación Administrador (Jefe de Almacén).....	80
3.4.5 Modelo de Presentación .....	80
3.4.5.1 Modelo de Presentación Administración .....	81
3.4.5.2 Modelo de Presentación de Funcionario.....	82
3.5 MODELO DE IMPLEMENTACION .....	82
3.5.1 Interfaz de Inicio de Sesión.....	82
3.5.2 Interfaz de Áreas (Administrador) .....	83
3.5.3 Interfaz Cargos: Administrador .....	84
3.5.4. Interfaz Funcionario: Administrador.....	84
3.5.5 Interfaz Usuario: Administrador .....	85
3.5.6. Interfaz Proveedores: Administrador .....	85
3.5.7. Interfaz Control de Materiales: Administrador .....	86
3.5.8 Interfaz Entrada y salida de Materiales: Administrador.....	88
3.5.9 Interfaz Reportes: Administrador .....	90
<b>4. CALIDAD Y SEGURIDAD .....</b>	<b>94</b>
4.1. PRUEBAS DE SOFTWARE.....	94
4.1.1 Pruebas de Caja Blanca .....	94
4.1.2. Pruebas De Caja Negra .....	102
4.1.3 Prueba de Carga .....	103
4.2 MÉTRICAS DE CALIDAD DE SOFTWARE .....	105
4.2.1 Funcionabilidad .....	105
4.2.2 Confiabilidad.....	110
4.2.3 Mantenibilidad .....	111
4.2.4 Usabilidad .....	113
4.2.4 Portabilidad.....	114
4.3 SEGURIDAD INFORMATICA .....	115
4.3.1 Sistema De Gestión De Seguridad De La Información Iso-27002.....	115
4.3.1.1 Seguridad Lógica.....	115
4.3.1.2 Seguridad Física.....	115
4.3.1.3 Seguridad Organizativa .....	116

4.3.1.4 Seguridad del Sistema.....	116
4.3.1.5 Políticas de Registros de Eventos.....	116
<b>5. ANALISIS DE COSTO DE SOFTWARE _____</b>	<b>119</b>
5.1 Método De Estimación Cócono.....	119
<b>6. CONCLUSIONES Y RECOMENDACIONES _____</b>	<b>124</b>
6.1 CONCLUSIONES.....	124
6.2 RECOMENDACIONES .....	125
<b>BIBLIOGRAFÍA _____</b>	<b>126</b>
<b>ANEXOS _____</b>	<b>130</b>

<b>MANUAL DEL USUARIO .....</b>	<b>i</b>
<b>1. INTRODUCCION.....</b>	<b>1</b>
<b>2. OBJETIVO.....</b>	<b>1</b>
<b>3. REQUERIMIENTOS DEL SISTEMA.....</b>	<b>1</b>
<b>4. TIPOS DE USUARIO .....</b>	<b>1</b>
4.1 INGRESO USUARIO ADMINISTRADOR.....	3
4.1.1 Módulo Área .....	3
4.1.1.1 Módulo Área - Botón Nueva Área .....	4
4.1.2 Módulo Cargos .....	5
4.1.2.1 Módulo Cargo - Botón Nueva Cargo.....	5
4.1.3 Módulo Funcionarios .....	6
4.1.4 Módulo Usuario .....	8
4.1.4.1 Módulo Usuario - Botón Nueva Usuario .....	9
4.1.5 Módulo Proveedor .....	10
4.1.5.1 Módulo Proveedor - Botón Nuevo Proveedor.....	11
Fuente: Elaboración propia.....	11
4.1.6 Módulo Control de Materiales .....	12
4.1.6.1 Sub-Módulo Categoría.....	12
4.1.6.1.1 Sub Módulo Categoría de Materiales - Botón Nueva Categoría. ....	13
4.1.6.2 Sub-Módulo Registro de Materiales.....	13
4.1.6.2.1 Sub Módulo Registro de Materiales - Botón Nuevo Material....	15
4.1.6.3 Sub-Módulo Cantidad Autorizada .....	16
4.1.6.4 Sub-Módulo Stock en Almacén .....	17
4.1.7 Módulo Entrada y Salida de Materiales .....	17
4.1.7.1 Sub-Módulo Solicitud de Material.....	18
4.1.7.2 Sub-Módulo Solicitar por Funcionario.....	19
4.1.7.3 Sub-Módulo Registro de Ingresos.....	20
4.1.7.3.1 Sub-Módulo Registro de Ingresos – Botón Nuevo Ingreso.....	21
4.1.7.4 Sub-Módulo Registro de Salida .....	22
4.1.8 Módulo Reportes .....	23

4.1.8.1 Sub-Módulo Reporte General.....	23
4.1.8.2 Sub-Módulo Reporte por Funcionario.....	24
4.2 INGRESO USUARIO FUNCIONARIO.....	25
4.2.1 Módulo Materiales .....	26
4.2.2 Módulo Solicitud de Materiales .....	27
4.2.3 Módulo Solicitudes .....	28

<b>MANUAL TÉCNICO</b> .....	<b>0</b>
<b>1. INTRODUCCIÓN</b> .....	<b>1</b>
<b>2. REQUERIMIENTOS TÉCNICOS</b> .....	<b>1</b>
2.1 REQUERIMIENTOS MÍNIMOS DE HARDWARE .....	1
2.2. REQUERIMIENTOS MÍNIMOS DE SOFTWARE.....	1
<b>3. HERRAMIENTAS UTILIZADAS PARA EL DESARROLLO</b> .....	<b>2</b>
3.1 PHP.....	2
3.2 MYSQL .....	2
3.3 APACHE .....	2
3.4 INSTALACIÓN DE WEB SERVER .....	3
<b>4. CREANDO PROYECTO CON ANGULAR JS</b> .....	<b>7</b>
4.1. CONFIGURAR LA BASE DE DATOS .....	8
4.2 SERVIDOR ACTIVO .....	9
<b>5. CASOS DE USO</b> .....	<b>9</b>
<b>6 MODULO DE ADMINISTRACION</b> .....	<b>10</b>

## INDICE DE FIGURAS

FIGURA N° 2. 1 GRÁFICA GENERAL DE UN SISTEMA.....	15
FIGURA N° 2. 2 SISTEMA DE INFORMACIÓN .....	16
FIGURA N° 2. 3 FASES DE INGENIERÍA DE SOFTWARE .....	25
FIGURA N° 2.4 DIAGRAMA DE CASOS DE USO.....	28
FIGURA N° 2. 5 DIAGRAMA DE CONTENIDO .....	28
FIGURA N° 2. 6 DIAGRAMA DE NAVEGACIÓN.....	29
FIGURA N° 2. 7 DIAGRAMA DE PRESENTACIÓN.....	30
FIGURA N° 2. 8 FASES DE LA METODOLOGÍA DE UWE .....	32
FIGURA N° 2. 9 ESTRUCTURA CAJA BLANCA.....	38
FIGURA N° 2. 10 ESTRUCTURA CAJA NEGRA .....	40
FIGURA N° 2. 11 NORMA ISO 9000 .....	43
FIGURA N° 2. 12 NORMA DE EVALUACIÓN DE ISO/IEC 9126.....	44
FIGURA N° 2. 13 CARACTERÍSTICAS DE CONFIABILIDAD ISO/IEC 9126 .....	48
FIGURA N° 2. 14 CARACTERÍSTICAS DE USABILIDAD ISO/IEC 9126.....	49
FIGURA N° 2. 15 CARACTERÍSTICAS DE EFICIENCIA ISO/IEC 9126.....	50
FIGURA N° 2. 16 CARACTERÍSTICAS DE CAPACIDAD DE MANTENIMIENTO ISO/IEC 9126 .....	51
FIGURA N° 2. 17 CARACTERÍSTICAS DE PROBABILIDAD ISO/IEC 9126 .....	53
FIGURA N° 2. 18 CALIDAD DE USO DE SOFTWARE ISO/IEC 9126.....	54
FIGURA N° 2. 19 ESTRUCTURA ISO 27002.....	56
FIGURA N° 2. 20 PUNTOS DE OBJETIVO.....	59
FIGURA N° 2. 21 DETALLES DE COEFICIENTE DE COCOMO.....	61
<i>FIGURA N° 3. 1 DIAGRAMA DE FLUJO DE DATOS .....</i>	<i>63</i>
<i>FIGURA N° 3. 2 ESCUELA DE FORMACIÓN DE MAESTRO EL ALTO E.S.F.M.T.H.A.....</i>	<i>65</i>
<i>FIGURA N° 3. 3 DIAGRAMA DE CASO DE USO: GENERAL DEL SISTEMA.....</i>	<i>69</i>
<i>FIGURA N° 3. 4 DIAGRAMA DE CASO DE USO: FUNCIONARIOS.....</i>	<i>70</i>
<i>FIGURA N° 3. 5 DIAGRAMA DE CASO DE USO: USUARIO .....</i>	<i>71</i>
<i>FIGURA N° 3. 6 DIAGRAMA DE CASO DE USO: ÁREA.....</i>	<i>72</i>
<i>FIGURA N° 3. 7 DIAGRAMA DE CASO DE USO: CARGO.....</i>	<i>73</i>
<i>FIGURA N° 3. 8 DIAGRAMA DE CASO DE USO: PROVEEDORES.....</i>	<i>74</i>
<i>FIGURA N° 3. 9 DIAGRAMA DE CASO DE USO: CONTROL DE MATERIALES.....</i>	<i>75</i>
<i>FIGURA N° 3. 10 DIAGRAMA DE CASO DE USO: ENTRADA Y SALIDAS DE MATERIALES .....</i>	<i>76</i>

FIGURA Nº 3. 11 DIAGRAMA DE CASO DE USO: REPORTE	77
FIGURA Nº 3. 12 MODELO DE CONTENIDOS	78
FIGURA Nº 3. 13 MODELO DE NAVEGACIÓN ADMINISTRADOR (JEFE DE ALMACÉN)	79
FIGURA Nº 3. 14 MODELO DE NAVEGACIÓN FUNCIONARIOS	80
FIGURA Nº 3. 15 MODELO DE PRESENTACIÓN: ADMINISTRADOR (JEFE DE ALMACÉN)	81
FIGURA Nº 3. 16 MODELO DE PRESENTACIÓN: FUNCIONARIO	82
FIGURA Nº 3. 17 AUTENTICACIÓN INICIAL CON EL SISTEMA	83
FIGURA Nº 3. 18 INTERFAZ DE ÁREAS (ADMINISTRADOR)	83
FIGURA Nº 3. 19 INTERFAZ CARGOS (ADMINISTRADOR)	84
FIGURA Nº 3. 20 INTERFAZ FUNCIONARIO (ADMINISTRADOR)	84
FIGURA Nº 3. 21 INTERFAZ USUARIO (ADMINISTRADOR)	85
FIGURA Nº 3. 22 REPORTE DE PROVEEDORES	85
FIGURA Nº 3. 23 INTERFAZ CONTROL DE MATERIALES- CATEGORÍA	86
FIGURA Nº 3. 24 INTERFAZ CONTROL DE MATERIALES – MATERIALES	86
FIGURA Nº 3. 25 INTERFAZ CONTROL DE MATERIALES – CONTROL DE ACTIVOS	87
FIGURA Nº 3. 26 INTERFAZ CONTROL DE MATERIALES – ALMACÉN	87
FIGURA Nº 3. 27 INTERFAZ DE ENTRADA Y SALIDA DE MATERIALES – SOLICITUD DE MATERIALES	88
FIGURA Nº 3. 28 INTERFAZ DE ENTRADA Y SALIDA DE MATERIALES – SOLICITUD POR FUNCIONARIO (ADMINISTRADOR)	88
FIGURA Nº 3. 29 INTERFAZ DE ENTRADA Y SALIDA DE MATERIALES – REGISTRO DE INGRESO	89
FIGURA Nº 3. 30 INTERFAZ DE ENTRADA Y SALIDA DE MATERIALES – REGISTRO DE SALIDA	89
FIGURA Nº 3. 31 INTERFAZ DE REPORTE – REPORTE GENERAL	90
FIGURA Nº 3. 32 INTERFAZ DE REPORTE – REPORTE POR PERSONA (ADMINISTRADOR)	90
FIGURA Nº 3. 33 INTERFAZ DE FUNCIONARIO	91
FIGURA Nº 3. 34 INTERFAZ DE MATERIALES– (FUNCIONARIO)	91
FIGURA Nº 3. 35 INTERFAZ DE SOLICITAR MATERIAL– (FUNCIONARIO)	92
FIGURA Nº 3. 36 INTERFAZ DE SOLICITUD– (FUNCIONARIO)	92
FIGURA Nº 3. 37 DATOS DE NAVEGACIÓN	104
FIGURA Nº4. 1 RESULTADOS DE ANÁLISIS DE NAVEGACIÓN DE SISTEMA	103
FIGURA Nº4. 2 DATOS DE NAVEGACIÓN	104
FIGURA Nº4. 3 TABLA DE REGISTRO DE EVENTOS	117





# CAPITULO I

---

## MARCO PRELIMINAR

*INGENIERIA DE SISTEMAS - UNIVERSIDAD PÚBLICA EL ALTO*

# 1. MARCO PRELIMINAR

## 1.1 INTRODUCCIÓN

En la actual modernidad, la informática juega un papel muy importante en el desarrollo de las organizaciones, en el control de las operaciones administrativas y financieras entre otras, la obtención de una información inmediata y eficaz dentro de la estructura organizativa, esto determina el éxito y el alcance de los objetivos con mayor eficacia dentro de la institución. (Flores, 2009)

El presente trabajo de investigación que lleva por título “Sistema de Información Para el Control de Inventario de Almacén” para la (E.S.F.M.T.H.E.A.) **“Escuela Superior de Formación de Maestros Tecnológico Humanístico El Alto”**, aborda una de las más triviales temáticas referente al ámbito educativo y comercial, como es el de llevar un control sobre sus archivos de inventario lo suficientemente eficaz, implicando con éste la automatización de la gestión académica de la institución.

El objetivo primordial del presente proyecto es desarrollar un sistema que permita efectuar un procesamiento de datos eficiente y oportuno para la toma de decisiones en “Escuela Superior de Formación de Maestros Tecnológico Humanístico El Alto”, actualizando los registros que se llevan en la hoja de cálculos Excel y de forma manual y teniendo como propósito final el evitar pérdidas de información y así almacenar los registros de movimiento del material respecto al stock existente en almacenes.

La institución realiza su inventario Logístico según el inventario de anticipación previsión, los inventarios varían en razón de su consumo de cada material que los compone en almacén, lo que da lugar al movimiento de las existencias por ingresos de nuevas cantidades y salida de estas a solicitud de los usuarios, produciendo la rotación de los materiales y la generación de utilidades en función de dicha rotación.

La metodología y herramientas a utilizar para el desarrollo del proyecto son: UWE, porque es una metodología que está orientada a la web, además para la parte del

diseño “Programación” CSS, JavaScript, HTML, PHP, en cuanto a la base de datos como gestor será MySQL, y el XAMPP como servidor. Por último, encontraremos las conclusiones y recomendaciones para ir mejorando el sistema, y se plantee realizar sistemas que gestionen este tipo de actividades no tan sólo a nivel institucional.

El sistema de control de información de inventario en Almacén será de útil ayuda, ofreciendo una alternativa de soluciones y optimizando las actividades dentro de la institución en al área de Almacén.

## **1.2 ANTECEDENTES**

### **1.2.1 Antecedentes de la Institución**

La “Escuela Superior de Formación de Maestros Tecnológico Humanístico El Alto” (E.S.F.M.T.H.E.A.) fue creado el 6 de marzo del año 2006, por Decreto Supremo N° 28625, elevado a rango de Ley N° 3441, y es dependiente del Ministerio de Educación, trabaja formando Maestras y Maestros en el enfoque Socio Comunitario Productivo, recuperando saberes y conocimientos de los pueblos indígena originarios, y promoviendo proyectos vinculados al desarrollo de la Intra-Interculturalidad y Plurilingüismo.

La E.S.F.M.T.H.E.A, a partir de 2007 ejecuta varios proyectos con el propósito de convertirse en un referente nacional de calidad y calidez dentro del marco del enfoque Sociocomunitario Productivo (S.C.P.) y el Modelo Educativo Sociocomunitario Productivo (M.E.S.C.P.) enmarcado en la Ley 070 Avelino Siñani-Elizardo Pérez. (Copyright, 2012).

Actualmente la institución viene realizando su control de inventarios de almacén en la hoja de cálculo office Excel y los trámites de pedido de materiales de forma manual, lo cual ocasiona pérdida de documentación y demora en la entrega del material solicitado por los administrativos para impartir clases y por otro lado no cuenta con un banco de datos de registro de los administrativos e inventarios de almacén.

Como antecedentes de tipo investigativo, se ha podido encontrar y evidenciar proyectos que sin duda han dado un enfoque amplio a la Investigación que se está realizando, entre las cuales se destacan:

### **1.2.2 Antecedentes Internacionales**

- [Patricio Andrés Lerou Godas, 2005] Sistema Para Control De Inventario, Venta Y Generación De Datos Comerciales De Restaurante.
- Objetivo general: “Desarrollar un Sistema para Control de Inventario Venta y Generación de Datos Comerciales para el Restaurante OK Corral” Universidad Austral De Chile Campus Puerto Montt Escuela De Ingeniería En Computación”, Puerto Montt - Chile.
- [William René Guerra Betancourt, 2016] “Sistema De Control De Inventario De Material Clínico Y Académico”.
- Objetivo general: Entregar a la Facultad de Odontología un Sistema que permita administrar y gestionar el inventario de activos no codificados sujetos a control, como es el material académico y odontológico. Universidad Central Del Ecuador, Facultad De Ingeniería, Ciencias Físicas Y Matemática, Carrera De Ingeniería Informática, Quito – Ecuador.
- [Chicaiza Tixilema Diana Ximena, 2015]” Implementación Del Sistema De Control Interno De Pedidos, Manejo De Inventario Y Facturación Para La Facultad De Ingeniería Civil En El Laboratorio De Suelos Y Materiales”.
- Objetivo general: Implementar el control y manejo de inventario, pedidos y facturación con seguridad de ingreso para facilitar el desarrollo de las tareas de los usuarios mediante el seguimiento de los procedimientos ejecutados a través de la automatización de sus procesos. Universidad Central Del Ecuador, Facultad De Ingeniería, Ciencias Físicas Y Matemática, Carrera De Ingeniería Informática, Quito – Ecuador.

### **1.2.3 Antecedentes Nacionales**

- [Grover Gutiérrez Vargas, 2015] “Sistema De Control De Ventas E Inventarios Para Almacenes De Aluminios Utilizando Dispositivos Móviles Caso: Técnica De Aluminio, Vidrio Y Servicios (Talviser)”
- Objetivo: Desarrollar un Sistema de Control de Ventas e Inventarios para Almacenes de Aluminios de la empresa de Técnica de Aluminio, Vidrio y Servicios (TALVISER) mediante el uso de Dispositivos Móviles. Universidad Mayor De San Andrés. (Gutierrez Vargas, 2015).
- [Univ. Wilmer David Callisaya Apaza, 2017] “Software De Gestión Y Control De Inventarios” CASO: AGADON S.R.L.
- Objetivo general: Implementar un Software de gestión y control de ventas e inventarios para el departamento de almacenes de la empresa de Servicios AGADON S.R.L. Universidad Mayor De San Andrés. (Callisaya Apaza, 2017)

## **1.3 PLANTEAMIENTO DEL PROBLEMA**

### **1.3.1 Problema Principal**

Actualmente la institución de formación académica de maestros realiza el manejo y control de material, así como la solicitud de materiales para almacén y stock mínimo disponible en forma manual registrado en cuadernos, luego se transcribe a la hoja de cálculo office Excel, por lo tanto, la información que se dispone no es confiable ni oportuna, lo que ocasiona demoras en el procesamiento de la información del inventario en almacenes, asimismo perdidas económicas e incorrecta toma de decisiones.

### **1.3.2 Problemas Secundarios**

1. No se cuenta con la información respecto a los proveedores, lo que ocasiona pérdida de tiempo en la búsqueda de nuevos proveedores así mismo la perdida de documentos y otros datos por ser clientes recurrentes.

2. No cuenta con la información respecto a los usuarios que solicitan los materiales, por lo que no se puede obtener informes confiables.
3. No se tiene información respecto a los materiales y sus características, ni stock mínimo disponible, lo que ocasiona retrasos en el cumplimiento de algunas solicitudes o generar la acumulación de algunos materiales en exceso.
4. No cuenta con un método para el procesamiento del inventario.
5. No se puede obtener informes de forma periódica siendo que la información no se encuentra centralizada, lo que causa la incorrecta toma de decisiones.
6. No se cuenta con estadísticas de los movimientos en el almacén y se tiene accesos no autorizados a la información ocasionando pérdidas económicas

Por tanto, nos planteamos la siguiente pregunta:

¿De qué manera un **Sistema de Información de Control de Inventario en Almacén** de la Escuela Superior de Formación de Maestros Tecnológico Humanístico El Alto, coadyuvaría en generar la información respecto al movimiento de inventario y stock de los materiales, registro de usuarios y proveedores, de manera confiable y oportuna para una correcta toma de decisiones?

## **1.4. OBJETIVO**

### **1.4.1 Objetivo General**

“Desarrollar un sistema de información de control de inventarios en almacén que permita efectuar un procesamiento de datos eficiente y oportuno para una toma de decisiones en la “Escuela Superior de Formación de Maestros Tecnológico Humanístico El Alto”

### **1.4.2 Objetivos Específicos**

1. Automatizar la información respecto a los proveedores, evitando perder el tiempo buscando nuevos proveedores, así mismo la pérdida de descuentos y otros beneficios por ser clientes frecuentes.
2. Diagnosticar la situación actual del proceso de movimiento de materiales que se realiza dentro el área de almacén.
3. Realizar un análisis a partir del área administrativo de la institución con el propósito de concretar los pedidos de material para tener un mejor control según requerimiento del área de almacén.
4. Diseñar y desarrollar un sistema para el control de inventario de almacén el mismo que pueda ser amigable para los usuarios.
5. Desarrollar un sistema de información que se encargue de generar reportes ágiles, dinámicos y actualizados.
6. Evaluar las pruebas y costos de resultados obtenidos del desarrollo del sistema.

## **1.5 JUSTIFICACIONES**

### **1.5.1 Justificación Técnica**

El presente perfil de grado tiene una justificación técnica a partir de que la institución cuenta con la tecnología adecuada para la implementación del proyecto.

La Institución cuenta con los medios técnicos disponibles como ser su propio servidor, acceso a red; adicionalmente, la institución cuenta con los equipos necesarios para la manipulación del sistema dentro y fuera de la institución.

Para el desarrollo del software se utilizará el lenguaje de programación Php y como gestor de base de datos a MySql, ambos de carácter libre.

### **1.5.2 Justificación Económica**

El presente proyecto de grado se justifica económicamente por el ahorro en material de escritorio, como, por ejemplo, hojas de papel: el cual es empleado para el registro de solicitudes, de inventario y seguimiento en almacén, puesto que esta característica será incorporada en el sistema.

La información generada en forma manual será reducida gracias a la implementación del proyecto y automatización de la misma, logrando reportes en tiempo real que ayuden a la dirección general de la institución a una mejor toma de decisiones, así como la oportuna verificación de datos en el área almacenamiento, cuyo beneficio final es la reducción de costos económicos para la institución.

El sistema se creará en un entorno web y con el gestor de base de datos MySql por su versatilidad y bajo costo económico, ya que debido a su bajo consumo puede ser ejecutado en una maquina con escasos recursos sin ningún problema.

A partir de la justificación económica descrita, la institución contará con información relevante, organizada, estructurada y confiable para los usuarios.

### **1.5.3 Justificación Social**

El presente perfil de proyecto de grado se justifica socialmente ya que ayudará a la institución “E.S.F.M.T.H.E.A.” por medio de su dirección general y administración a interactuar directamente con el área contable y almacenes, acortando y agilizando las solicitudes o pedidos de materiales, así como nuevos requerimientos o compra de materiales simplificando los respectivos tramites evitando recurrir a diferentes oficinas de la institución con los respectivos trámites.

La dirección general de la institución tendrá acceso inmediato a información estadística y reportes en tiempo real, y un pleno conocimiento sobre los procesos de registro en almacén.

Con la ayuda de la web, la implementación del proyecto coadyuvará de gran manera al registro sistematizado de toda la información generada en el área de almacenes.

Se evitará la gran acumulación de papales de pedidos y la reducción del tiempo que lleva hacerlo de forma manual.

## **1.6 METODOLOGÍAS**

### **1.6.1 Metodología UWE**

Es una herramienta que nos permitirá modelar aplicaciones web, utilizada en la ingeniería web, prestando especial atención en sistematización y personalización (sistemas adaptativos). UWE es una propuesta basada en el proceso unificado y UML, pero adaptados 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. (Quiroga, 2015)

### **1.6.2 Normas de Seguridad**

Control de calidad ISO 9126 es un estándar internacional para la evaluación de la calidad del software está dividido en cuatro partes las cuales dirigen: modelo de calidad, métricas externas, métricas internas y calidad en las métricas de uso. Este estándar está pensado para los desarrolladores, adquirentes, personal que asegure la calidad y evaluadores independientes, responsables de especificar y evaluar la calidad del producto software. Por tanto, puede servir para validar la completitud de una definición de requisitos, identificar requisitos de calidad de software, objetivos de diseño y prueba, criterios de aseguramiento de la calidad, etc. (Herrera, 2014)

El **mantenimiento** con la norma **ISO 9001**, la organización debe determinar, proporcionar y mantener la infraestructura necesaria para lograr la conformidad con los requisitos del producto.

La infraestructura incluye, cuando sea aplicable:

- a) Edificios, espacio de trabajo y servicios asociados.
- b) Equipo para los procesos, (tanto hardware como software)
- c) Servicios de apoyo tales (como transporte o comunicación).

La norma de seguridad **ISO/IEC 17799:2005** establece las guías y los principios generales para iniciar, implementar, mantener y proporcionar la gestión de la seguridad en una organización. Se incluyen las mejores prácticas y los objetivos normalmente aceptados en la gestión de la seguridad en la organización. (Juan., 2005).

## **1.7 HERRAMIENTAS**

En el presente proyecto las herramientas a utilizar son:

### **1.7.1 Lenguajes de Programación**

- **PHP:**(acrónimo recursivo de PHP: Hipertext Preprocessor) Es un lenguaje de código abierto muy popular especialmente adecuado para el desarrollo web de contenido dinámico y que puede ser incrustado con HTML5.

PHP se considera uno de los lenguajes más flexibles, potentes y de alto rendimiento conocido hasta el día de hoy y lo que ha ocasionado el interés de múltiples sitios con gran demanda de tráfico como Facebook. Es popular porque un gran número de páginas web o portal web han sido realizados con PHP.

Es código abierto de uso libre y gratuito para todos los programadores que quieran usarlo. La incrustación con HTML5 significa que es un mismo archivo que vamos a poder combinar con PHP y con código HTML5. (PHP5 Orientado a Objetos, 2012), (Lopez D. , 2006).

- **JAVASCRIPT.** - Es un lenguaje de programación, al igual que PHP, si bien tiene diferencias importantes con éste. JavaScript se utiliza principalmente del lado del cliente (es decir, se ejecuta en nuestro ordenador, no en el servidor) permitiendo crear efectos atractivos y dinámicos en las páginas web. Los navegadores modernos interpretan el código JavaScript integrado en las páginas web. La ventaja de JavaScript es que al estar alojado en el ordenador del usuario los efectos son muy rápidos y dinámicos. Al ser un

lenguaje de programación permite toda la potencia de la programación como uso de variables, condicionales, bucles, etc. (Valdés, 2007), (Juan., 2005).

### ➤ **GESTOR DE BASE DE DATOS**

- **MySQL.-** Es un sistema de gestión de bases de datos relacional desarrollado bajo licencia dual: Licencia publica general/Licencia comercial por Oracle Corporation y está considerada como la base datos open source más popular del mundo, y una de las más populares en general junto a Oracle y Microsoft SQL Server, sobre todo para entornos de desarrollo web. (wikipedia, 2012).

### ➤ **SERVIDOR WEB**

- **XAMPP.** - Es un servidor independiente de plataforma, software libre, que consiste principalmente en el sistema de gestionamiento de bases de datos MySQL, el servidor web Apache y los interpretes para lenguajes de script: php y perl. (wikipedia, 2016).

### ➤ **DISEÑO DE INTERFACES**

- **HTML.-** Hace referencia al lenguaje de marcado para la elaboración de páginas web. Es un estándar que sirve de referencia del software que conecta con la elaboración de páginas web en sus diferentes versiones, define una estructura básica y un código (denominado código HTML) para la definición de contenido de una página web, como texto, imágenes, videos, juegos, entre otros. (Born, 2001)
- **Materialize.** - Materialize CSS representa una arquitectura de software, siendo un **framework** dispuesto a facilitar el proceso del diseño web para un escritorio o distintos dispositivos móviles de cualquier tamaño. Se basa en el Material Design y aumenta el ritmo de su desarrollo. (wiki, 2015).

## ➤ EDITORES

- **DREAMWEVER.** - Es la herramienta de diseño de páginas web más avanzada, tal como se ha afirmado en muchos medios. Aunque sea un experto programador de HTML el usuario que lo maneje, siempre se encontrarán en este programa razones para utilizarlo, sobre todo en lo que a productividad se refiere. Cumple perfectamente el objetivo de diseñar páginas con aspecto profesional, y soporta gran cantidad de tecnologías, además muy fáciles de usar: Hojas de estilo y capas, JavaScript para crear efectos e interactividades, inserción de archivos multimedia. (Alvarez, 2001)
- **SUBLIME TEXT.-** Es un editor de texto y editor de código fuente, está escrito en C++ y Python para los plugins. Desarrollado originalmente como una extensión de Vim, con el tiempo fue creando una identidad propia, por esto aún conserva un modo de edición tipo vi llamado Vintage mode.

Se puede descargar y evaluar de forma gratuita. Sin embargo no es software libre o de código abierto y se debe obtener una licencia para su uso continuado, aunque la versión de evaluación es plenamente funcional y no tiene fecha de caducidad. (wikipedia, 2016)

## 1.8. LÍMITES Y ALCANCES

### 1.8.1 Límites

El presente proyecto se limitará a realizar el análisis, diseño y desarrollo del sistema de control de inventarios en almacenes de la institución ESFMTHA y la parte administrativa, dentro el área de almacén limitado a las siguientes condiciones:

- El sistema se limita a realizar el control de registros de ingreso y egreso de materiales y emisión de reportes el mismo que beneficiara únicamente al área de almacén y administración.

- El sistema contará con un control de permisos y accesos para cada rol, generando restricciones de algunas funciones al personal autorizado según el área específica.
- El sistema de control y registro de almacén no está orientado ni vinculado a un sistema de contabilidad.
- El Sistema no está proyectado para realizar transacciones monetarias o pagos vía internet.
- El sistema no contempla y no es parte de un sistema de inventarios de activo fijo de la institución.
- A solicitud del administrador, el sistema no automatizará la asignación de códigos para materiales sean estos nuevos o reasignado

### **1.8.2. Alcances**

El presente proyecto pretende desarrollar un sistema de información y control de inventarios en almacén, el cual permitirá realizar la creación de una base de datos óptima y segura para el registro y almacenamiento de la información del proceso de movimiento de materiales en el área de almacén de la institución y contempla los siguientes alcances:

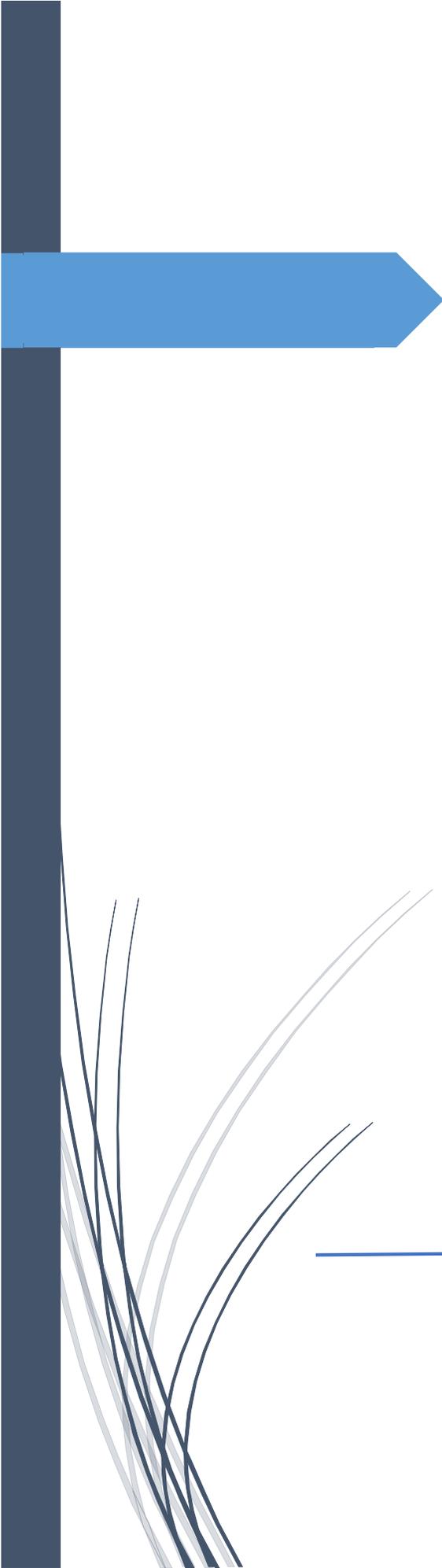
- El sistema permitirá a la parte de almacenes, registrar, guardar y mostrar el stock de todo el material en existencia.
- El sistema creará un módulo de registro de proveedores para la adquisición de materiales.
- El sistema creará un registro de acceso para funcionarios autorizados de acuerdo a necesidades de cada área dentro la institución, para realizar los respectivos requerimientos de materiales.

Para la realización del presente proyecto se tomará en cuenta los siguientes módulos:

- Área
- Cargo
- Funcionarios
- Usuarios
- Proveedores
- Control de Materiales
  - Categoría de materiales
  - Registro de nuevos materiales
  - Cantidad autorizada
  - Stock en almacén
- Entrada y salida de Materiales
  - Solicitud de materiales
  - Solicitud por funcionario
  - Registro de ingreso
  - Registro de salida
- Reportes

## **1.9 APORTES**

Con la implementación del presente proyecto, se pretende beneficiar directamente a la administración de la institución y su área de almacén, el cual permitirá tener un control exclusivo sobre el manejo de los materiales adquiridos, así como requeridos por cada unidad o área dentro la institución, permitiendo agilizar cada una de las actividades académicas impartidas y al mismo tiempo el área de almacenes podrá tener un control óptimo del stock de sus materiales.



# CAPITULO II

---

## MARCO TEORICO

*INGENIERIA DE SISTEMAS - UNIVERSIDAD PÚBLICA EL ALTO*

## 2 MARCO TEORICO

En este capítulo se introducen el concepto más relevante de las metodologías, métodos y herramientas utilizadas para el desarrollo del presente proyecto de grado.

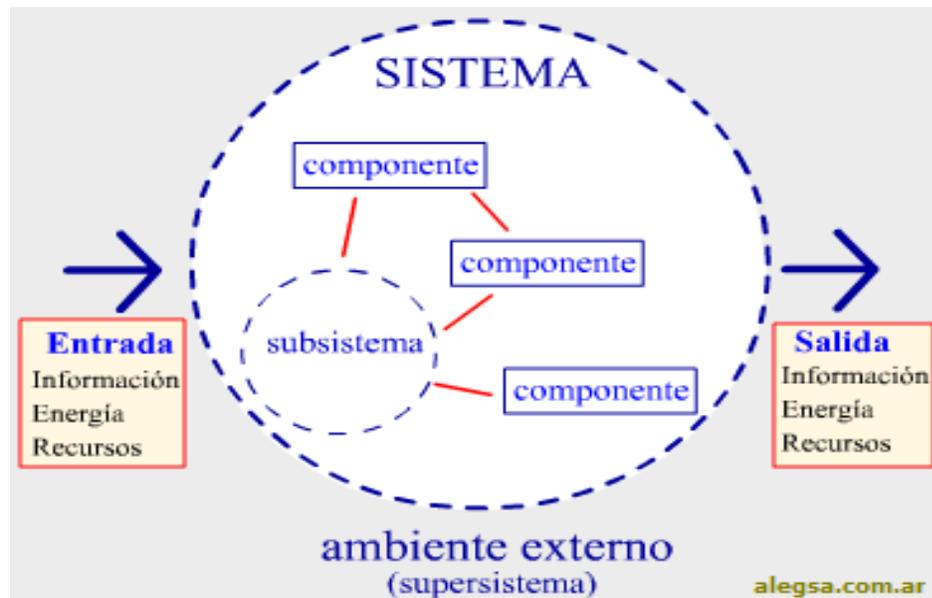
La teoría constituye la base donde se sustentará cualquier análisis o propuesta de desarrollo. A continuación, se desarrollará los siguientes conceptos.

### 2.1 SISTEMA

Sistema es un conjunto de partes o elementos organizados y relacionados que interactúan entre sí para lograr un objetivo. Los sistemas reciben datos, energía o materia del ambiente (entrada) y proveen información, energía o materia (salida) (Alegsa, 2018).

Así mismo, un sistema es la composición de pequeños subsistemas y elementos, el mismo tiene como objetivo lograr un fin en común.

**Figura N° 2. 1 Gráfica General de un sistema**



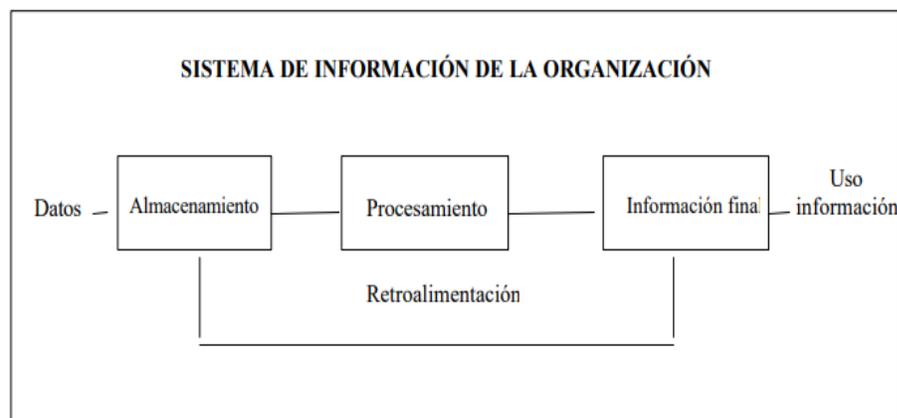
**Fuente:** ( Alegsa, 2018)

## 2.2 SISTEMA DE INFORMACION

Sistema de información es aquel conjunto de componentes interrelacionados que capturan, almacenan, procesan y distribuyen la información para apoyar la toma de decisiones, el control, análisis y visión de una organización.

El sistema de información utiliza como materia prima los datos, los cuales almacena, procesa y transforma para obtener como resultado final información, la cual será suministrada a los diferentes usuarios del sistema, existiendo además un proceso de retroalimentación o “feedback”, en la cual se ha de valorar si la información obtenida se adecua a lo esperado. (Hernandez Trasobares, 2000).

**Figura N° 2. 2 Sistema de información**



**Fuente:** (HERNANDEZ TRASOBARES, 2000)

## 2.3 CONTROL

Control es el proceso de verificar el desempeño de distintas áreas o funciones de una organización. Usualmente implica una comparación entre un rendimiento esperado y un rendimiento observado, para verificar si se están cumpliendo los objetivos de forma eficiente y eficaz. El control permite tomar acciones correctivas cuando sea necesario.

El control es una de las principales actividades administrativas de las organizaciones. El control se relaciona con la planeación, porque el control busca

que el desempeño se ajuste a los planes. El proceso administrativo, desde el punto de vista tradicional, es un proceso circular que se retroalimenta.

## **2.4. INVENTARIO**

Son bienes tangibles que se tienen para la venta en el curso ordinario del negocio o para ser consumidos en la producción de bienes o servicios para su posterior comercialización. Los inventarios comprenden, además de las materias primas, productos en proceso, productos terminados o mercancías, materiales, repuestos y accesorios para ser consumidos en la producción de bienes fabricados, empaques, envases e inventarios en tránsito. Para poder manejar un inventario se debe tomar en cuenta el movimiento de un producto, las causas externas e internas de la empresa, los históricos de ventas, etc., de tal manera que se pueda tener un stock mínimo que no aumente costos de almacenamiento, y tener un balance entre la atención al cliente y los activos de la empresa.

### **2.4.1 Funciones del control de inventarios**

#### **2.4.1.1 Las principales funciones son:**

1. Mantener un registro actualizado de las existencias. La periodicidad depende de unas empresas a otras y del tipo de producto.
2. Informar del nivel de existencias, para saber cuándo se debe de hacer un pedido y cuanto se debe de pedir de cada uno de los productos.
3. Notificar de las situaciones anormales, que pueden constituir síntomas de errores o de un mal funcionamiento del sistema.
4. Elaborar informes para la dirección y para los responsables de los inventarios ( Aranguren, S.F).

#### **2.4.1.2 Método de Inventario PEPS**

Procedimiento contable de valuación de inventario, a través del cual los primeros artículos que ingresan al stock son los primeros en salir.

**Definición:**

El método PEPS (primero entrado, primero salido) o FIFO (first input, first output) por sus siglas en inglés, es un procedimiento contable de valuación de inventario donde los primeros artículos que ingresan al stock son los primeros en salir.

Se aplica generalmente a productos perecederos o artículos con fecha de vencimiento. Cuando se realiza una venta, el costo asociado se calcula en base al valor histórico de los primeros artículos incorporados al inventario.

Esta técnica se usa mucho en los productos frescos evitando que se malogren.

**Características del método PEPS**

Las características principales del método PEPS son las siguientes:

Realiza una valuación ordenada cronológicamente y establece el orden de salida de las mercaderías (primero las más antiguas).

Refleja con mayor precisión los costos de reemplazo del inventario por basarse en costos de adquisiciones recientes.

El método PEPS realiza una valuación ordenada y cronológicamente.

Cuando existe una inflación en las mercaderías se juntan los últimos costos de adquisición.

Su costo de inventario es muy bajo que los demás métodos de "Kardex", también influye en las ganancias de las existencias.

Tiene una estructura de salida de mercadería muy adecuada.

**Objetivos:**

Cuando se retiene el inventario que no sale en un corto plazo logra ser vendido.

El costo del inventario requiere ser definido.

Al calcular debidamente el inventario final, podrá realizar debidamente los estados financieros.

➤ **Ventajas**

Las ventajas de utilizar el método PEPS son las siguientes:

Reporta una mayor ganancia para la empresa dado que al venderse los artículos que ingresaron anteriormente, el costo de las mercaderías vendidas es menor.

Reduce el mantenimiento de registros históricos porque las existencias que salen primero son las más antiguas.

Cuando el estado de resultados es menor su utilidad bruta es mayor.

Cuando se genera una inflación se toma en cuenta las existencias con los costos altos, observándose un valor mayor en el estado de resultados, y así valorando el precio de los activos en el mercado actual.

Los inventarios que se venden se calculan con los costos de su compra, para que llegue a la conclusión de que su estado de resultado sea menor.

➤ **Desventajas**

Las desventajas del método PEPS son las siguientes:

Se pagan más impuestos debido a que el sistema PEPS produce una mayor ganancia.

No hay garantía de que los artículos más antiguos se vendan primero, lo que podría hacer que el producto llegase a su fecha de vencimiento antes de que se venda.

Los costos de producción y ventas son menores, porque muestran el incremento de las utilidades, originando un mayor impuesto.

Cuando exista una inflación la utilidad resulta ser muy exagerada porque compara los costos de compras pasadas con los precios de ventas actuales.

Los costos de inventarios con el paso del tiempo, su ingreso tributario suele ser más alto que el costo del inventario inicial.

Cuando la utilidad es alta influye en un mayor pago de impuestos. (Económica, 20)

## **2.5 ALMACEN**

El almacén es un lugar especialmente estructurado y planificado para custodiar, proteger y controlar los bienes de activo fijo o variable de la empresa, antes de ser requeridos para la administración, la producción o a la venta de artículos o mercancías.

Es necesario informar a control de inventarios y contabilidad todos los movimientos del almacén (entradas y salidas) y a programación de y control de producción sobre las existencias. Se debe asignar una identificación a cada producto y unificarla por el nombre común y conocido de compras, control de inventario y producción.

Toda operación de entrada o salida del almacén requiere documentación autorizada según sistemas existentes. La entrada al almacén debe estar prohibida a toda persona que no esté asignada a él, y estará restringida al personal autorizado por la gerencia o departamento de control de inventarios.

### **2.5.1 Funciones del Almacén**

La manera de organizar y administrar los almacenes depende de varios factores tales como el tamaño, el grado de descentralización deseado, la variedad de productos fabricados, la flexibilidad relativa de los equipos y facilidades de manufactura. La programación de la producción establece los parámetros que serán claves para los diseños futuros de los almacenes. Para proporcionar un servicio eficiente, las siguientes funciones son comunes a todo tipo de almacenes:

- Recepción de materiales
- Registro de entradas y salidas del Almacén.
- Almacenamiento de materiales.
- Mantenimiento de materiales y de almacén.

- Despacho de materiales.
- Coordinación del almacén con los departamentos de control de inventarios y contabilidad. (spcgroup, 2014)

## **2.6 INGENIERIA DE REQUERIMIENTOS**

La ingeniería de requerimientos (IR) cumple un papel primordial en el proceso de producción del software, ya que se enfoca en un área fundamental: la definición de lo que se desea producir, su principal tarea consiste en la generación de especificaciones correctas que describan con claridad, sin antigüedades, en la forma consistente y compacta las necesidades de los usuarios o clientes de esta manera se pretende minimizar los problemas relacionados por mala gestión de los requerimientos en el desarrollo de sistemas.

Los requerimientos de software pueden dividirse en dos categorías: requerimientos funcionales y no funcionales. (Arias, 2006)

La ingeniería de requerimientos es elemental para el desarrollo de un sistema ya que se basa en la definición de lo que se desea producir y cómo será el producto final, es necesario para dar solución a los problemas de la institución.

### **2.6.1 Requerimientos Funcionales**

Los requerimientos funcionales son los que definen funciones que el sistema será capaz de realizar, describen las transformaciones que el sistema realiza sobre las entradas para producir salidas, es importante que se describa el ¿Qué? Y no el ¿Cómo? se debe hacer las transformaciones.

Estos requerimientos al tiempo que avanza el proyecto de software se convierten en los algoritmos, la lógica y gran parte del código del sistema (Alvarado, 2012).

### **2.6.2. Requerimientos No Funcionales**

Los requerimientos no funcionales, como su nombre sugieren, son aquellos requerimientos que no se refieren directamente a las funciones específicas que

proporciona el sistema, sino a las propiedades emergentes de este como la fiabilidad, el tiempo de respuesta y la capacidad de almacenamiento. De forma alternativa, definen las restricciones del sistema como la capacidad de los dispositivos de entrada/salida y las representaciones de datos que se utilizan en las interfaces del sistema. (Arias, 2006)

Esto significa que a menudo son más críticos que los requerimientos funcionales. Los usuarios del sistema normalmente pueden encontrar formas de trabajar alrededor de una función del sistema que realmente no cumple sus necesidades. Sin embargo, el incumplimiento de un requerimiento no funcional puede significar que el sistema entero sea inutilizable.

Los requerimientos no funcionales a su vez se asocian con características particulares del sistema. Más bien, estos requerimientos especifican o restringen las propiedades emergentes del sistema.

Los tipos de requerimientos no funcionales son:

- ✓ **Requerimientos del producto**, estos requerimientos especifican el comportamiento del producto. Algunos ejemplos son los requerimientos de rendimiento en la rapidez de ejecución del sistema y cuanta memoria se requiere; los requerimientos de fiabilidad que fijan la tasa de fallos para que el sistema sea aceptable; los requerimientos de portabilidad, y los requerimientos de usabilidad.
  
- ✓ **Requerimiento Organizacional**, estos requerimientos se derivan de políticas y procesamientos existentes en la organización del cliente y en la del desarrollador. Algunos ejemplos son estándares en los procesos que deben utilizarse; los requerimientos de implementación, como los lenguajes de programación o el método de diseño a utilizar, y los requerimientos de entrega que especifican cuando se entregará el producto y su documentación.

- ✓ **Requerimientos Externos**, este gran apartado incluye todos los requerimientos que se derivan de los factores externos al sistema y de su proceso de desarrollo. Estos pueden incluir los requerimientos de interoperabilidad que definen la manera en que el sistema interactúa con sistemas de otras organizaciones; los requerimientos legislativos que deben seguirse para asegurar que el sistema funcione dentro de la ley, y los requerimientos éticos. (Montesinos, 2014)

Los requerimientos no funcionales se dividen en: Requerimientos del producto, determinan cuanta memoria se requiere y rapidez de ejecución del sistema; Requerimientos Organizacional, son los lenguajes de programación los métodos el diseño a utilizar; Requerimientos externos, definen la manera en que el sistema interactúa con sistemas de otras organizaciones para verificar que el sistema funcione dentro de la ley.

## **2.7 INGENIERIA DE SOFTWARE**

Con objeto de elaborar software listo para enfrentar los retos del siglo XXI, el lector debe aceptar algunas realidades sencillas:

- El software se ha incrustado profundamente en casi todos los aspectos de nuestras vidas y, como consecuencia, el número de personas que tienen interés en las características y funciones que brinda una aplicación específica ha crecido en forma notable. Cuando ha de construirse una aplicación nueva o sistema incrustado, deben escucharse muchas opiniones. Y en ocasiones parece que cada una de ellas tiene una idea un poco distinta de cuáles características y funciones debiera tener el software. Se concluye que debe hacerse un esfuerzo concertado para entender el problema antes de desarrollar una aplicación de software.
- Los requerimientos de la tecnología de la información que demandan los individuos, negocios y gobiernos se hacen más complejos con cada año que pasa. En la actualidad, grandes equipos de personas crean programas de

cómputo que antes eran elaborados por un solo individuo. El software sofisticado, que alguna vez se implementó en un ambiente de cómputo predecible y auto contenido, hoy en día se halla incrustado en el interior de todo, desde la electrónica de consumo hasta dispositivos médicos o sistemas de armamento. La complejidad de estos nuevos sistemas y productos basados en computadora demanda atención cuidadosa a las interacciones de todos los elementos del sistema. Se concluye que el diseño se ha vuelto una actividad crucial.

- Los individuos, negocios y gobiernos dependen cada vez más del software para tomar decisiones estratégicas y tácticas, así como para sus operaciones y control cotidianos. Si el software falla, las personas y empresas grandes pueden experimentar desde un inconveniente menor hasta fallas catastróficas. Se concluye que el software debe tener alta calidad.
- A medida que aumenta el valor percibido de una aplicación específica se incrementa la probabilidad de que su base de usuarios y longevidad también crezcan. Conforme se extienda su base de usuarios y el tiempo de uso, las demandas para adaptarla y mejorarla también crecerán. Se concluye que el software debe tener facilidad para recibir mantenimiento.

La ingeniería de software es una tecnología con varias capas, cualquier enfoque de ingeniería (incluso la de software) debe basarse en un compromiso organizacional con la calidad. La administración total de la calidad, Six Sigma y otras filosofías similares alimentan la cultura de mejora continua, y es esta cultura la que lleva en última instancia al desarrollo de enfoques cada vez más eficaces de la ingeniería de software. El fundamento en el que se apoya la ingeniería de software es el compromiso con la calidad. El fundamento para la ingeniería de software es la capa proceso. El proceso de ingeniería de software es el aglutinante que une las capas de la tecnología y permite el desarrollo racional y oportuno del software de cómputo. El proceso define una estructura que debe establecerse para la obtención eficaz de tecnología de ingeniería de software. El proceso de software forma la base para el control de la administración de proyectos de software, y establece el contexto en el

que se aplican métodos técnicos, se generan productos del trabajo (modelos, documentos, datos, reportes, formatos, etc.), se establecen puntos de referencia, se asegura la calidad y se administra el cambio de manera apropiada. Los métodos de la ingeniería de software proporcionan la experiencia técnica para elaborar software. Incluyen un conjunto amplio de tareas, como comunicación, análisis de los requerimientos, modelación del diseño, construcción del programa, pruebas y apoyo. Los métodos de la ingeniería de software se basan en un conjunto de principios fundamentales que gobiernan cada área de la tecnología e incluyen actividades de modelación y otras técnicas descriptivas. Las herramientas de la ingeniería de software proporcionan un apoyo automatizado o semiautomatizado para el proceso y los métodos. Cuando se integran las herramientas de modo que la información creada por una pueda ser utilizada por otra, queda establecido un sistema llamado ingeniería de software asistido por computadora que apoya el desarrollo de software. (Pressman, 2010)

**Figura N° 2. 3 Fases de ingeniería de software**



**Fuente:** (Pressman, 2010)

## **2.8 INGENIERIA WEB**

La ingeniería web es la aplicación de metodologías sistemáticas, disciplinadas y cuantificables al desarrollo eficiente, operación y evolución de aplicaciones web de alta calidad en la World Wide Web.

En este sentido la ingeniería web hace referencia a las metodologías, técnicas y herramientas que se utilizan en el desarrollo de aplicaciones web complejas y de

gran dimensión en las que se apoyan la evolución, diseño, desarrollo, implantación y evolución de dichas aplicaciones. (R., Pressman, 2000)

### **2.8.1 Características de la Ingeniería Web**

La ingeniería web utiliza principios de la ingeniería de software, incluye nuevos enfoques, metodologías, herramientas, técnicas, guías y patrones para cubrir los requisitos únicos de Aplicaciones Web. (Pressman, 2010)

Los principales aspectos de la ingeniería web incluyen entre otros los siguientes temas:

- ✓ Desarrollo web colaborativo.
- ✓ Modelo conceptual de aplicaciones web.
- ✓ Diseño de modelos de datos para sistemas de información WEB.
- ✓ Ingeniería web empírica.
- ✓ Ingeniería de requisitos para aplicaciones web.
- ✓ Usabilidad de aplicaciones web.
- ✓ Accesibilidad para la web.
- ✓ Metodologías de diseño web.
- ✓ Diseño de interfaces de usuario.
- ✓ Desarrollo y despliegue de servidores web.

El desarrollo de aplicaciones web, posee determinadas características que lo hacen diferente del desarrollo de aplicaciones o software tradicional y sistemas de información.

## **2.9 METODOLOGIA DE INGENIERIA WEB**

Una metodología es un conjunto de procedimientos, técnicas, herramientas y un soporte documental que ayuda a los desarrolladores a realizar un nuevo software.

Una metodología de desarrollo de software se refiere al entorno que usa para estructurar, planificar y controlar el proceso de desarrollo de un sistema de información. (Pressman, 2010)

### **2.9.1. 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, basado en las técnicas de UML, la notación de UML y mecanismos de extensión de UML.

Entre los principales modelos de UWE podemos citar el modelo lógico-conceptual, modelo navegacional, modelo de presentación, visualización de escenarios web y la interacción temporal entre los diagramas: diagramas de estado, secuencia, colaboración y actividad.

### **2.9.2. Características de la Metodología UWE**

UML es una técnica de modelamiento propietaria, es la aceptación de UML en el proceso de desarrollo de software, la flexibilidad para la definición de un lenguaje de modelamiento específico en el dominio WEB, también llamado perfil UML.

UWE hace uso de notación UML pura y los tipos de diagramas UML en donde sea posible para el análisis y diseño de aplicaciones WEB.

### **2.9.3. Modelo de la Metodología De UWE**

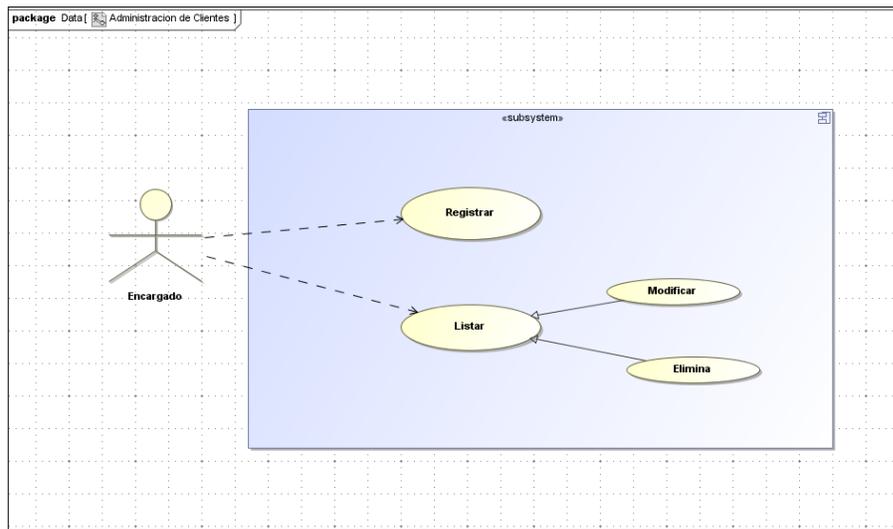
El modelo que propone UWE está compuesto por submodelos.

#### **2.9.3.1. Modelo de Casos de USO**

Dentro de este ciclo se realiza un análisis del sistema construyendo para ello diagramas de casos de uso en una descripción de los pasos o las actividades que deberían realizarse para llevar a cabo algún proceso.

Los personajes o entidades que participaran en un caso de uso se denominan actores UWE provee diferentes estereotipos. Un caso de uso especifica el comportamiento de un sistema o una parte del mismo, y es una descripción de un conjunto de secuenciales de acciones, donde cada secuencia representa la interacción de los elementos externos del sistema (actores) con el propio sistema.

**Figura N° 2.4 Diagrama de casos de uso**

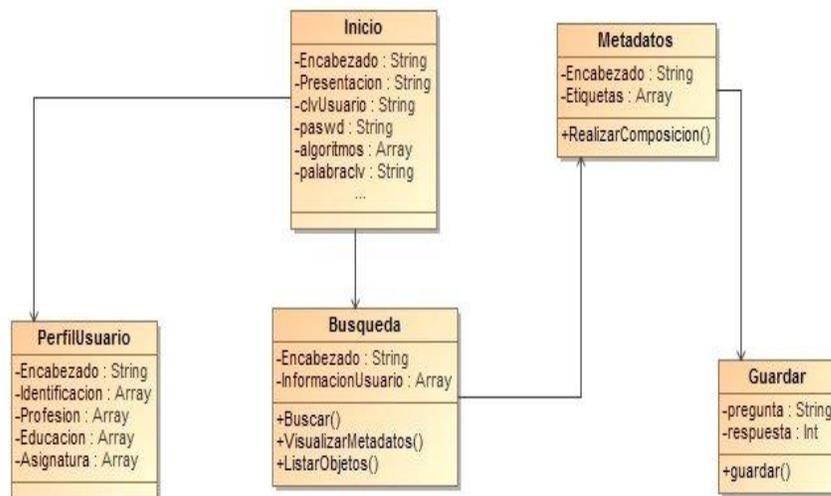


**Fuente:** (Alvarez, 2001)

**2.9.3.2. Modelo de Contenido**

El diseño conceptual está basado en el análisis de requerimientos del paso previo a los objetos involucrados en la interacción entre usuario y la aplicación, especificando en los casos de uso. Apunta a la construcción de modelos de clase con objetos, que intentan ignorar tanto como sea posible los caminos de navegación y los pasos de presentación.

**Figura N° 2. 5 Diagrama de contenido**



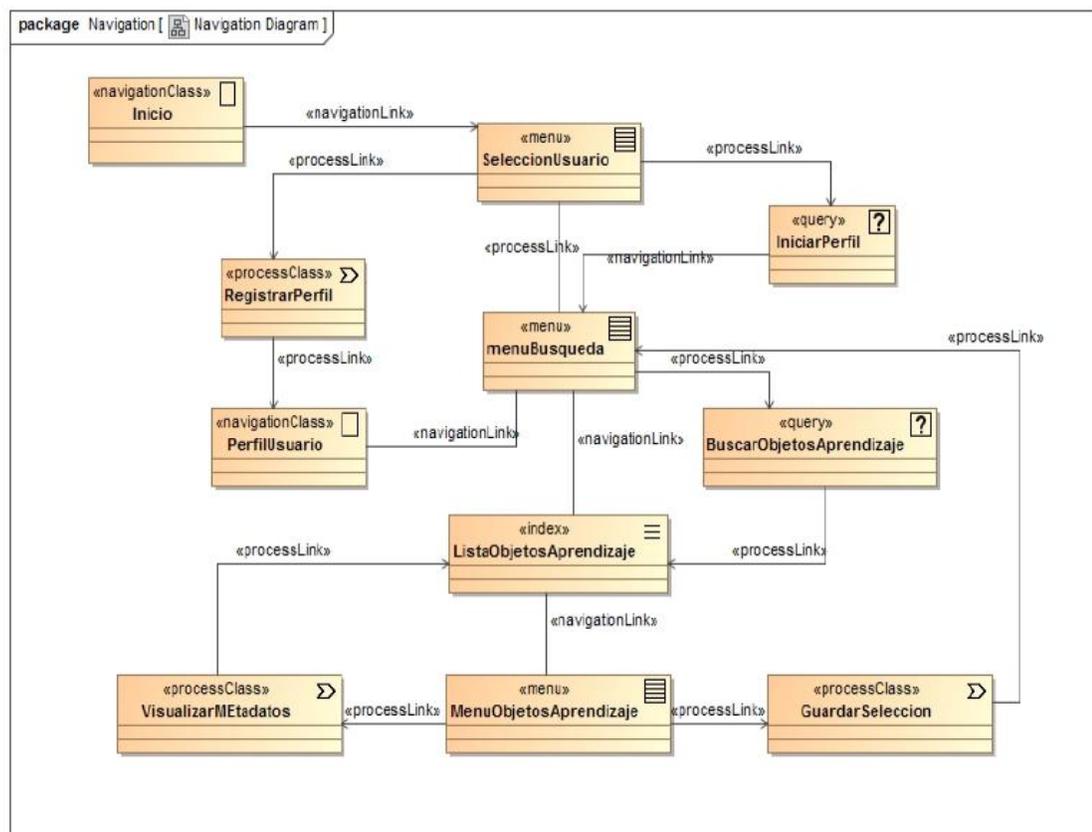
**Fuente:** (Proyecto fin de master (pdf), 2016)

### 2.9.3.3. Modelo de Navegación

El modelo de espacio de navegación es construido con las clases de navegación y las asociaciones de navegación y están representadas gráficamente de clase de UML(Koch,2000).

La clase de navegación modela una clase cuyas instancias son visibles por usuarios durante la navegación se le asigna el nombre que se diera a las correspondientes clases conceptuales, sin embargo se diferencia de este estereotipo, además una clase de navegación puede contener atributos de otras clases de modelo conceptual, siempre que la clase de navegación tenga una asociación con la clase de la que se presta o los atributos, para diferenciar dichos atributos se coloca una barra inclinada a la derecha antes del nombre.

**Figura N° 2. 6 Diagrama de navegación**



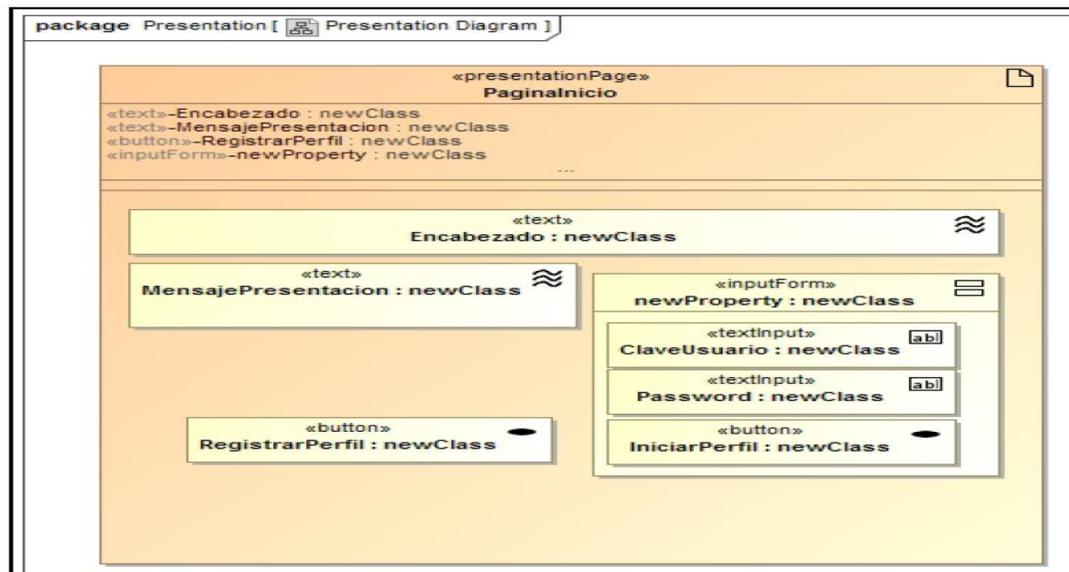
Fuente: (Alvarez, 2001)

#### 2.9.3.4. Modelo de Presentación

El modelo de presentación ofrece una visión abstracta de la interfaz de usuario de aplicaciones web. Se basa en el modelo de navegación y en los aspectos concretos de la interfaz de usuario (IU).

Describe la estructura básica de la IU es decir los elementos de la interfaz de usuario (texto, imágenes, enlaces, formularios), su ventaja es que es independiente de las técnicas actuales que utilizan para implementar un sitio web, lo que permite a las partes interesadas discutir la convivencia de la presentación antes de que realmente se aplique.

**Figura N° 2. 7 Diagrama de presentación**



Fuente: (Flores, 2009)

#### 2.10. MODELO DE IMPLEMENTACION

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.

### **2.10.1 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 palabras simples 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 información, las necesidades de navegación, 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 prototipado de la interfaz de usuario.

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

**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 de la fase de pruebas de control.

**Figura N° 2. 8 Fases de la Metodología de UWE**



**Fuente:** (R., Pressman, 2000)

## 2.11 HERRAMIENTAS

### 2.11.1 Servidor Apache

El servidor HTTP Apache es un servidor web HTTP de código abierto, para plataformas Unix (BSD, GNU/Linux, etc.), Microsoft Windows, Macintosh y otras, que implementa el protocolo HTTP y la noción de sitio virtual. Además, Apache consistía solamente en un conjunto de parches a aplicar al servidor de NCSA. En inglés, a patchy server (un servidor "parcheado") suena igual que Apache Server. El servidor Apache es desarrollado y mantenido por una comunidad de

usuarios bajo la supervisión de la Apache Software Foundation dentro del proyecto HTTP Server (httpd).

Apache presenta entre otras características altamente configurables, bases de datos de autenticación y negociado de contenido, pero fue criticado por la falta de una interfaz gráfica que ayude en su configuración.

Disponible en: (Alegsa, alegsa.com.ar, 2018)

### **2.11.2 Base De Datos**

El modelo de base de datos entidad-relacional (ER) permite describir los datos implicados en empresas reales en término de objeto y de sus relaciones, y se emplea mucho para desarrollar el diseño preliminar de la base de datos, aporta conceptos útiles que permiten pasar de una descripción informal de lo que los usuarios desean de su base de datos a otra más detallada y precisa que se pueda implementar.

#### **2.11.2.1 Gestor de Base de Datos MYSQL**

Una base de datos es una colección estructurada de tablas que contienen datos. Esta puede ser desde una simple lista de compras a una galería de pinturas o el vasto volumen de información en una red corporativa. Para agregar, acceder a y procesar datos guardados en un computador, usted necesita un administrador como MySQL Server.

MySQL es un sistema de gestión de bases de datos relacional desarrollado bajo licencia dual GPL/Licencia comercial por Oracle Corporation y está considerada como la base datos open source más popular del mundo.

Dado que los computadores son muy buenos manejando grandes cantidades de información, los administradores de bases de datos juegan un papel central en computación, como aplicaciones independientes o como parte de otras aplicaciones. Una base de datos relacional archiva datos en tablas separadas en vez de colocar todos los datos en un gran archivo. Disponible en: (MySQL revista).

### **2.11.3 Lenguaje De Programación PHP**

El código es interpretado por un servidor web con un módulo de procesador de PHP que genera la página web resultante. PHP es un lenguaje de programación de propósito general de código del lado del servidor originalmente diseñado para el desarrollo web de contenido dinámico. Fue uno de los primeros lenguajes de programación del lado del servidor que se podían incorporar directamente en el documento HTML en lugar de llamar a un archivo externo que procese los datos. PHP ha evolucionado por lo que ahora incluye también una interfaz de línea de comandos que puede ser usada en aplicaciones gráficas independientes.

Puede ser usado en la mayoría de los servidores web al igual que en casi todos los sistemas operativos y plataformas sin ningún costo. Disponible en: (Porto, 2012)

### **2.11.4 Herramientas de Diseño**

#### **2.11.4.1 HTML**

HTML no es un lenguaje de programación, HTML es un lenguaje de marcado de hipertexto o “HyperText Markup Language” por el desarrollo de sus iniciales en inglés, básicamente este lenguaje se escribe en su totalidad con elementos, estos elementos están constituidos por etiquetas, contenido y atributos, que explicaremos de una manera más detallada en algunas líneas más abajo.

HTML es un lenguaje que interpreta el navegador web para mostrar los sitios o aplicaciones web tal y como estamos acostumbrados.

#### **2.11.4.2 Javascript**

JavaScript, llamado script, se introduce directamente en el documento HTML y no necesita ser compilado, es el propio navegador el que se encarga de traducir dicho código. Gracias a JavaScript podemos desarrollar programas que se ejecuten directamente en el navegador (cliente) de manera que este puede efectuar determinadas operaciones o toma de decisiones sin necesidad de acceder al servidor. JavaScript es un lenguaje de programación creado por Netscape con el

objetivo de integrarse en HTML y facilitar la creación de páginas interactivas sin necesidad de utilizar script. (Juan., 2005)

### **2.11.4.3 CSS3**

CSS significa Cascade Style Sheets, también llamado Hojas de Estilo en Cascada. CSS es un lenguaje de marcado que se emplea para dar formato a un sitio web. Es decir, funciona en conjunto con los archivos HTML. Por esta razón, para crear un sitio web debes saber tanto HTML como CSS.

Podemos indicar propiedades como el color, el tamaño de la letra, el tipo de letra, si es negrita, si es itálica, también se puede dar forma a otras cosas que no sean letras, como colores de fondo de una página, tamaños de un elemento.

Disponible en: (Ecured, 2012).

### **2.11.5 Framework Angular**

Angular es un framework de desarrollo para JavaScript creado por Google. La finalidad de Angular es facilitarnos el desarrollo de aplicaciones web SPA y además darnos herramientas para trabajar con los elementos de una web de una manera más sencilla y óptima.

Otro propósito que tiene Angular es la separación completa entre el front-end y el back-end en una aplicación web. (Robles, 2017)

### **2.11.6 Framework Materialize**

Materialize CSS. Concebido especialmente para proyectos que hagan del Material Design su bandera, este framework CSS nos permite ahorrar mucho tiempo y esfuerzo al implementar y optimizar nuestros proyectos web. No sólo contiene multitud de clases CSS ya configuradas, además incorpora código JavaScript para añadir a nuestras interfaces.

Las ventajas de manejo de Materialize CSS se reflejan en:

- El tiempo de desarrollo es menor, pues la mayor parte del código nos lo encontramos ya escrito.

- Si no somos muy hábiles en el diseño podemos aplicar un framework CSS para conseguir una bonita estética en nuestros proyectos.
- Los diseños son más robustos y la estética final es más homogénea.
- Materialize sólo ocupa aproximadamente unos 140 KB con su CSS, a lo que hay que añadirle 180 KB con el JS. (ARSYS, 2019)

## 2.12 PRUEBAS DE SOFTWARE

Las pruebas de software tienen los siguientes objetivos:

- Descubrir un error.
- Mostrar un error no descubierto hasta ese momento.
- Descubrir un error no detectado hasta ese momento.

Las pruebas tienen los siguientes principios:

- Las pruebas deben tener un seguimiento hasta los requisitos del cliente.
- La prueba debe planificarse antes de que empiecen.
- Es aplicable el principio de Pareto a la prueba del software (El 80% de los errores está en el 20% de los módulos, hay que identificar esos módulos y probarlos muy bien).
- No es posible las pruebas exhaustivas.

Un software debe ser fácil de probar, para lo cual se puede tener en cuenta las siguientes características que propone Pressman:

Característica	Observación
<b>Operatividad</b>	<p>Cuánto mejor funcione, más eficientemente se puede probar</p> <ul style="list-style-type: none"> <li>• El sistema tiene pocos errores</li> <li>• Ningún error bloquea la ejecución de las pruebas</li> <li>• El producto evoluciona en fases funcionales</li> </ul>
<b>Observabilidad</b>	<p>Lo que se ve es lo que se prueba</p> <ul style="list-style-type: none"> <li>• Se genera una salida distinta para cada entrada</li> <li>• Todos los factores que afectan a los resultados están visibles</li> <li>• Un resultado incorrecto se identifica fácilmente</li> <li>• Los errores internos se detectan automáticamente</li> <li>• Se informa automáticamente de los errores internos</li> <li>• El código fuente es accesible.</li> </ul>

<b>Controlabilidad</b>	<p>Cuánto mejor se pueda controlar el software, más se puede automatizar y optimizar</p> <ul style="list-style-type: none"> <li>• Todo el código es ejecutable a través de alguna combinación de entrada</li> <li>• El ingeniero de pruebas puede controlar los estados y las variables del hardware y software</li> <li>• Los formatos de las entradas y los resultados son consistentes y estructurados</li> </ul>
<b>Capacidad de descomposición</b>	<p>Controlando el ámbito de las pruebas, podemos aislar más rápidamente los problemas y llevar a cabo pruebas de regresión</p> <ul style="list-style-type: none"> <li>• El software está construido con módulos independientes</li> <li>• Los módulos de software se pueden probar independientemente.</li> </ul>
<b>Simplicidad</b>	<p>Cuánto menos haya que probar, más rápidamente se puede probar</p> <ul style="list-style-type: none"> <li>• Simplicidad funcional</li> <li>• Simplicidad estructural</li> <li>• Simplicidad del código</li> </ul>

<b>Característica</b>	<b>Observación</b>
<b>Estabilidad</b>	<p>Cuanto menos cambios, menos interrupciones a las pruebas</p> <ul style="list-style-type: none"> <li>• Los cambios del software son infrecuentes</li> <li>• Los cambios del software están controlados</li> <li>• Los cambios del software no invalidan las pruebas existentes</li> <li>• El software se recupera bien de los fallos</li> </ul>
<b>Facilidad de comprensión</b>	<p>Cuanta más información se tenga, más inteligentes eran las pruebas</p> <ul style="list-style-type: none"> <li>• El diseño se ha entendido perfectamente</li> <li>• Las dependencias entre los componentes internos, externos y compartidos se han entendido perfectamente</li> <li>• Se han comunicado los cambios del diseño</li> <li>• La documentación técnica es accesible</li> </ul>

### 2.12.1 Diseño de casos de prueba:

El diseño de casos de prueba, tiene un único objetivo: tener la mayor probabilidad de encontrar el mayor número de errores con la mínima cantidad de esfuerzo y tiempo posible:

Cualquier software puede aprobarse de una de las siguientes formas:

1. Conociendo la función para la que fue diseñado el producto.

Se pueden utilizar pruebas para: Comprobar su función operativa y buscar errores de cada función.

## 2. Conociendo el funcionamiento del producto:

Se pueden utilizar pruebas para: Comprobar que las operaciones están de acuerdo con las especificaciones y para comprobar que los componentes internos funcionan de forma adecuada.

Las pruebas pueden ser:

- estáticas, que buscan fallos sobre el sistema en reposo, o
- dinámicas que se ejecutan y observa el comportamiento de un proceso (Estimulo-Proceso-Respuesta).

Las pruebas o técnicas de comprobación dinámica, siempre implican la ejecución del programa, permiten:

- Evaluar la calidad de un producto
- Mejorar identificando defectos y problemas

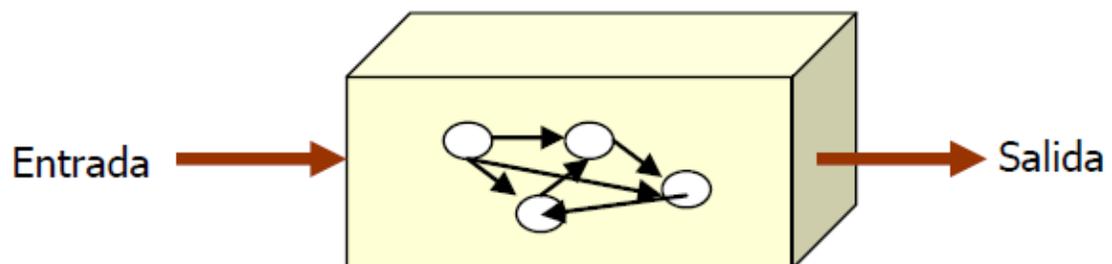
Entre las pruebas dinámicas se tiene dos tipos o dos enfoques diferentes:

- Caja Blanca (Estructural) basado en información sobre como el software ha sido diseñado o codificado.
- Caja Negra (Funcional), los casos de prueba se basan solo en el comportamiento de entrada/salida.

### 2.12.1.1 Pruebas de Caja Blanca

Esta prueba se centra en la estructura interna del programa, en este caso la prueba consiste en probar todos los posibles caminos de ejecución a través de las instrucciones del código, que puedan trazarse.

**Figura N° 2. 9 Estructura caja blanca**



**Fuente:**(Ingenieria de Software, 2011)

Mediante esta prueba, el ingeniero desarrollador del software puede:

1. Garantizar que se recorre por lo menos una vez todos los caminos independientes de cada módulo.
2. Recorrer todas las decisiones lógicas en sus condiciones verdadera y falsa.
3. Recorrer todos los bucles en sus límites y con sus límites operacionales.
4. Recorrer las estructuras internas de datos para asegurar su validez

### **2.12.1.2 Pruebas de Caja Negra**

Consiste en estudiar la especificación de las funciones, la entrada y la salida para derivar los casos. La prueba ideal del software consiste en probar todas las posibles entradas y salidas del programa. Consideran la función específica para la cual fue creado el producto (qué es lo que hace).

Las pruebas se llevan a cabo sobre la interfaz del sistema.

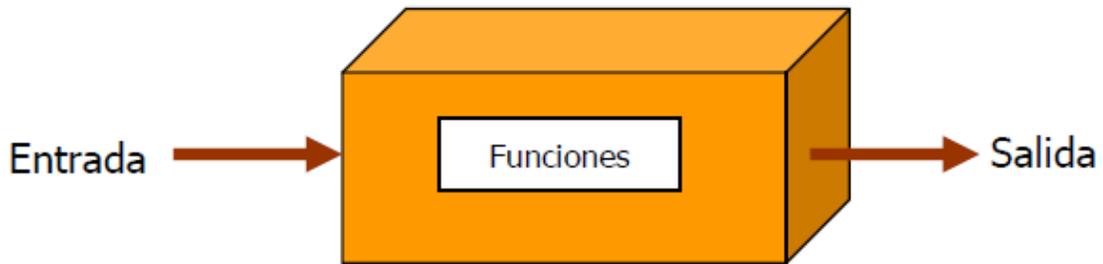
No es una alternativa a la prueba de caja blanca.

Complementan a las pruebas de caja blanca

La prueba de caja negra, también denominados pruebas de comportamiento encuentra errores de:

- Funciones incorrectas o ausentes
- Errores de interfaz
- Errores en estructuras de datos o en accesos a bases de datos externas
- Errores de rendimiento
- Errores de inicialización y de terminación

**Figura N° 2. 10 Estructura caja negra**



**Fuente:** (Ingeniería de Software, 2011)

### **Técnicas de prueba de Caja negra**

- Partición equivalente.
- Análisis de valores límite.

#### **Prueba de partición equivalente •**

- Se basa en la división de los parámetros de entrada en un conjunto de clases de datos denominadas clases de equivalencia.
- Se parte de la premisa de que cualquier elemento de una clase de equivalencia es representativo del resto del conjunto.
- A partir de las clases de equivalencia se obtienen las clases de equivalencias válidas e inválidas.

#### **Prueba de Valores Límite (VL)**

- Se basa en la evidencia experimental de que los errores suelen aparecer con mayor probabilidad en los extremos de los campos de entrada.
- Un análisis de las condiciones límite de las clases de equivalencia aumenta la eficiencia de las pruebas.
- Condiciones límite: valores justo por encima y por debajo de los márgenes de la clase de equivalencia. (A. Goñi, 2018).

## 2.13 METRICAS DE CALIDAD

- ✓ **Calidad de Software:** “La calidad del software es el grado con el que un sistema, componente o proceso cumple los requerimientos especificados y las necesidades o expectativas del cliente o usuario”. (IEEE, Std. 610-1990). “Concordancia del software producido con los requerimientos explícitamente establecidos, con los estándares de desarrollo prefijados y con los requerimientos implícitos no establecidos formalmente, que desea el usuario” (R., Pressman, 2000)
- ✓ **Calidad:** La calidad es un término que actualmente se encuentra en multitud de contextos y con el que se busca despertar en quien lo escucha una sensación positiva, transmitiendo la idea de que algo es mejor. La palabra calidad tiene muchos significados. La calidad de un producto o servicio es la percepción que el cliente tiene del mismo, es una fijación mental del consumidor que asume conformidad con un producto o servicio determinado, que solo permanece hasta el punto de necesitar nuevas especificaciones. La calidad es la capacidad de un producto o servicio para satisfacer las necesidades del cliente o usuario.
- ✓ *Disponible en: (AIGP2 Calidad Informatica, 2011)*
- ✓ **Métrica:** Históricamente se habló de métrica en referencia a los sistemas que existían para escribir versos diferenciados en base al número de sílabas que contenía cada verso, así como en referencia al estudio y “**medición**” de la cantidad de sílabas y estrofas que contenían los versos.
- ✓ En informática, el término métrica hace referencia a la medición del software en base a parámetros predeterminados, como puede ser el número de líneas de código de que consta o el volumen de documentación asociada. A veces en vez de hablar de métrica se usa el término “**Indicadores**” del software. Algunos ingenieros lo usan como sinónimos mientras que otros les atribuyen significados distintos.

Algunas métricas o indicadores pueden ser:

- a) **Índice de productividad = tamaño / esfuerzo = líneas de código generado / horas trabajadas.**
  
- b) **Tasa de defectos = defectos / tamaño = número de errores / líneas de código generadas.**

Disponible en: (Calidad de Software Metricas y Flabilidad de Aplicaciones, s.f.)

- ✓ **Métricas de calidad del software:** Los sistemas de métricas de calidad del software tradicionales se han centrado fundamentalmente en las métricas de procesos, de productos y de recursos.

Los sistemas de métricas hoy en día son los usados en los *profiles* o aplicaciones para probar las aplicaciones. Este tipo de aplicaciones usan sistemas de métricas en tiempo de ejecución para medir tiempos, buscar cuellos de botella en las aplicaciones, medir capacidades máximas, etc.

Así, las métricas tratan de servir de medio para entender, monitorizar, controlar, predecir y probar el desarrollo software y los proyectos de mantenimiento.

Los tres objetivos fundamentales de la medición son (Fenton y Pfleeger, 1997):

1. Entender qué ocurre durante el desarrollo y el mantenimiento.
2. Controlar qué es lo que ocurre en nuestros proyectos.
3. Mejorar nuestros procesos y nuestros productos.

*Disponible en: (Proyecto fin de master (pdf), 2016)*

### **2.13.1 ISO 9000**

La serie de Normas ISO 9000 son un conjunto de enunciados, los cuales especifican que elementos deben integrar el Sistema de Gestión de la Calidad de una Organización y como deben funcionar en conjunto estos elementos para asegurar la calidad de los bienes y servicios que produce la Organización. Al hablar de

Organización nos estamos refiriendo a una Empresa, Compañía o cualquier Estructura Organizada que genere o comercialice productos o servicios de algún tipo:

**Figura N° 2. 11 Norma ISO 9000**



**Fuente:** (Las Normas Iso 9000 (PDF), s.f.)

### **2.13.2. Estándares ISO/IEC 9126**

Esta norma Internacional fue publicada en 1992, la cual es usada para la evaluación de la calidad de software, llamado "Information technology-Software product evaluation-Quality characteristics and guidelines for their use"; o también conocido como ISO 9126 (o ISO/IEC 9126).

Este estándar describe 6 características generales: Funcionalidad, Confiabilidad, Usabilidad, Eficiencia, Mantenibilidad, y Portabilidad.

La norma ISO/IEC 9126 permite especificar y evaluar la calidad del software desde diferentes criterios asociados con adquisición, requerimientos, desarrollo, uso, evaluación, soporte, mantenimiento, aseguramiento de la calidad y auditoria de software. Al unir la calidad interna y externa con la calidad en uso se define un modelo de evaluación más completo, se puede pensar que la usabilidad del modelo de calidad externa e interna pueda ser igual al modelo de calidad en uso, pero no, la usabilidad es la forma como los profesionales interpretan o asimilan la funcionabilidad del software y la calidad en uso se puede asumir como la forma que lo asimila o maneja el usuario final.

Si se unen los dos modelos, se puede definir que los seis indicadores del primer modelo tienen sus atributos y el modelo de calidad en uso sus 4 indicadores pasarían hacer sus atributos.

### **2.13.2.1 Norma de Evaluación ISO/IEC 9126**

Se establecen categorías para las cualidades de la calidad externa e interna y calidad en uso del software, teniendo en cuenta estos 7 indicadores (funcionalidad, confiabilidad, utilidad, eficiencia, capacidad de mantenimiento, portabilidad y calidad en uso), que se subdividen a su vez en varios indicadores; estas se pueden medir por métrica interna o externa.

En la Figura N° 2.10 se puede visualizar la norma 9126 y sus características principales.

**Figura N° 2. 12 Norma de Evaluación de ISO/IEC 9126**



**Fuente:** (Evaluación Interna, externa y Calidad de Uso ISO/IEC 9126)

La norma ISO 9126 contempla la funcionalidad, eficiencia, usabilidad, mantenibilidad, portabilidad y calidad de uso.

Las definiciones se dan para cada característica y sub-característica de calidad del software que influye en la calidad. Para cada característica y sub-característica, la capacidad del software es determinada por un conjunto de atributos internos que pueden ser medidos.

Las características y sub-características se pueden medir externamente por la capacidad del sistema que contiene el software.

## **FUNCIONALIDAD**

Funcionalidad es la capacidad del software de cumplir y proveer las funciones para satisfacer las necesidades explícitas e implícitas cuando es utilizado en condiciones específicas. A continuación, se muestra la característica de Funcionalidad y las sub-características que cubre:

### **Característica de funcionalidad.**

La funcionalidad se divide en 5 criterios:

- ✓ **Adecuación:** La capacidad del software para proveer un adecuado conjunto de funciones que cumplan las tareas y objetivos especificados por el usuario.
- ✓ **Exactitud:** La capacidad del software para hacer procesos y entregar los resultados solicitados con precisión o de forma esperada.
- ✓ **Interoperabilidad:** La capacidad del software de interactuar con uno o más sistemas específicos.
- ✓ **Seguridad:** La capacidad del software para proteger la información y los datos de manera que los usuarios o los sistemas no autorizados no puedan acceder a ellos para realizar operaciones, y la capacidad de aceptar el acceso a los datos de los usuarios o sistemas autorizados
- ✓ **Conformidad de la funcionalidad:** La capacidad del software de cumplir los estándares referentes a la funcionalidad.

La métrica orientada a la función, utiliza una medida de la funcionalidad entregada por la aplicación como un valor de normalización. Ya que la funcionalidad no se puede medir directamente es necesario derivar mediante otras medidas directas como el punto función.

### ✓ **Punto Función**

La métrica de punto de función (PF), se puede usar como medio para predecir el tamaño de un sistema que se va a obtener de un modelo de análisis. Los puntos de función se obtienen utilizando una relación empírica basada en

medidas cuantitativas del dominio de información de software y valorización subjetiva de la complejidad del software.

Para poder determinar la funcionalidad del sistema se debe determinar cinco características del dominio de información.

- **Número de Entradas de Usuario:** Se cuenta cada entrada de usuario que proporciona diferentes datos orientados a la aplicación. Las entradas se deberían diferenciar de las peticiones, las cuales se cuentan de forma separada, estas aplicaciones pueden ser: insertar, actualizar, borrar datos del sistema.
- **Número de Salidas de Usuario:** Se cuenta cada salida que proporciona al usuario información orientada a la aplicación. En este contexto la salida se refiere a informes, datos en pantalla, mensaje de error, etc. Los elementos de datos particulares dentro de un informe no se cuentan de forma separada.
- **Número de Archivos:** Se cuenta cada archivo maestro lógico (esto es. Un grupo lógico de datos que puede ser una parte de una gran base de datos o un archivo independiente).
- **Número de Interfaces Externas:** Se cuentan todas las interfaces legibles por la maquina (por ejemplo: archivos de datos de cinta o disco) que se utilizan para transmitir información a otro sistema.

**Tabla Nº 2. 1 Calculo de Punto de Fusión no Ajustado**

Parámetros de Medición	Cuentas	Factor de Ponderación			
		Simple	Medio	Complejo	Total
Nº de Entradas de Usuario		3	4	6	
Nº de Salidas de Usuario		4	5	7	
Nº de Peticiones de Usuario		3	4	6	
Nº de Archivos en Operación		7	10	15	
Nº de Interface Externos		5	7	10	

Fuente: (R., Pressman, 2000)

El punto función se puede calcular mediante la siguiente ecuación:

$$PF = \text{Cuenta Total} * (X + Y * \sum F_i)$$

Dónde:

Cuenta Total: Es la suma de todas las entradas obtenidas en N° de Entradas, N° de Salidas, N° de Archivos y N° de Interfaces Externas.

X: Nivel de confiabilidad del sistema es de (0,65).

Y: Nivel de error igual a (0,01).

F<sub>i</sub> (i=1 a 14): Son los valores de ajustes de complejidad según las respuestas a las preguntas destacados en la siguiente tabla.

**Tabla N° 2. 2 Calculo de Punto de Fusión Ajustado**

N°	Factores de Complejidad	Valor
1	¿Requiere el sistema copias de seguridad y de recuperación fiable?	
2	¿Se requiere comunicación de datos?	
3	¿Existen funciones de procesamiento distribuido?	
4	¿Es crítico el rendimiento?	
5	¿Se ejecutan el sistema en un entorno operativo existente y fuertemente utilizado?	
6	¿Requiere el sistema entrada de datos interactiva?	
7	Facilidad Operativa.	
8	¿Se actualizan los archivos maestros de forma interactiva?	
9	¿Son complejas as entradas, las salidas, los archivos o las peticiones?	
10	¿Es complejo el procesamiento interno?	
11	¿Se ha diseñado el código para ser reutilizable?	
12	Facilidad de Instalar.	
13	¿Se ha diseñado el sistema para soportar múltiples instalaciones en diferente organización?	
14	Facilidad de cambio.	
Total		

**Fuente:** (R., Pressman, 2000)

Cada una de las preguntas anteriores es respondida usando una escala con rangos desde **0 (lo importante o aplicable)** hasta **5 (absolutamente esencial)**. Los valores constantes de la ecuación y los factores de peso que se aplican a las cuentas de los dominios de información se determina empíricamente.

$$\%Funcionalidad = \$PF = \frac{PF_{Calculada}}{PF_{Maxima}} * 100\%$$

## ✚ CONFIABILIDAD

La confiabilidad es la capacidad del software para asegurar un nivel de funcionamiento adecuado cuando es utilizado en condiciones específicas. En este caso a la confiabilidad se amplía sostener un nivel especificado de funcionamiento y no una función requerida.

**Figura N° 2. 13 Características de confiabilidad ISO/IEC 9126**



**Fuente:** (norma evaluacion iso 9126, 2013)

La confiabilidad se calculará siguiendo las siguientes formulas, donde se determinará la probabilidad de fallas y la probabilidad de no tener fallas en el sistema.

### **Característica de Confiabilidad**

La confiabilidad se divide en 4 criterios:

- ✓ **Madurez:** La capacidad que tiene el software para evitar fallas cuando encuentra errores. Ejemplo, la forma como el software advierte al usuario cuando realiza operaciones en la unidad de diskett vacía, o cuando no encuentra espacio suficiente el disco duro donde esta almacenando los datos.
- ✓ **Tolerancia a errores:** La capacidad que tiene el software para mantener un nivel de funcionamiento en caso de errores.
- ✓ **Recuperabilidad:** La capacidad que tiene el software para restablecer su funcionamiento adecuado y recuperar los datos afectados en el caso de una falla.

- ✓ **Conformidad de la fiabilidad:** La capacidad del software de cumplir a los estándares o normas relacionadas a la fiabilidad.

Por lo tanto, para poder medir la confiabilidad del sistema, se tomó la siguiente fórmula que calcula la confiabilidad del sistema.

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

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

Con:  $F(t) = F_c * (e^{-\lambda / 7 * 12})$

Dónde:

$F_c = 0.87$  Funcionalidad del sistema

$\lambda = 1$  tasa de fallos dentro de un mes

También utilizando la métrica de madurez de la siguiente tabla.

## ✚ USABILIDAD

La usabilidad es la capacidad del software de ser entendido, aprendido, y usado en forma fácil y atractiva. Algunos criterios de funcionalidad, fiabilidad y eficiencia afectan la usabilidad, pero para los propósitos de la ISO/IEC 9126 ellos no clasifican como usabilidad. La usabilidad está determinada por los usuarios finales y los usuarios indirectos del software, dirigidos a todos los ambientes, a la preparación del uso y el resultado obtenido.

**Figura N° 2. 14 Características de usabilidad ISO/IEC 9126**



**Fuente:** (norma evaluacion iso 9126, 2013)

## Característica de Usabilidad

La usabilidad se divide en 5 criterios.

- ✓ **Entendimiento:** La capacidad que tiene el software para permitir al usuario entender si es adecuado, y de una manera fácil como ser utilizado para las tareas y las condiciones particulares de la aplicación. En este criterio se debe tener en cuenta la documentación y de las ayudas que el software entrega.
- ✓ **Aprendizaje:** La forma como el software permite al usuario aprender su uso. También es importante considerar la documentación.
- ✓ **Operabilidad:** La manera como el software permite al usuario operarlo y controlarlo.
- ✓ **Atracción:** La presentación del software debe ser atractiva al usuario. Esto se refiere a las cualidades del software para hacer más agradable al usuario, ejemplo, el diseño gráfico.
- ✓ **Conformidad de uso:** La capacidad del software de cumplir los estándares o normas relacionadas a su usabilidad.

## + EFICIENCIA

La eficiencia del software es la forma del desempeño adecuado, de acuerdo a al número de recursos utilizados según las condiciones planteadas. Se debe tener en cuenta otros aspectos como la configuración de hardware, el sistema operativo, entre otros.

**Figura N° 2. 15 Características de eficiencia ISO/IEC 9126**



**Fuente :**(norma-de-evaluacion-isoiec-9126, 2013)

## Características de Eficiencia

La eficiencia se divide en 3 criterios:

- ✓ **Comportamiento de tiempos:** Los tiempos adecuados de respuesta y procesamiento, el rendimiento cuando realiza su función en condiciones específicas. Ejemplo, ejecutar el procedimiento más complejo del software y esperar su tiempo de respuesta, realizar la misma función, pero con más cantidad de registros.
- ✓ **Utilización de recursos:** La capacidad del software para utilizar cantidades y tipos adecuados de recursos cuando este funciona bajo requerimientos o condiciones establecidas. Ejemplo, los recursos humanos, el hardware, dispositivos externos.
- ✓ **Conformidad de eficiencia:** La capacidad que tiene el software para cumplir con los estándares o convenciones relacionados a la eficiencia.

## Capacidad de Mantenimiento

La capacidad de mantenimiento es la cualidad que tiene el software para ser modificado. Incluyendo correcciones o mejoras del software, a cambios en el entorno, y especificaciones de requerimientos funcionales.

**Figura N° 2. 16 Características de capacidad de mantenimiento ISO/IEC 9126**



**Fuente:** (norma-de-evaluacion-isoiec-9126, 2013).

Para determinar la mantenibilidad de un sistema la cantidad de módulos actuales, los módulos nuevos cambiado y los módulos actuales nuevo.

## Característica de Mantenimiento

El mantenimiento se divide en 5 criterios:

- ✓ **Capacidad de ser analizado:** La forma como el software permite diagnósticos de deficiencias o causas de fallas, o la identificación de partes modificadas.
- ✓ **Confiabilidad:** La capacidad del software para que la implementación de una modificación se pueda realizar, incluye también codificación, diseño y documentación de cambios.
- ✓ **Estabilidad:** La forma como el software evita efectos inesperados para modificaciones del mismo.
- ✓ **Facilidad de prueba:** La forma como el software permite realizar pruebas a las modificaciones sin poner el riesgo los datos.
- ✓ **Conformidad de facilidad de mantenimiento:** La capacidad que tiene el software para cumplir con los estándares de facilidad de mantenimiento.

Es la facilidad con que una modificación puede ser realizada en el sistema. Las modificaciones puedes incluir correcciones, mejoras o adaptar si su entorno cambia, o mejorar si el cliente desea un cambio de requisitos. Para este fin, Pressman nos sugiere el Índice de Madurez de Software (IMS) para determinar la estabilidad de un producto software. Dicha IMS es calculada por la siguiente ecuación:

$$IMS = \frac{[Mt - (Fa + Fc + Fd)]}{Mt}$$

Dónde:

Mt: Número de módulos de la versión actual.

Fc: Número de módulos en la versión actual que se han cambiado.

Fa: Número de módulos en la versión actual que se han añadido.

Fd: Número de módulos de la versión anterior que se han borrado en la versión actual.

## Portabilidad

La capacidad que tiene el software para ser trasladado de un entorno a otro.

La portabilidad se define como la característica que posee un software para ejecutar en diferentes plataformas, el código fuente del software es capaz de reutilizarse en vez de crearse un nuevo código cuando el software pasa de una plataforma a otra. A mayor portabilidad es menor la dependencia del software con respecto a la plataforma.

La ISO 9126 se basa en que el objetivo no es necesariamente alcanzar una calidad perfecta, sino la necesaria y suficiente para cada contexto de uso a la hora de la entrega y del uso del software por parte de los usuarios y es necesario comprender las necesidades reales de los usuarios con tanto detalle como sea posible (requisitos).

**Figura N° 2. 17 Características de probabilidad ISO/IEC 9126**



**Fuente:** (norma-de-evaluacion-isoiec-9126, 2013).

### **Características de Portabilidad**

La usabilidad se divide en 5 criterios:

- ✓ **Adaptabilidad:** Es como el software se adapta a diferentes entornos especificados (hardware o sistemas operativos) sin que implique reacciones negativas ante el cambio. Incluye la escalabilidad de capacidad interna (Ejemplo: Campos en pantalla, tablas, volúmenes de transacciones, formatos de reporte, etc.).
- ✓ **Facilidad de instalación:** La facilidad del software para ser instalado en un entorno específico o por el usuario final.

- ✓ **Coexistencia:** La capacidad que tiene el software para coexistir con otro o varios softwares, la forma de compartir recursos comunes con otro software o dispositivo.
- ✓ **Reemplazabilidad:** La capacidad que tiene el software para ser reemplazado por otro software del mismo tipo, y para el mismo objetivo. Ejemplo, la Reemplazabilidad de una nueva versión es importante para el usuario, la propiedad de poder migrar los datos a otro software de diferente proveedor.
- ✓ **Conformidad de portabilidad:** La capacidad que tiene el software para cumplir con los estándares relacionados a la portabilidad.

### Calidad en Uso

Calidad en uso es la calidad del software que el usuario final refleja, la forma como el usuario final logra realizar los procesos con satisfacción, eficiencia y exactitud.

La calidad en uso debe asegurar la prueba o revisión de todas las opciones que el usuario trabaja diariamente y los procesos que realiza esporádicamente relacionados con el mismo software.

**Figura N° 2. 18 Calidad de Uso de Software ISO/IEC 9126**



**Fuente:** (norma-de-evaluacion-isoiec-9126, 2013).

### Características Calidad de Uso

La calidad de uso se divide en 4 criterios:

- ✓ **Eficacia:** La capacidad del software para permitir a los usuarios finales realizar los procesos con exactitud e integridad.
- ✓ **Productividad:** La forma como el software permite a los usuarios emplear cantidades apropiadas de recursos, en relación a la eficacia lograda en un contexto específico de uso. Para una empresa es muy importante que el software no afecte a la productividad del empleado
- ✓ **Seguridad:** Se refiere a que el Software no tenga niveles de riesgo para causar daño a las personas, instituciones, software, propiedad intelectual o entorno. Los riesgos son normalmente el resultado de deficiencias en la funcionalidad (Incluyendo seguridad), fiabilidad, usabilidad o facilidad de mantenimiento.
- ✓ **Satisfacción:** La satisfacción es la respuesta del usuario a la interacción con el software, e incluye las actitudes hacia el uso del mismo.

## **2.14 SEGURIDAD INFORMATICA**

### **2.14.1 Estándar ISO/IEC 27000**

Estándar para la seguridad de información, aprobada y publicada en octubre de 2005 por ISO. Especifica los requisitos necesarios para establecer, mantener y mejorar un sistema de Gestión de la seguridad de la información (SGSI).SGSI también llamada Sistema de Gestión de la Seguridad de la Información, conjunto de políticas y procedimiento que normaliza la gestión de la seguridad de la información de toda una organización, debe garantizar la disponibilidad, confidencialidad e integridad de la información.

### **2.14.2 ISO 27002**

Es una guía de buenas prácticas que permite a las organizaciones mejorar la seguridad de su información mediante control y gestión. (*Maidana, Seguridad Informatica, 2014*)

**Figura N° 2. 19 Estructura ISO 27002**



**Fuente:** (Maidana, Seguridad Informática, 2014)

## 2.15 METODOS DE ESTIMACION DE COSTOS DE SOFTWARE

La estimación de los costos de desarrollo de software es un factor muy importante en el análisis de los proyectos informáticos, constituye un tema estratégico contar con indicadores para medir el costo de los mismos, garantizando la eficiencia, excelencia, calidad y la competitividad. El análisis de costo es el proceso de identificación de los recursos necesarios para llevar a cabo el trabajo o proyecto eficientemente.

La evaluación del costo determina la calidad y cantidad de los recursos necesarios en términos de dinero, esfuerzo, capacidad, conocimientos y tiempo incidiendo en la gestión empresarial.

En la actualidad existen un conjunto de métricas que no se utilizan, y que pueden ser aplicables a cualquier tipo de proyecto de software para calcular el costo de los mismos.

Esta investigación propone el diseño de un conjunto de métricas para calcular el costo en el proceso de desarrollo de software.

Hoy en día existen diversas herramientas y metodologías que nos permiten estimar costos y existen muchos factores que afectan las estimaciones de costo como:

- ✓ Incertidumbre en los requerimientos.

- ✓ Términos contractuales rígidos.
- ✓ Salud financiera (ganar licitaciones sacrificando costo y tiempo).
- ✓ Falta de experiencia con “X” tecnología.

Disponible en: (Estimacion de costos de desarrollo de software , 2014).

### **2.15.1 Modelo COCOMO**

Modelo de estimación que se encuentra en la jerarquía de modelos de estimación de software con el nombre de COCOMO, por Constructive Cost Model (Modelo Constructivo de Coste). El modelo COCOMO original se ha convertido en uno de los modelos de estimación de coste del software más utilizados y estudiados en la industria.

#### **Características**

- Es una herramienta basada en las líneas de código la cual la hace muy poderosa para la estimación de costos y no como otros que solamente miden el esfuerzo en base al tamaño.
- Representa el más extenso modelo empírico para la estimación de software.
- Existen herramientas automáticas que estiman costos basados en COCOMO como ser: Costar, COCOMO 8

#### **Objetivos para la construcción de COCOMO**

- Desarrollar un modelo de estimación de costo y cronograma de proyectos de software que se adaptará tanto a las prácticas de desarrollo en tiempos actuales como a las futuras.
- Construir una base de datos de proyectos de software que permitiera la calibración continua del modelo, así incrementar la precisión en la estimación.
- Implementar una herramienta de software que soportara el modelo.
- Proveer un marco analítico cuantitativo y un conjunto de herramientas y técnicas que evaluaran el impacto de las mejoras tecnológicas de software

sobre los costos y tiempos en las diferentes etapas del ciclo de vida de desarrollo.

Los tres modelos de COCOMO, se adaptan tanto a las necesidades de los diferentes sectores, como al tipo y cantidad de información disponible en cada etapa del ciclo de vida de desarrollo, lo que se conoce por granularidad de la información. Estos tres modelos son:

- 1 Modelo de composición de aplicación.** Utilizado durante las primeras etapas de la Ingeniería del software, donde el prototipado de las interfaces de usuario, la interacción del sistema y del software, la evaluación del rendimiento, y la evaluación de la madurez de la tecnología son de suma importancia.
- 2 Modelo de fase de diseño previo.** Utilizado una vez que se han estabilizado los requisitos y que se ha establecido la arquitectura básica del software.
- 3 Modelo de fase posterior a la arquitectura.** Este modelo se aplica durante la construcción o desarrollo del software. Es el modelo de estimación más detallado y se aplica cuando la arquitectura del proyecto está completamente definida.

*Disponible en: (Modelo de estimacion de proyectos de software, s.f.)*

### **2.15.2 Modelo de Estimación**

En la estimación del tamaño de software COCOMO utiliza tres técnicas:

- Puntos Objeto,
- Puntos Función No Ajustados y
- Líneas de Código Fuente.

Además, se emplean otros parámetros relativos al tamaño que contemplan aspectos tales como: reusó, reingeniería, conversión y mantenimiento.

**Puntos objeto:** El procedimiento para determinar Puntos Objeto en un proyecto software se resume en:

1. Determinar Cantidad de Objetos: Estimar la cantidad de pantallas, reportes, componentes que contendrá la aplicación.

2. Clasificar cada instancia de un objeto según sus niveles de complejidad (simple, media o difícil)
3. Dar el peso a cada objeto según el nivel de complejidad. Los pesos reflejan el esfuerzo relativo requerido para implementar una instancia de ese nivel de complejidad.
4. Determinar la cantidad de Puntos Objeto, sumando todos los pesos de las instancias de los tipos de objetos especificados.

**Figura N° 2. 20 Puntos de objetivo**

Para Pantallas			
Cantidad de Vistas Contenidas	Cantidad y fuente de las tablas de datos		
	Total < 4 ( < 2 servidor < 3 cliente)	Total < 8 ( < 2 - 3 servidor < 3 - 5 cliente)	Total 8 + ( > 3 servidor < 5 cliente)
< 3	Simple	Simple	Media
3 - 7	Simple	Media	Difícil
> 8	Media	Difícil	Difícil
Para Reportes			
Cantidad de Vistas Contenidas	Cantidad y fuente de las tablas de datos		
	Total < 4 ( < 2 servidor < 3 cliente)	Total < 8 ( < 2 - 3 servidor < 3-5 cliente)	Total 8 + ( > 3 servidor < 5 cliente)
0 o 1	Simple	Simple	Media
2 o 3	Simple	Media	Difícil
4 +	Media	Difícil	Difícil

**Fuente disponible** en (Modeloe de estimacion, 2012)

### **Puntos Función:**

El modelo COCOMO usa Puntos Función y/o Líneas de Código Fuente (SLOC) como base para medir tamaño en los modelos de estimación de Diseño Temprano y Post-Arquitectura.

Los puntos función están basados en información disponible en las etapas tempranas del ciclo de vida del desarrollo de software.

COCOMO considera solamente UFP (Puntos Función no ajustados).

$$FP = UFP \times TCF$$

Donde UFP: Puntos Función no Ajustados

TCF: Factor de Complejidad Técnica

Para calcular los UFP, se deben identificar los siguientes elementos:

- ✓ **Entradas Externas (Inputs):** Entrada de datos del usuario o de control que ingresan desde el exterior del sistema para agregar y/o cambiar datos a un archivo lógico interno.
- ✓ **Salidas Externas (Outputs):** Salida de datos de usuario o de control que deja el límite del sistema de software.
- ✓ **Archivo Lógicos Internos (Archivos):** Incluye cada archivo lógico, es decir cada grupo lógico de datos que es generado, usado, o mantenido por el sistema de software.
- ✓ **Archivos Externos de Interface (Interfaces):** Archivos transferidos o compartidos entre sistemas de software.
- ✓ **Solicitudes Externas (Queries):** Combinación única de entrada- salida, donde una entrada causa y genera una salida inmediata, como un tipo de solicitud externa.

Una vez identificados los elementos se clasifican de acuerdo al grado de complejidad en: bajo, promedio o alto. Se asigna un peso a cada ítem según el tipo y el grado de complejidad correspondiente. Finalmente, los UFP son calculados sumando los pesos de todos los ítems identificados.

### **Líneas de Código Fuente (SLOC):**

El objetivo es medir la cantidad de trabajo intelectual puesto en el desarrollo de un programa. Definir una línea de código es difícil debido a que existen diferencias conceptuales cuando se cuentan sentencias ejecutables y de declaraciones de datos en lenguajes diferentes.

A los efectos de COCOMO, se eliminan las categorías de software que consumen poco esfuerzo. Así no están incluidas librerías de soporte, sistemas operativos, librerías comerciales, etc., ni tampoco el código generado con generadores de código fuente.

## Conversión de Puntos Función a Líneas de Código Fuente (SLOC):

Para determinar el esfuerzo nominal en el modelo COCOMO, puntos función no ajustados tienen que ser convertidos a líneas de código fuente considerando el lenguaje de implementación.

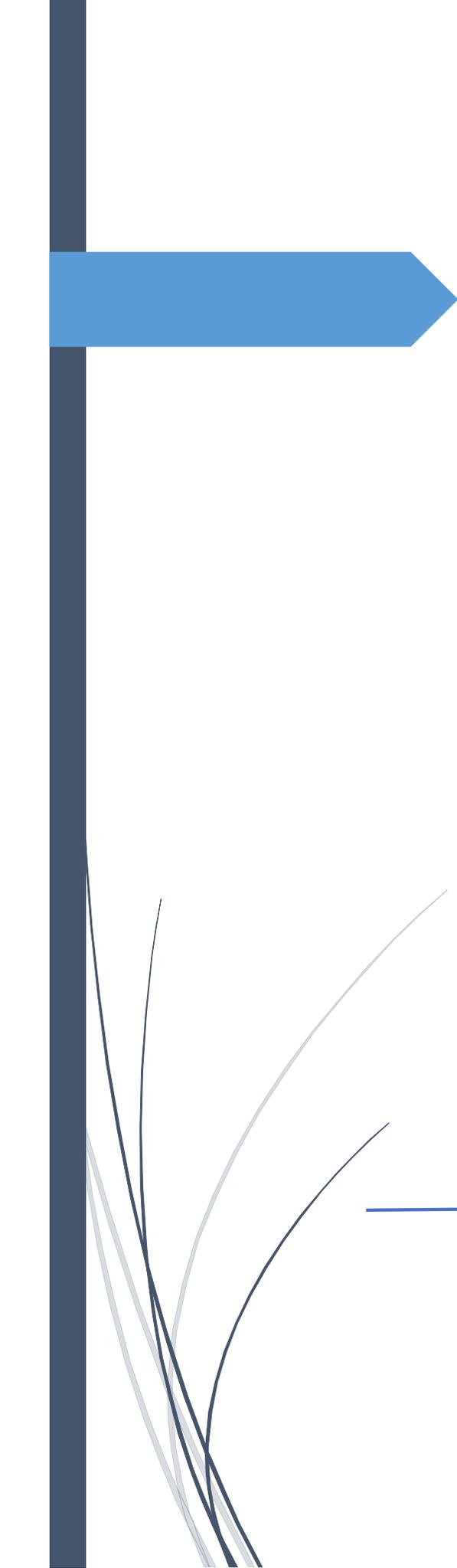
Por otro lado, el modelo COCOMO (CONstructive COst MOdel) Calcula esfuerzo y coste en función del tamaño del programa (LDC). COCOMO está definido para tres tipos de proyectos de Software:

1. Modo orgánico: Proyectos pequeños y sencillos, con equipos de experiencia en la aplicación y requisitos poco rígidos.
2. Modo semi-acoplado: Proyectos intermedios (más complejos), con equipos que poseen variados niveles de experiencia y requisitos más rígidos.
3. Modo empotrado: Proyectos que deben ser desarrollados en un conjunto de Hardware, Software y restricciones muy grandes.
4. Disponible en: (Estimaciones,2007).

**Figura N° 2. 21 Detalles de coeficiente de COCOMO**

PARA PUNTOS DE FUNCIÓN:			COCOMO BASICO				
DOMINIO	COMPLEJIDAD	PESO	Modo (Tipo de Proyecto)	a	b	c	d
Salidas	Alta	7	Orgánico	2.4	1.05	2.5	0.38
	Media	5					
	Baja	4					
Entradas	Alta	6	Semiacoplado	3.0	1.12	2.5	0.35
	Media	4					
	Baja	3					
Consultas	Alta	7	Empotrado	3.6	1.20	2.5	0.32
	Media	5					
	Baja	4					
Archivo Interno	Alta	15	$E = a * (KLDC)^b$ $D = c * (E)^d$				
	Media	10					
	Baja	7					
Archivo Externo / Interfaces	Alta	10	COCOMO INTERMEDIO				
	Media	7	Modo (Tipo de Proyecto)	a	b		
	Baja	5	Orgánico	3.2	1.05		
			Semiacoplado	3.0	1.12		
			Empotrado	2.8	1.20		
			$E = a * (KLDC)^b * FAE$ $D = 2.5 * (E)^{0.38}$				

Fuente: (IS Estimaciones(pdf)).



# CAPITULO III

---

## MARCO APLICATIVO

*INGENIERIA DE SISTEMAS - UNIVERSIDAD PÚBLICA EL ALTO*

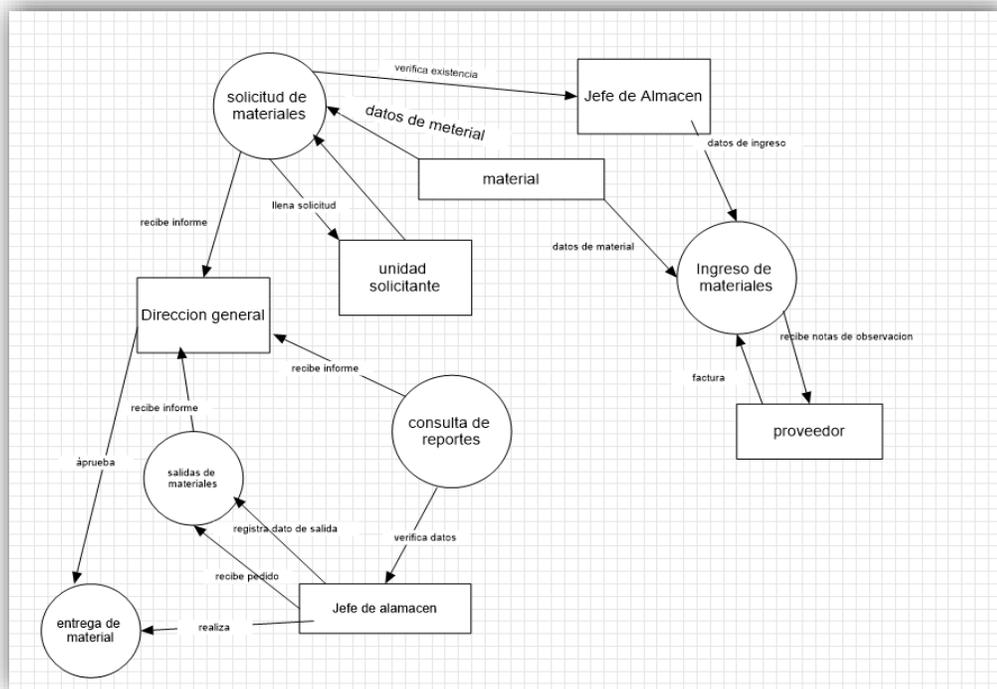
### 3. MARCO APLICATIVO

En este capítulo se llevará a cabo el desarrollo del sistema de información de control de inventarios en almacén para la Normal de El Alto E.S.T.H.E.A., se aplicará las metodologías, normas y técnicas mencionadas en el marco teórico del capítulo II.

#### 3.1. ANÁLISIS DEL SISTEMA ACTUAL

El área de Almacén de inventarios de la institución E.S.F.M.T.H.A., entre las actividades que realiza son controlar las adquisiciones de materiales y clasificar los movimientos, procesos que se realiza de forma manual, actividad que representa un esfuerzo adicional para los encargados de esta área, retrasando la entrega de materiales, pero si contaran con un sistema de información esta actividad se realizaría en menor tiempo agilizando todo los procesos y actividades de la Institución.

**Figura N° 3. 1 Diagrama de Flujo de Datos**



**Fuente:** Elaboración propia

### 3.2 OBTENCIÓN DE REQUISITOS

La ingeniería de requisitos es fundamental para la realización de un sistema sea exitoso, en este caso para el presente proyecto se detallará el siguiente recuadro de actividades.

**Tabla N° 3. 1 Obtención de Requisitos**

TAREAS	DESCRIPCION
<b>Entrevistas</b>	Se realizó la entrevista a los funcionarios de la institución: <ul style="list-style-type: none"><li>❖ Jefa de Almacén, se encarga de suministrar, supervisar el almacén</li><li>❖ Jefes de Áreas, cada jefe de área solicitan materiales según la necesidad del área.</li></ul>
<b>Observaciones</b>	Los problemas que se observaron en la institución en área de almacenes es falta de control de stock de materiales y coordinación de solicitudes de requerimientos.
<b>Documentación</b>	Se obtuvo a la siguiente documentación de la institución: reportes de solicitud de pedido y entrega de materiales y así mismo se dio acceso a información de registro en stock de materiales.

#### 3.2.1 Identificación de los Actores

La identificación de actores permite conocer a las personas involucradas en el proceso de control de almacén y solicitud de materiales de la institución.

**Figura N° 3. 2 Escuela de Formación de Maestro El Alto E.S.F.M.T.H.A**



**Fuente:** Elaboración Propia

**Tabla N° 3. 2 Descripción de Actores**

ACTOR	DESCRIPCION
❖ <b>Jefe de Almacén</b>	Encargado de realizar las siguientes funciones: <ul style="list-style-type: none"> <li>• Verifica el material, el stock y registra el nuevo ingreso de materiales y realiza los reportes generales.</li> <li>• Realiza el registro de proveedores</li> <li>• Controla y crea los nuevos usuarios para cada funcionario autorizados</li> </ul>
❖ <b>Funcionarios</b>	Encargado de realizar las siguientes funciones: <ul style="list-style-type: none"> <li>• Realizan la solicitud de materiales a almacén</li> </ul>

**Fuente:** Elaboración Propia

### 3.3 INGENIERÍA DE REQUERIMIENTOS

La ingeniería de requerimientos cumple un papel primordial en el proceso de construcción y producción de un software, es decir que estará basada en función de las necesidades planteadas por los clientes en un nivel muy general, donde se describe, documenta analiza y se define los servicios o componentes de lo que se desea producir, además de las restricciones que tendrá el producto a seguir en la construcción de un software y de facilitar la comprensión de lo que el cliente requiera. (Arias, 2006).

Trata de diferente forma las necesidades de información, las necesidades de navegación, las necesidades de adaptación y la 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. (Arias, 2006)

Las funciones que se deben realizar se clasifican en tres categorías que son las siguientes:

**Tabla Nº 3. 3 Categoría de Funciones**

CATEGORIA DE LA FUNCION	SIGNIFICADO
<b>Evidente</b>	Debe realizarse, los usuarios estén conscientes que se ha realizado
<b>Oculto</b>	Debe realizarse, aunque no sea visible para los usuarios. Esto aplica a muchos servicios técnicos, como guardar información es un mecanismo persistente de almacenamiento, las funciones ocultas muchas veces se omiten durante el proceso de obtención de los requerimientos.
<b>Superflua</b>	Opcionales: su inclusión no repercute significativamente en el costo ni en las otras funciones.

**Fuente:** (Arias, 2006)

### 3.3.1 Requerimientos Funcionales

Un requisito funcional define la función del sistema de software sus componentes, esta descrita como un conjunto de entradas comportamientos y salidas.

Los requisitos funcionales de un software se suelen registrar en la matriz de trazabilidad de requerimientos y en la especificación de requerimientos de software, este último, documenta las operaciones y actividades que el sistema debe poder desempeñar.

Los requisitos funcionales definen una función del sistema de software o sus componentes, una función es descrita como un conjunto de entradas y salidas.

Los requerimientos funcionales pueden ser cálculos, detalles técnicos, manipulación de datos y otras funcionalidades específicas que se supone un sistema debe cumplir. (Arias, 2006)

**Tabla Nº 3. 4 Requerimientos Funcionales**

REF.	FUNCION	CATEGORIA
R.1.1	El sistema debe tener seguridad en el acceso a la información	Evidente
R.1.2	Acceder al sistema por tipos de usuarios (jefe de almacén, funcionarios)	Evidente
R.1.3	Registro de áreas	Evidente
R.1.4	Registro de funcionarios	Evidente
R.1.5	Registro de proveedores	Evidente
R.1.6	Registro de entrada y salida de materiales	Evidente
R.1.7	Generar Reportes	Evidente
R.1.8	Cerrar sesión	Oculto

**Fuente:** Elaboración Propia

### 3.3.2 Requerimientos No Funcionales

**Tabla N° 3. 5 Obtención de Requisitos no Funcionales**

REF	FUNCION	CATEGORIAS
R.1.1	El sistema debe visualizarse y funcionar en cualquier tipo de navegador como ser Chrome, Mozilla, Explore, etc.	Evidente
R.1.2	Soporte y mantenimiento periódico para asegurar el buen rendimiento del sistema	Evidente
R.1.3	El sistema debe tener un entorno amigable al usuario, para poder tener un entendimiento de los funcionalidades y operaciones que realizara el sistema.	Evidente
R.1.4	Mantenimiento adecuado a la red	Evidente

**Fuente:** Elaboración Propia

## 3.4 APLICACION DEL MODELO UWE

### 3.4.1 Modelo De Casos De Uso

En este punto se realiza el análisis de requerimientos del sistema mediante el diseño de casos de uso, que describe el comportamiento del sistema y además elementos que permiten la abstracción de problema.

#### 3.4.1.1 Identificación de los Actores

La identificación de los actores permite conocer a las personas involucradas a continuación se describe a los actores identificados.

- Jefe de Almacén.
- Funcionarios

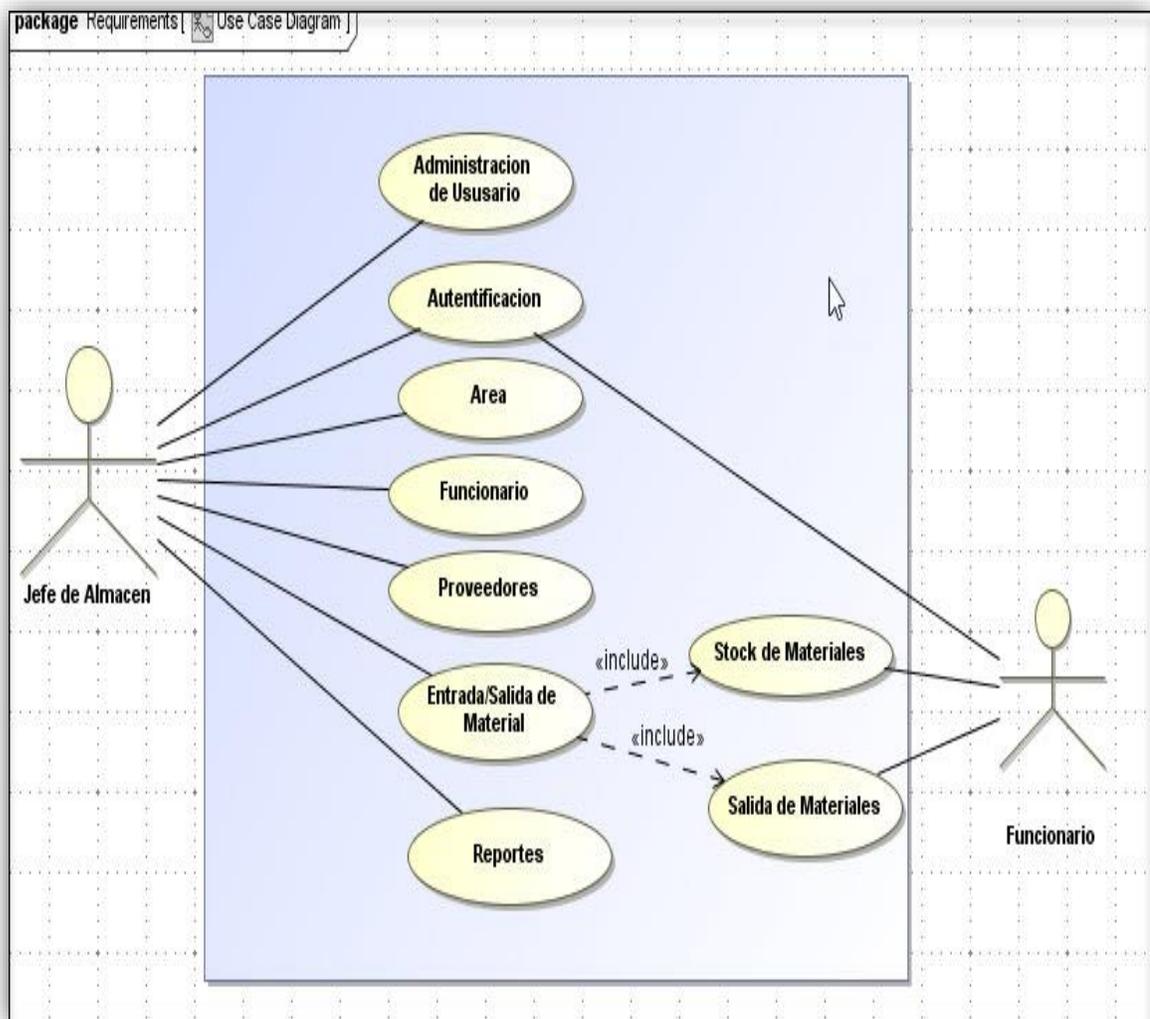
### 3.4.2 Diagrama de casos de uso

Los casos de uso, ayudan a capturar los requisitos adecuados para el diseño y análisis, ayudan a describir que es lo que el sistema debe hacer.

A continuación, se plasma el análisis de requerimientos del sistema mediante el diseño de casos de uso expresados en el comportamiento del sistema frente a acciones de los actores.

#### 3.4.2.1. Diagrama de casos de uso General del Sistema

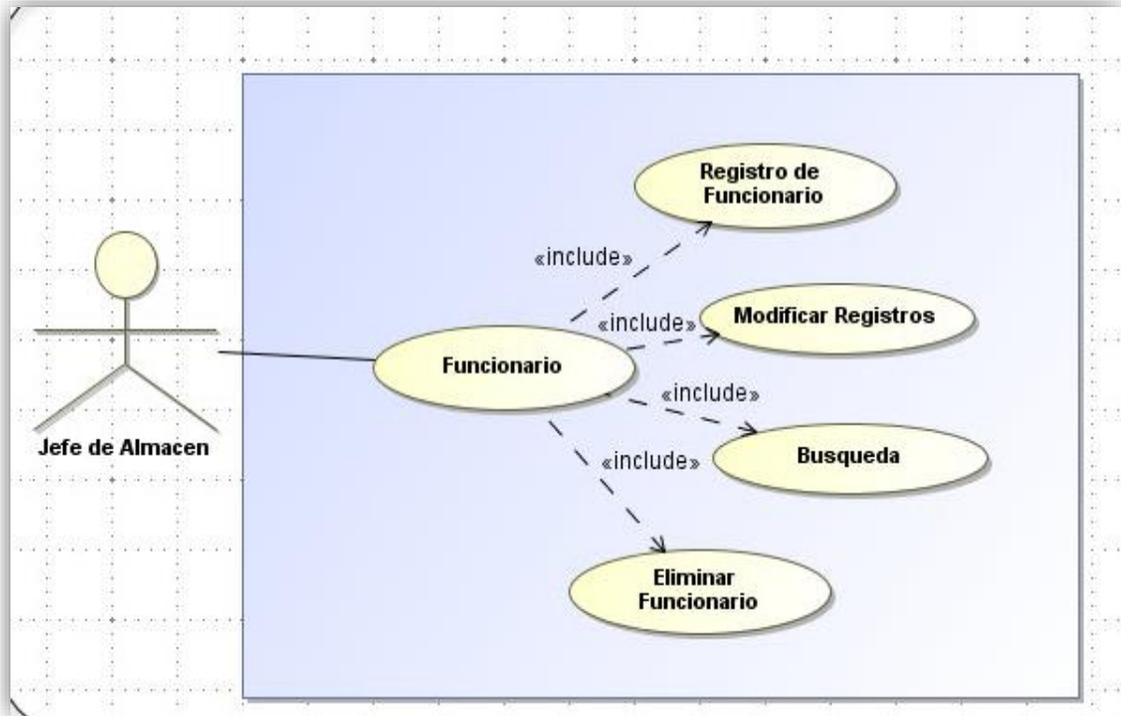
**Figura N° 3. 3 Diagrama de Caso de Uso: GENERAL DEL SISTEMA**



**Fuente:** Elaboración Propia

**3.4.2.2. Diagrama de Caso de Uso: Administración de Usuarios: Funcionarios**

**Figura N° 3. 4 Diagrama de Caso de uso: Funcionarios**



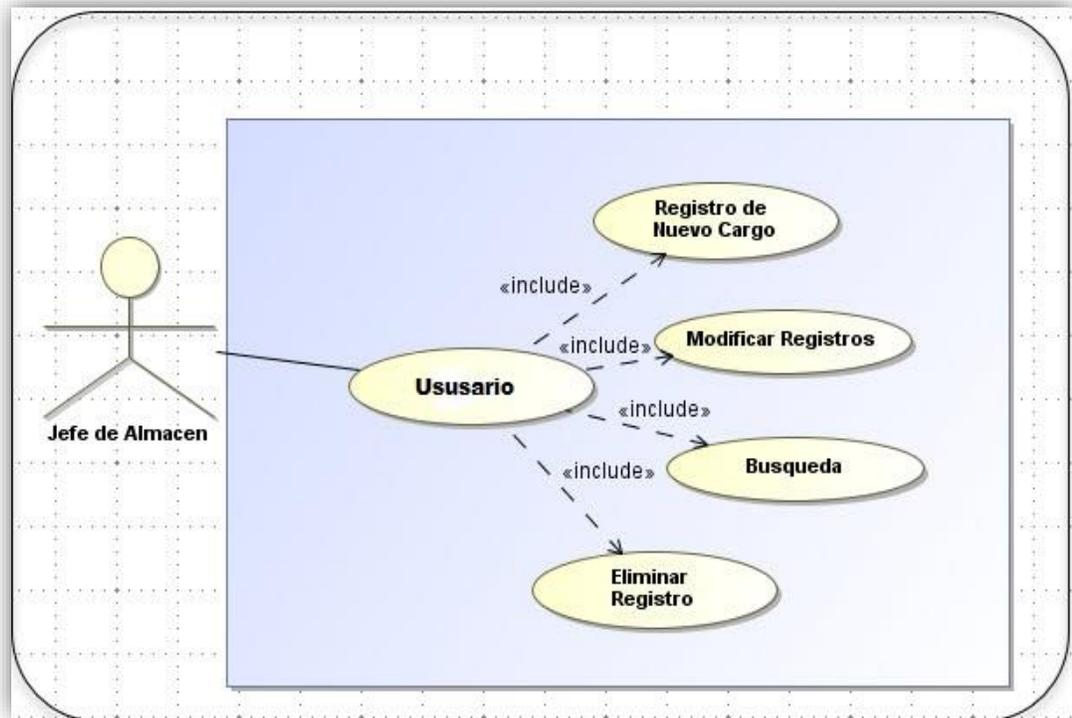
**Fuente:** Elaboración Propia

*Tabla N° 3. 6 Especificación de Caso de Uso: Administración de Usuarios- Funcionarios.*

Caso de Uso :Administración de Usuarios -Funcionarios	
<b>Actor</b>	Jefe de Almacén
<b>Tipo</b>	Primario Esencial
<b>Descripción</b>	El jefe de almacén inicia sesión como administrador, este usuario puede realizar las siguientes acciones: crear nuevo usuario, modificar.

**Fuente:** Elaboración Propia

**Figura Nº 3. 5 Diagrama de Caso de Uso: Usuario**



**Fuente:** Elaboración Propia

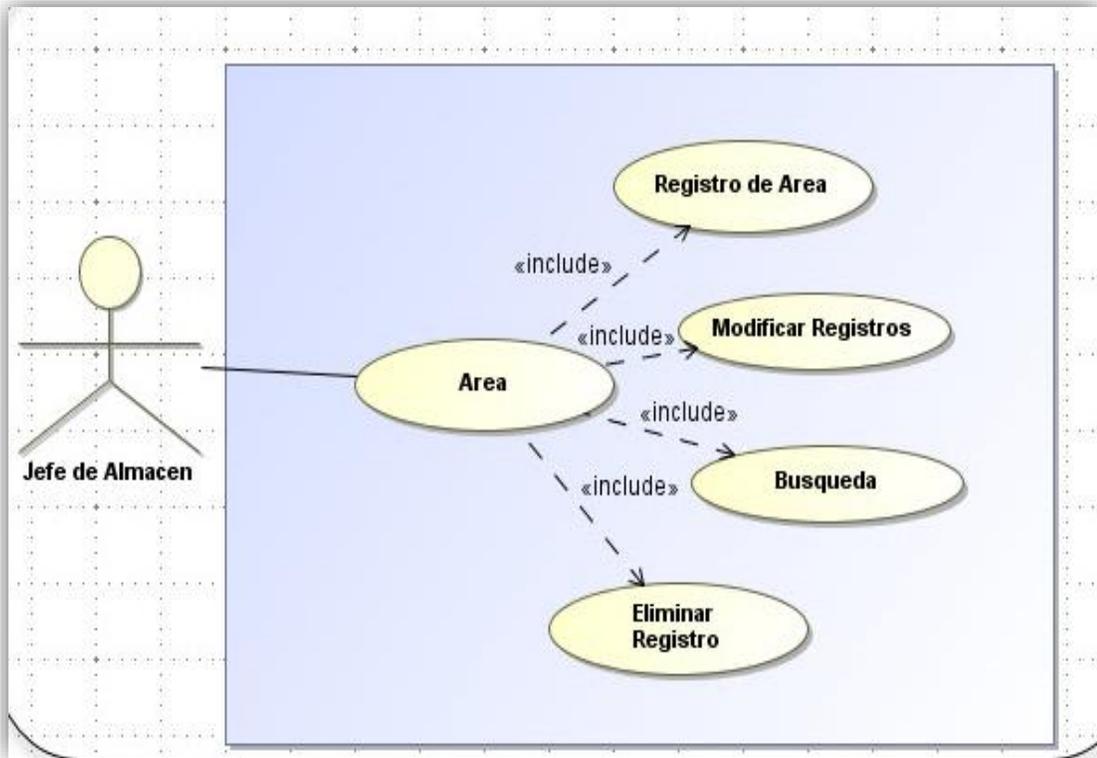
**Tabla Nº 3. 7 Especificación de Caso de Uso: Usuario**

Caso de Uso :Administración de Usuarios - Usuario	
<b>Actor</b>	Jefe de Almacén
<b>Tipo</b>	Primario Esencial
<b>Descripción</b>	El jefe de almacén ,ingresa al módulo de Usuario y puede realizar las siguientes acciones :crear nuevo área, modificar, eliminar.

**Fuente:** Elaboración Propia

### 3.4.2.3. Diagrama de Caso de Uso: Administración de Usuarios: Área

Figura N° 3. 6 Diagrama de Caso de Uso: Área



Fuente: (Elaboración Propia)

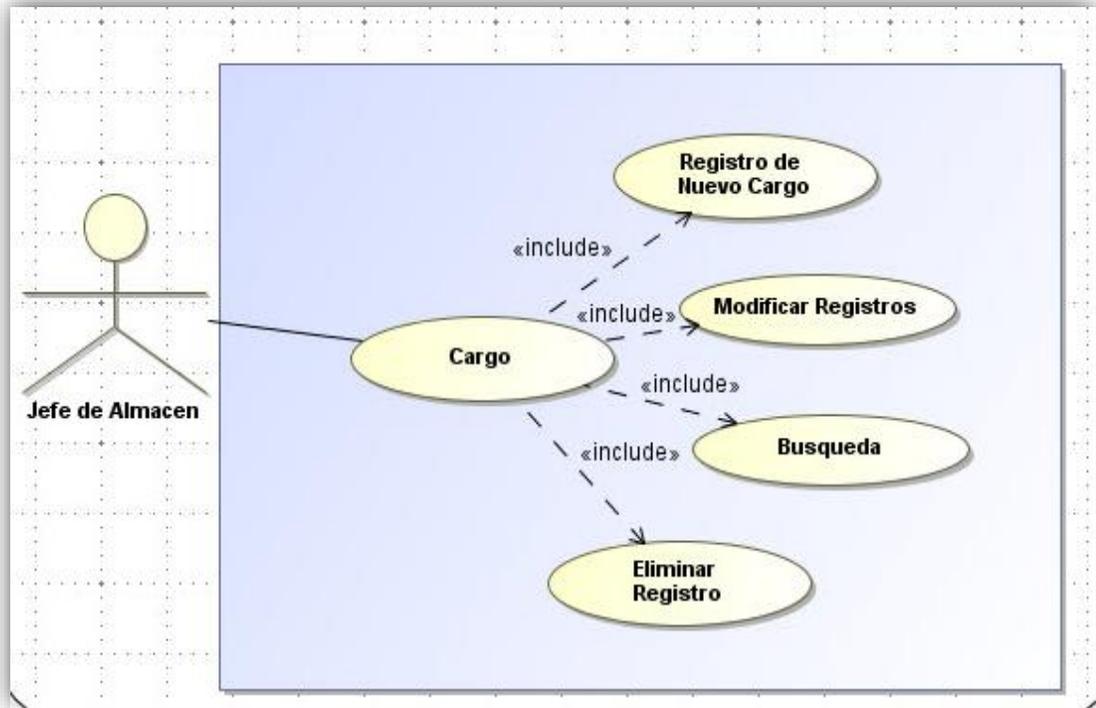
Tabla N° 3. 8 Especificación de Caso de Uso: Área

Caso de Uso :Administración de Usuarios - Área	
<b>Actor</b>	Jefe de Almacén
<b>Tipo</b>	Primario Esencial
<b>Descripción</b>	El jefe de almacén, ingresa al módulo de administración de usuarios y al sub módulo área y puede realizar las siguientes acciones: crear nueva área, modificar, eliminar.

Fuente: (Elaboración Propia)

### 3.4.2.4 Diagrama de Caso de Uso: Administración de Usuarios: Cargo

Figura Nº 3. 7 Diagrama de Caso de Uso: Cargo



Fuente: (Elaboración Propia)

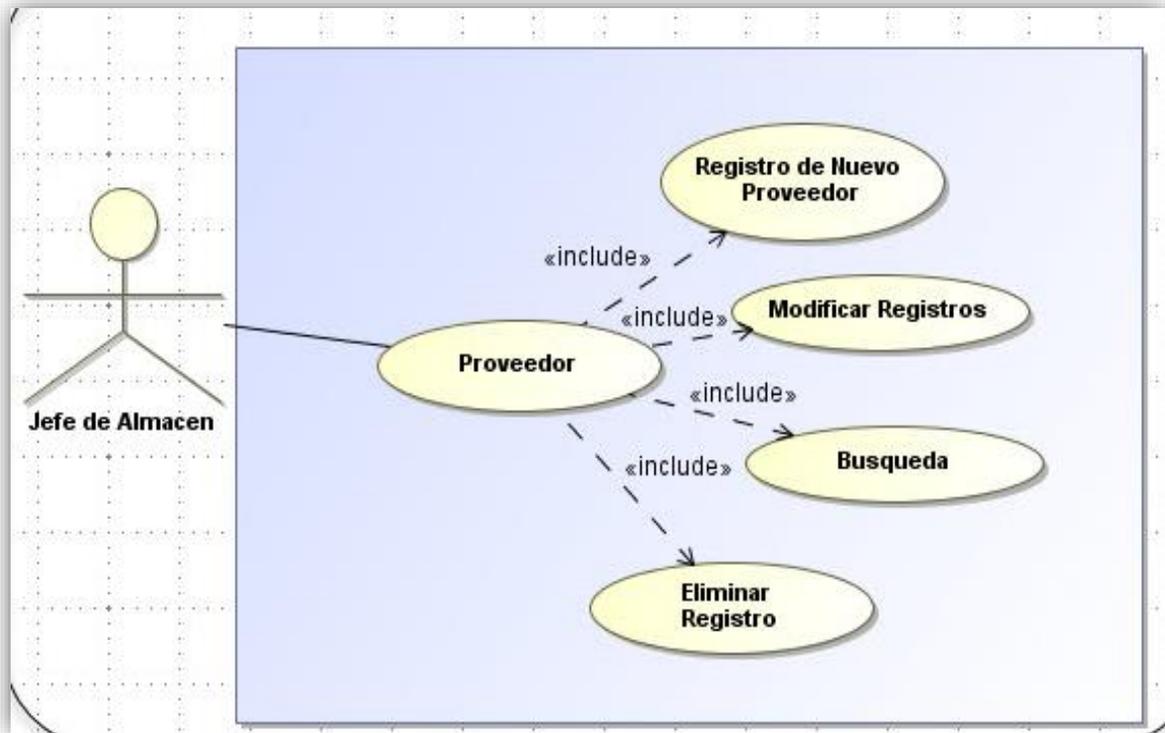
Tabla Nº 3. 9 Especificación de Caso de Uso: Cargo

Caso de Uso :Administración de Usuarios - Cargo	
<b>Actor</b>	Jefe de Almacén
<b>Tipo</b>	Primario Esencial
<b>Descripción</b>	El jefe de almacén, ingresa al módulo de administración de usuarios y al sub módulo cargo y puede realizar las siguientes acciones: crear nuevo área, modificar, eliminar.

Fuente: (Elaboración Propia)

### 3.2.4.5 Diagrama de Caso de Uso: Proveedores

Figura Nº 3. 8 Diagrama de Caso de Uso: Proveedores



Fuente: (Elaboración Propia)

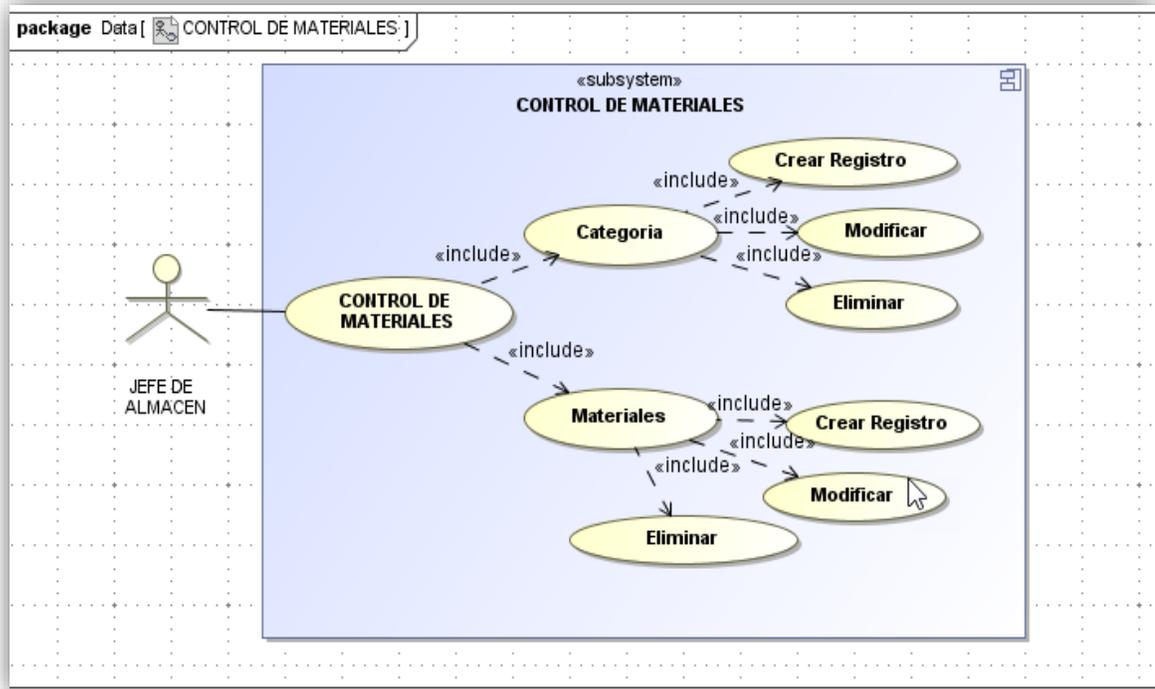
Tabla Nº 3. 10 Especificación de Caso de Uso: Proveedores

Caso de Uso :Administración de Usuarios - Proveedores	
<b>Actor</b>	Jefe de Almacén
<b>Tipo</b>	Primario Esencial
<b>Descripción</b>	El jefe de almacén, ingresa al módulo de Proveedores y puede realizar las siguientes acciones: crear nuevo área, modificar, eliminar.

Fuente: (Elaboración Propia)

### 3.2.4.6. Diagrama de Caso de Uso: Control de Materiales

Figura N° 3. 9 Diagrama de Caso de Uso: Control de Materiales



Fuente: (Elaboración Propia)

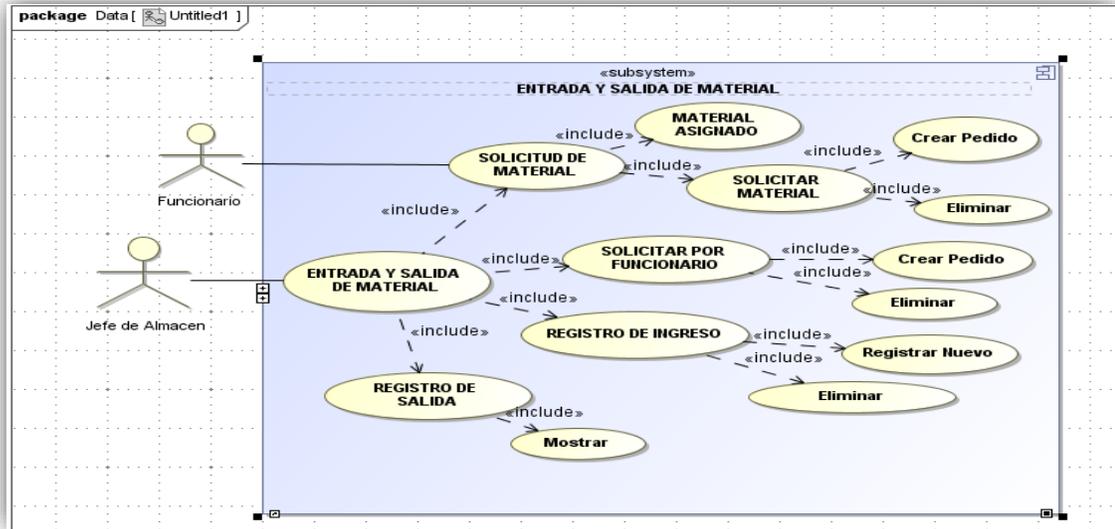
Tabla N° 3. 11 Especificación de Caso de Uso: Control de Materiales

Caso de Uso :Administración de Usuarios – Control de Materiales	
<b>Actor</b>	Jefe de Almacén
<b>Tipo</b>	Primario Esencial
<b>Descripción</b>	<p>El jefe de almacén, ingresa al módulo de Control de Materiales tiene los siguientes submenús:</p> <ul style="list-style-type: none"> <li>❖ Menú Categoría: tiene las siguientes acciones, crear registro, modificar, eliminar.</li> <li>❖ Menú Materiales: tiene las siguientes acciones, crear registro, modificar, eliminar.</li> </ul>

Fuente: (Elaboración Propia)

### 3.2.4.7. Diagrama de Caso de Uso: Entrada y Salida de Materiales

Figura N° 3. 10 Diagrama de Caso de Uso: Entrada y Salidas de Materiales



Fuente: (Elaboración Propia)

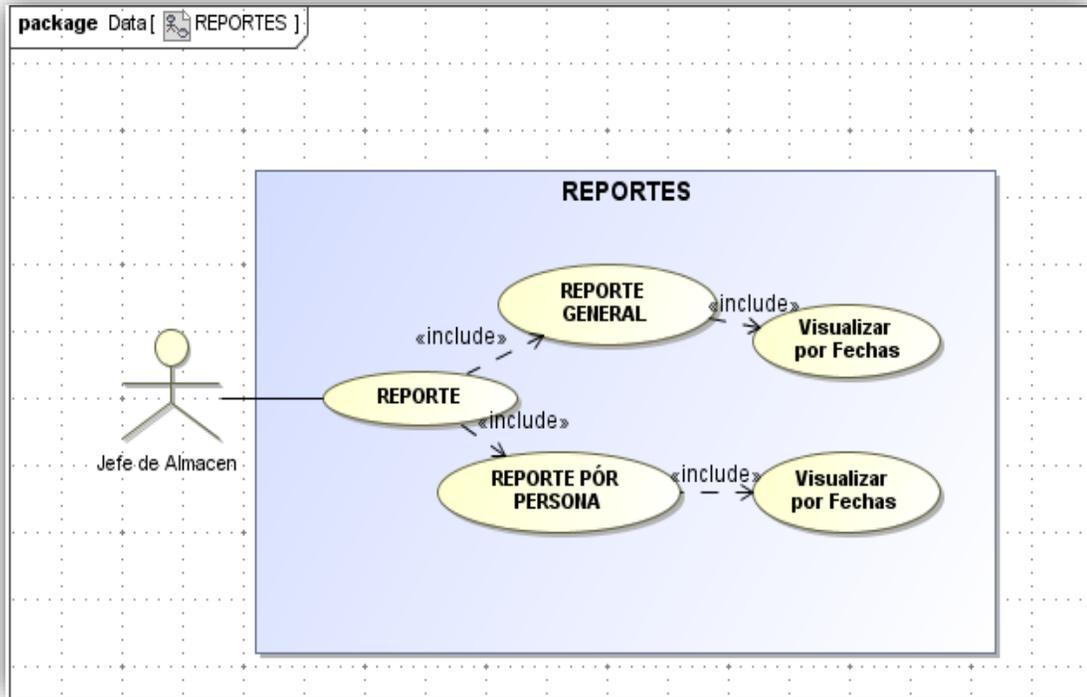
Tabla N° 3. 12 Especificación de Caso de Uso: Entradas y Salidas de Materiales

Caso de Uso :Administración de Usuarios – Entradas y Salidas de Materiales	
<b>Actor</b>	Jefe de Almacén y Funcionario
<b>Tipo</b>	Primario Esencial
<b>Descripción</b>	<p>El <b>jefe de almacén</b>, ingresa a inicio de sesión como administrador, realiza las siguientes acciones.</p> <ul style="list-style-type: none"> <li>❖ Menú solicitud de Material: donde los mostrará dos submenús Material asignado, Solicitar material el mismo realizará la acción “crear pedido y eliminar”</li> <li>❖ Menú Solicitar por Funcionario: tiene las siguientes acciones, crear pedido y eliminar.</li> <li>❖ Menú Registro de Ingreso: tiene las siguientes acciones, Registro Nuevo y eliminar.</li> <li>❖ Menú Registro de Salida: tiene las siguientes acciones, mostrar.</li> </ul> <p>El <b>Funcionario</b>, puede realizar el pedido de material desde su plataforma web asimismo podrá ver la disponibilidad de material.</p>

Fuente: (Elaboración Propia)

### 3.2.4.8. Diagrama de Caso de Uso: Reportes

Figura N° 3. 11 Diagrama de Caso de Uso: REPORTE



Fuente: (Elaboración Propia)

Tabla N° 3. 13 Especificación de Caso de Uso: REPORTE

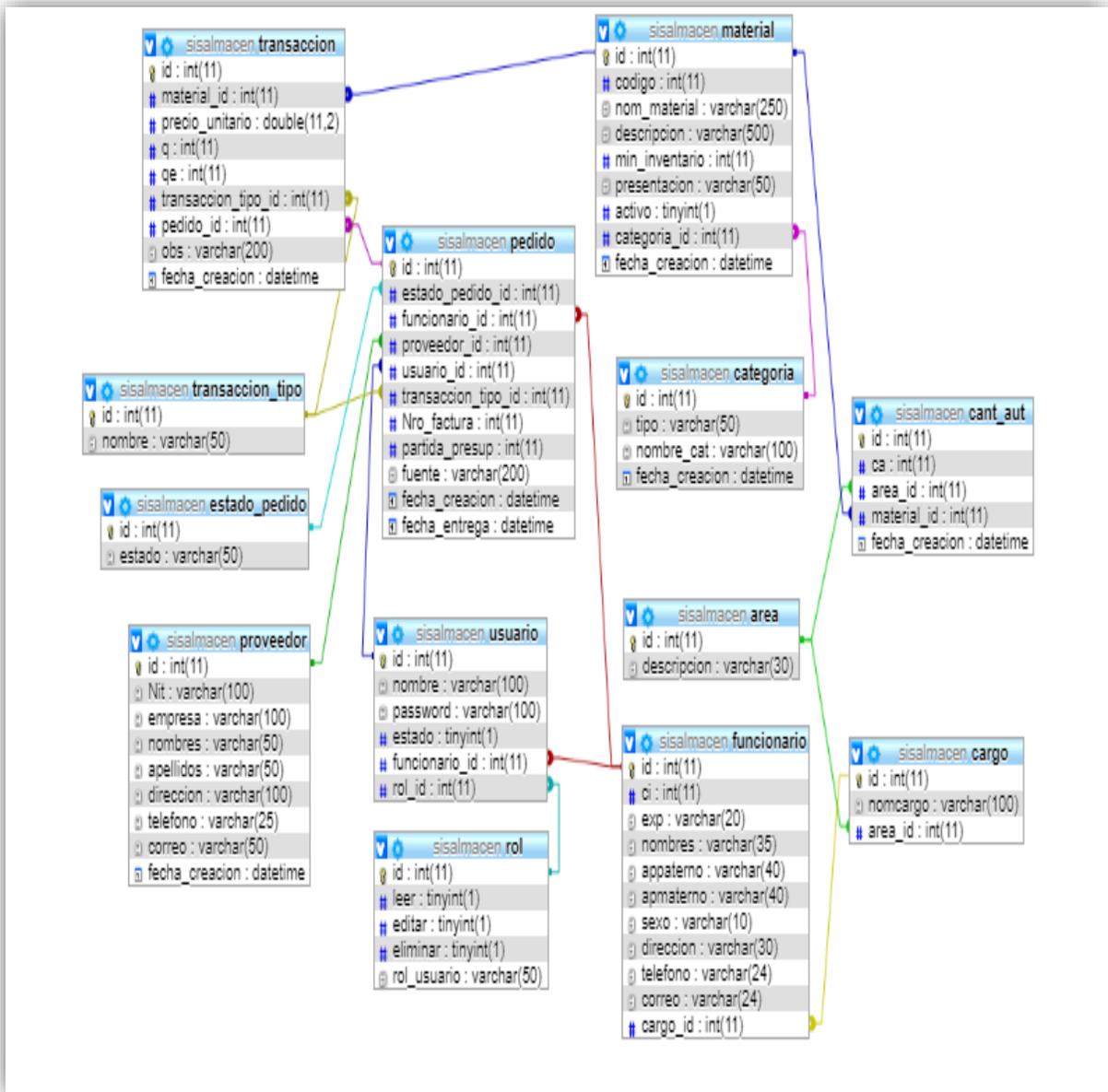
Caso de Uso :Administración de Usuarios – Reportes	
<b>Actor</b>	Jefe de Almacén
<b>Tipo</b>	Primario Esencial
<b>Descripción</b>	El jefe de almacén ingresa al módulo de Reportes y puede realizar las siguientes acciones: Visualizar y Generar los reportes generales y por persona.

Fuente: (Elaboración Propia)

### 3.2.4.9 Diagrama de Caso de Uso: Proveedores

### 3.4.3 Modelo De Contenido

Figura N° 3. 12 Modelo de Contenidos



Fuente: (Elaboración Propia)

### 3.4.4. Modelo De Navegación

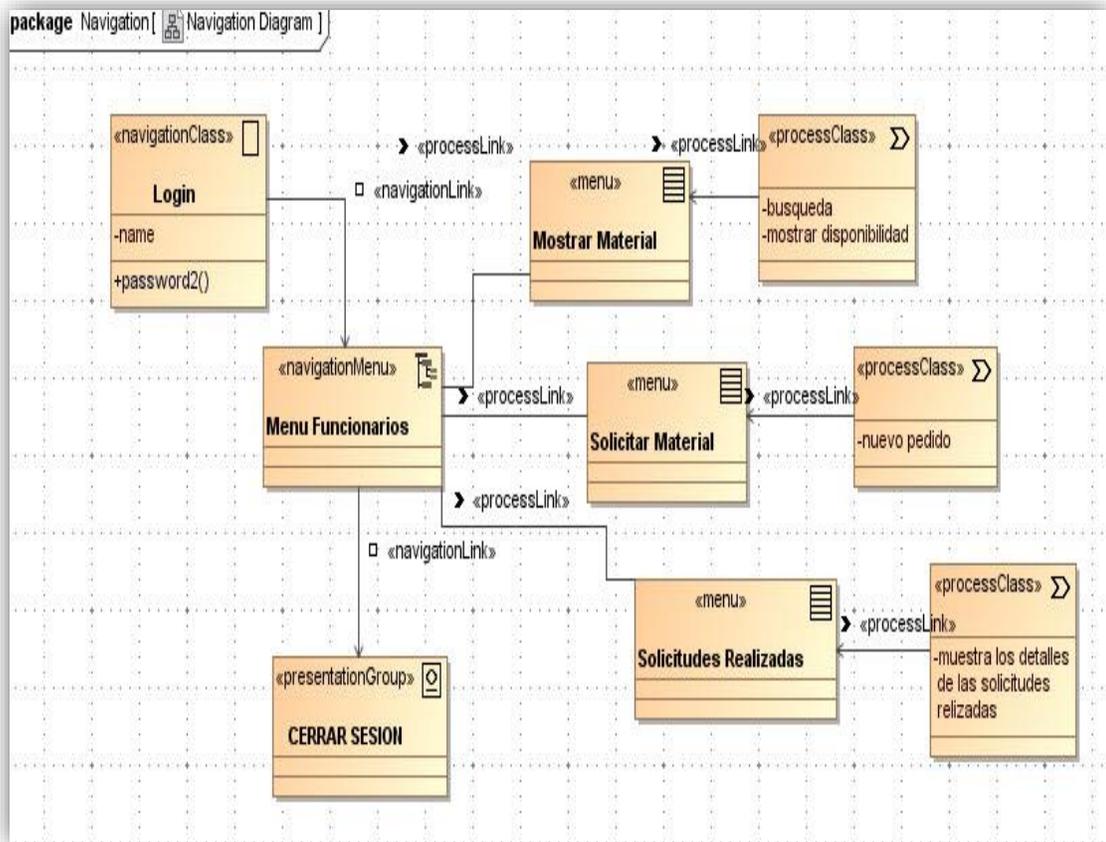
A continuación, Se realizará el modelado donde se podrá apreciar cómo interactúan los usuarios en el modelo de navegación del sistema.



### 3.4.4.2 Modelo de Navegación Administrador (Jefe de Almacén)

En la Figura 3.13 se puede visualizar el diagrama de navegación del usuario Funcionario donde se podrá ver los módulos que abarcara el sistema.

**Figura Nº 3. 14 Modelo de Navegación Funcionarios**



**Fuente:** (Elaboración Propia)

### 3.4.5 Modelo de Presentación

A continuación, se muestra los modelos de presentación del sistema de información para el control de manejo de Inventarios de Almacén del Instituto de Formación de Maestros de la Normal de El Alto (E.S.F.T.M.A).

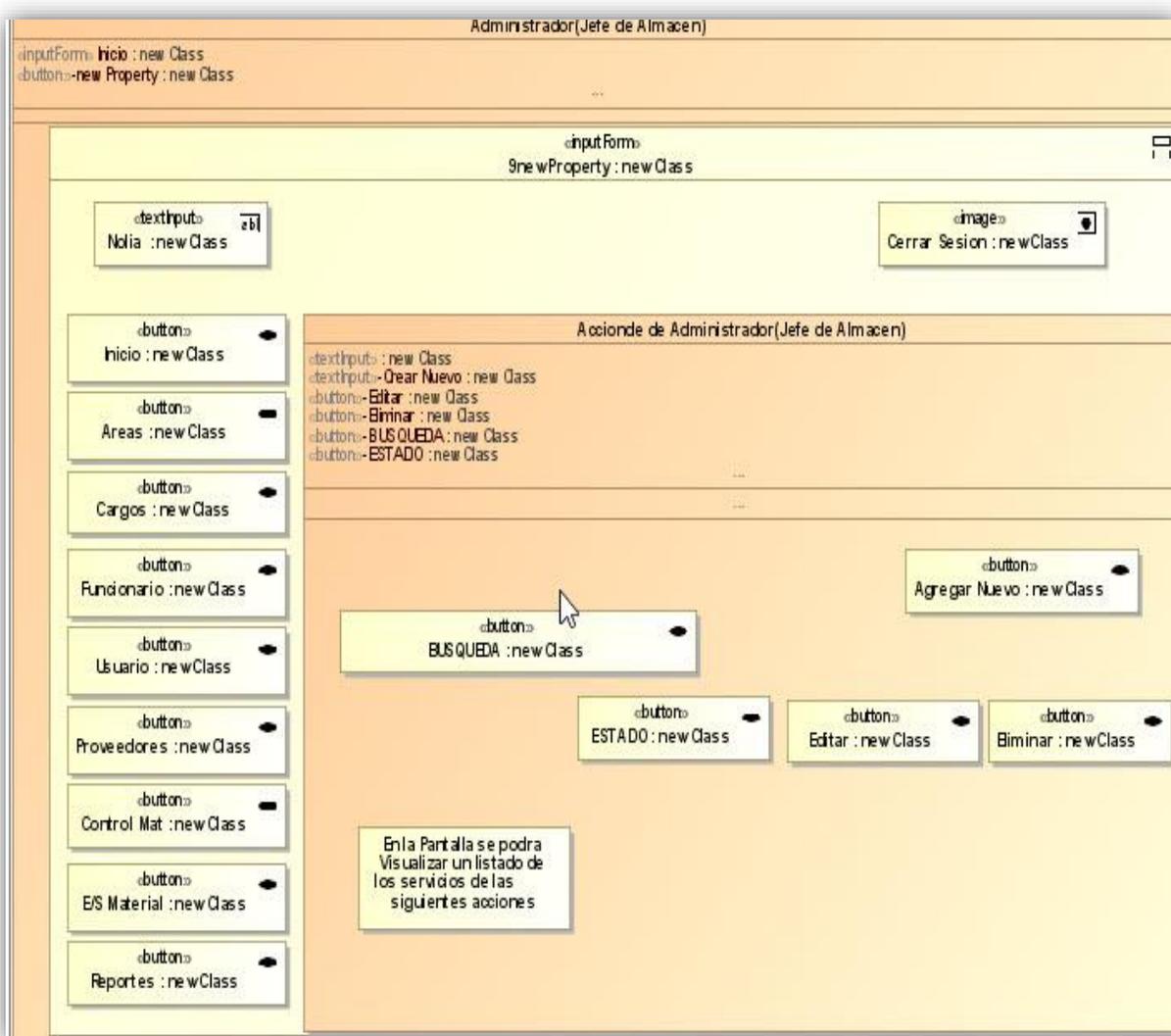
Según UWE propone la construcción de las páginas en forma de bosquejos derivadas desde la Figura 3.14 hasta Figura 3.15 donde se muestra como los

usuarios podrían acceder al sistema mostrando los menús correspondientes según el tipo de usuario.

- ✓ Modelo de Presentación: Administrador (Jefe de Almacén)
- ✓ Modelo de Presentación: Funcionarios.

### 3.4.5.1 Modelo de Presentación Administración

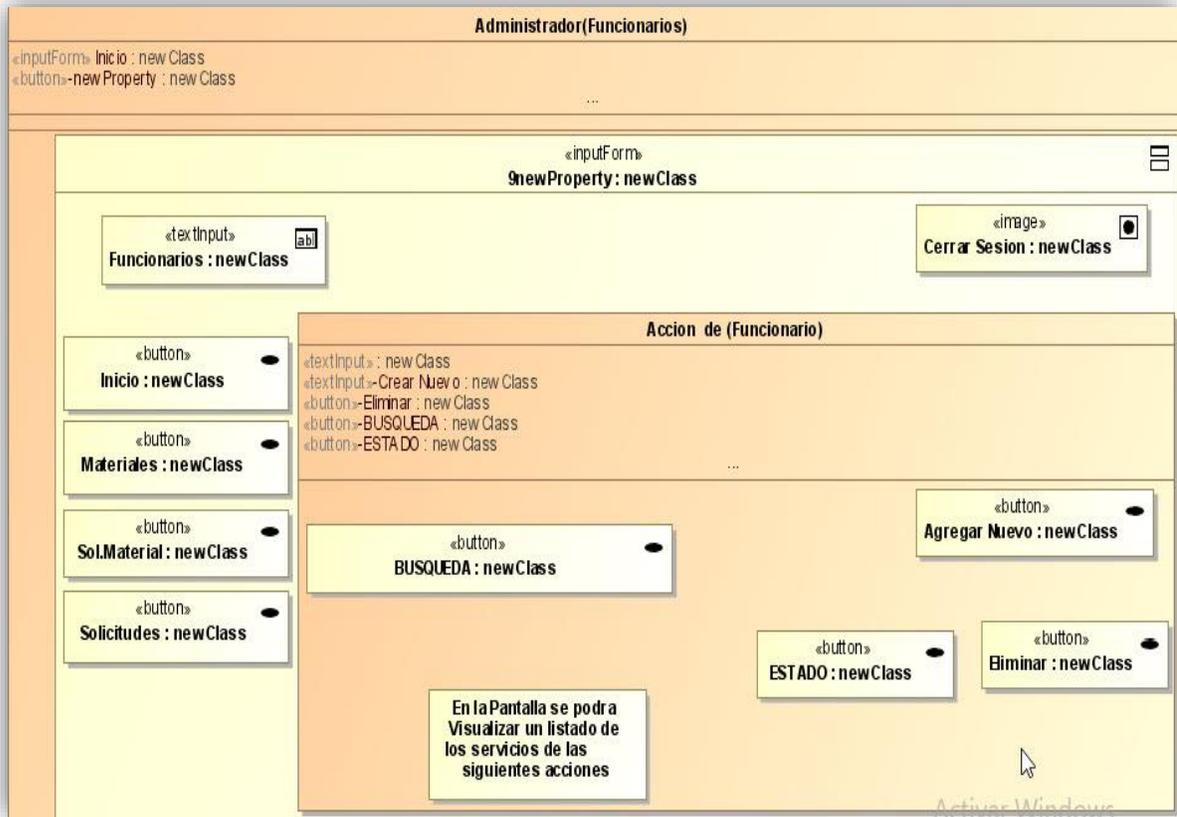
**Figura N° 3. 15 Modelo de Presentación: Administrador (Jefe de Almacén)**



Fuente: (Elaboración Propia)

### 3.4.5.2 Modelo de Presentación de Funcionario

Figura N° 3. 16 Modelo de Presentación: Funcionario



Fuente: (Elaboración Propia)

## 3.5 MODELO DE IMPLEMENTACION

En esta fase de implementación consiste en mostrar el desarrollo de la presentación de las interfaces del sistema y sus elementos construidos a partir del diseño del modelo de Presentación UWE.

### 3.5.1 Interfaz de Inicio de Sesión

Autenticarse en el sistema para acceder al sistema correspondientes, ingresando nombre usuario y contraseña. Se deberá ingresar con una cuenta de tipo usuario proporcionada.

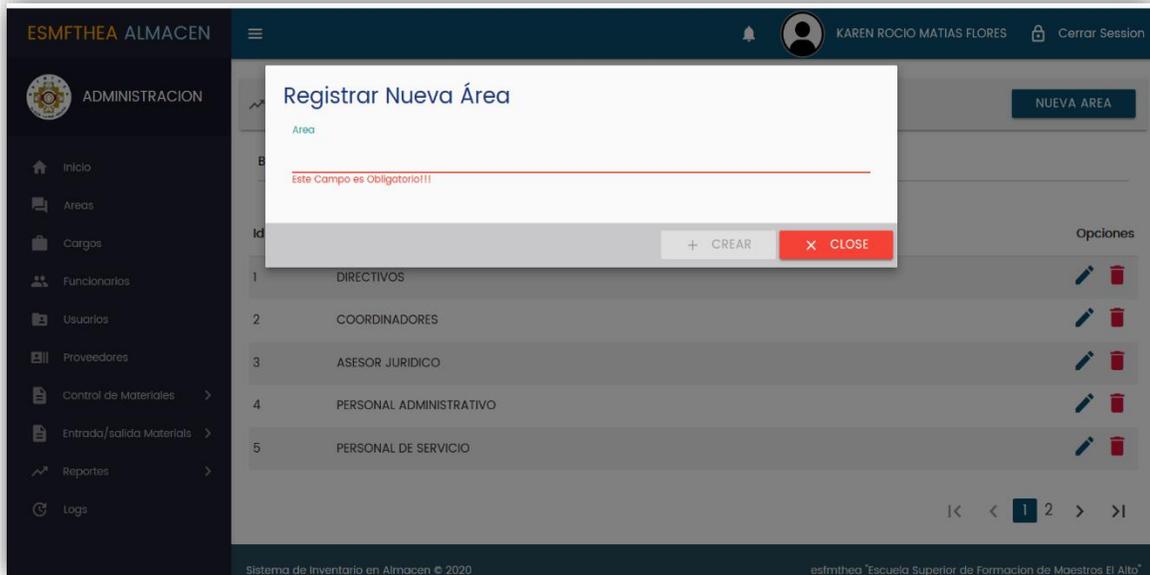
**Figura N° 3. 17 Autenticación Inicial con el sistema**



Fuente: (Elaboración Propia)

### 3.5.2 Interfaz de Áreas (Administrador)

**Figura N° 3. 18 Interfaz de Áreas (Administrador)**



Fuente: (Elaboración Propia)

### 3.5.3 Interfaz Cargos: Administrador

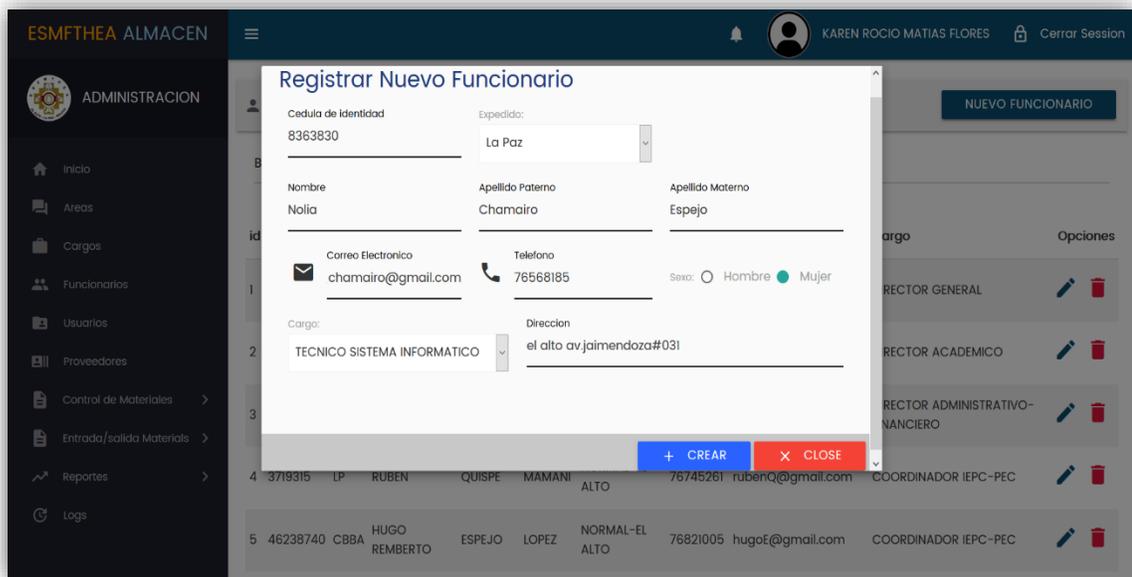
Figura Nº 3. 19 Interfaz Cargos (Administrador)



Fuente: (Elaboración Propia)

### 3.5.4. Interfaz Funcionario: Administrador

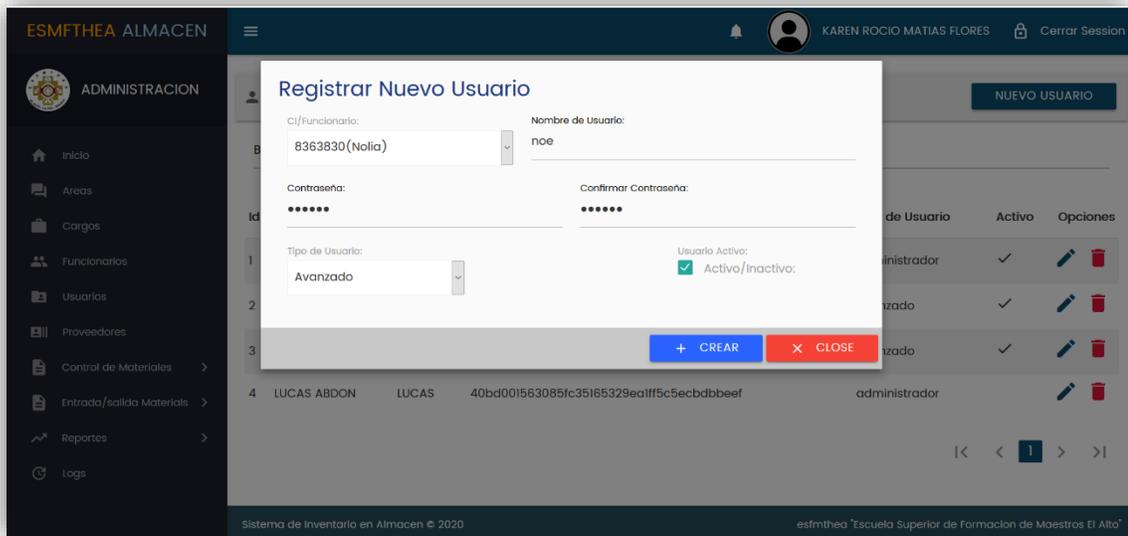
Figura Nº 3. 20 Interfaz Funcionario (Administrador)



Fuente: (Elaboración Propia)

### 3.5.5 Interfaz Usuario: Administrador

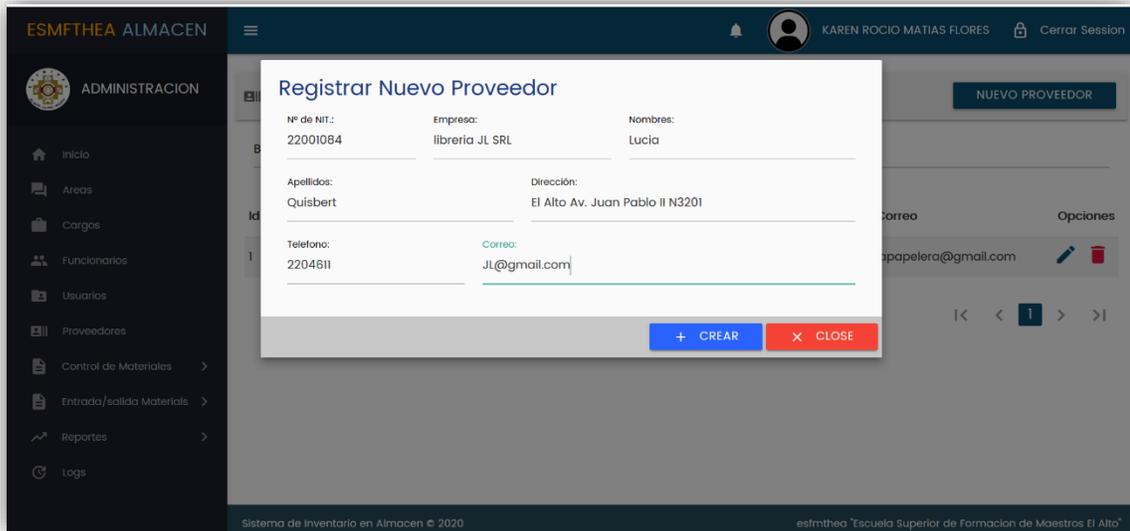
Figura N° 3. 21 Interfaz Usuario (Administrador)



Fuente: (Elaboración Propia)

### 3.5.6. Interfaz Proveedores: Administrador

Figura N° 3. 22 Reporte de Proveedores



Fuente: (Elaboración Propia)

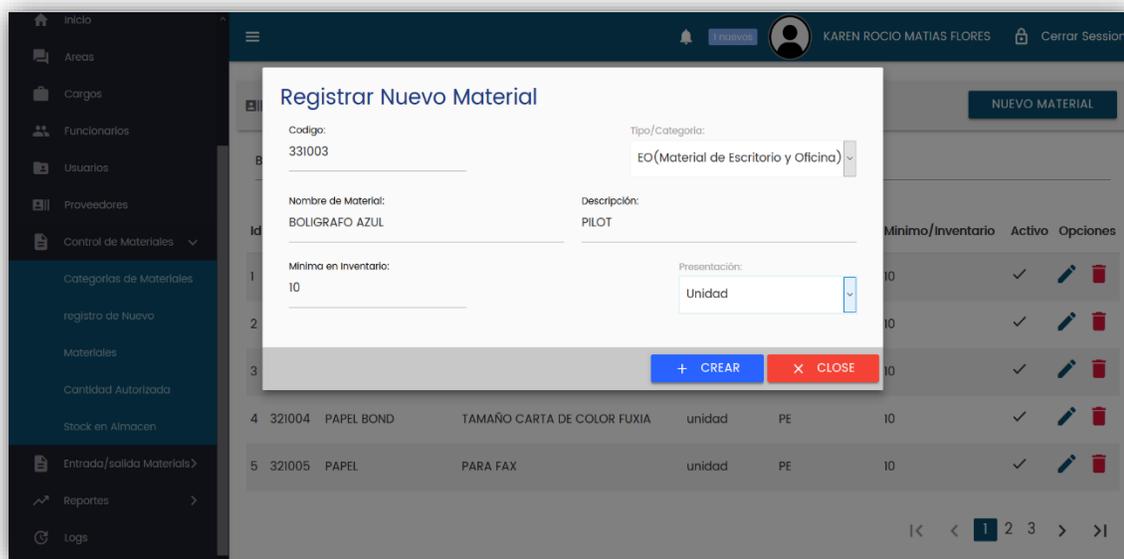
### 3.5.7. Interfaz Control de Materiales: Administrador

**Figura N° 3. 23 Interfaz Control de Materiales- Categoría**



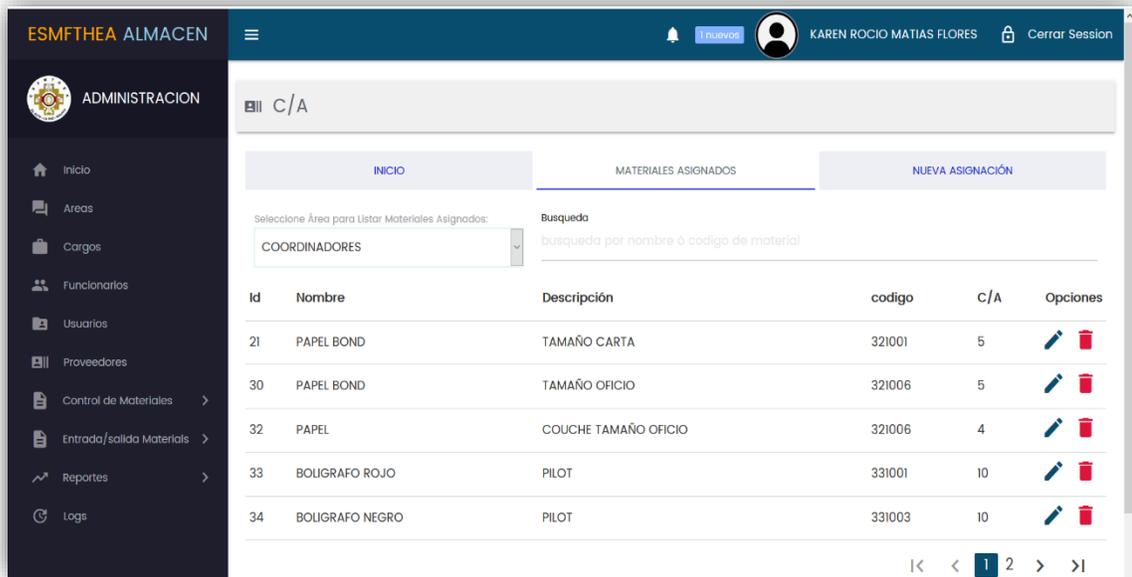
Fuente: (Elaboración Propia)

**Figura N° 3. 24 Interfaz Control de Materiales – Materiales**



Fuente: (Elaboración Propia)

**Figura N° 3. 25 Interfaz Control de Materiales – Control de Activos**



Fuente: (Elaboración Propia)

**Figura N° 3. 26 Interfaz Control de Materiales – Almacén**



Fuente: (Elaboración Propia)

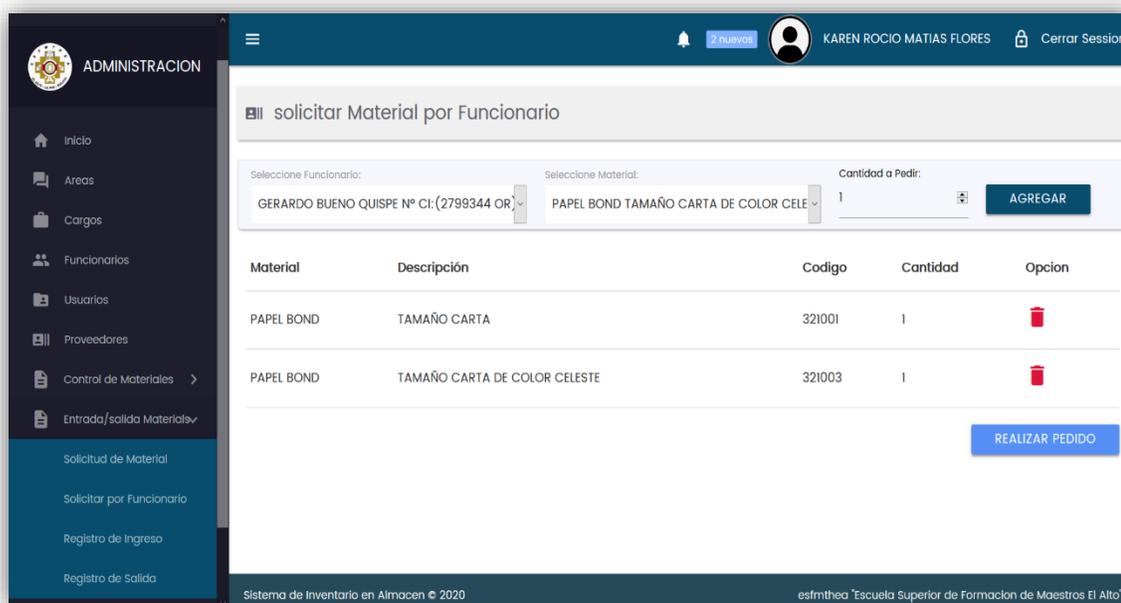
### 3.5.8 Interfaz Entrada y salida de Materiales: Administrador

**Figura Nº 3. 27 Interfaz de Entrada y Salida de Materiales – Solicitud de Materiales**



Fuente: (Elaboración Propia)

**Figura Nº 3. 28 Interfaz de Entrada y Salida de Materiales – Solicitud por Funcionario (Administrador)**



Fuente: (Elaboración Propia)

**Figura Nº 3. 29 Interfaz de Entrada y Salida de Materiales – Registro de Ingreso**

Selección Proveedor:  
Lucia Quisbert (libreria JL SRL)

Nit: 22001084

Partida Presup.: 351002      Nº de Factura: 022      Fuente: 03/0820

Selección Material:  
DETERGENTE EN POLVO ACE OMO (351001)      cantidad:      P/U:      AGREGAR

Item	Codigo	Material	Presentacion	Disponibles	Cantidad	P/Unitario	Total	Opciones
1	351002	ESCOBAS PLASTICAS DOBLES	unidad	0	10	15	150 Bs.	

Fuente: (Elaboración Propia)

**Figura Nº 3. 30 Interfaz de Entrada y Salida de Materiales – Registro de Salida**

Registros de Salidas

Busqueda: GER

Item	Funcionario	Cargo	Area	Fecha de Entrega	Opciones
3	GERARDO BUENO QUISPE	DIRECTOR GENERAL	DIRECTIVOS	2018-11-10 22:13:14	<i>i</i> <i>+</i>
9	GERARDO BUENO QUISPE	DIRECTOR GENERAL	DIRECTIVOS	2018-11-29 19:16:55	<i>i</i> <i>+</i>

Detalle de Entrega de Materiales

Item	Codigo	Material	Unidad	Pedido_id	Cantidad/solicitada	Cantidad/entregada	Observaciones	Fecha de entrega
------	--------	----------	--------	-----------	---------------------	--------------------	---------------	------------------

Fuente: (Elaboración Propia)

### 3.5.9 Interfaz Reportes: Administrador

**Figura Nº 3. 31 Interfaz de Reportes – Reporte General**

ADMINISTRACION

Reporte General

Para generar un Reporte Seleccione una fecha inicial y una fecha final

Fecha Inicial: 29 / 06 / 2018 Fecha Final: 05 / 08 / 2020

EXPLORAR PDF

Codigo	Tipo	Material Suministro	Unidad	transaccion	Fecha
1	321001	PE PAPEL BOND TAMAÑO CARTA	Pqte.	entrada	2018-11-10 20:52:19
1	321001	PE PAPEL BOND TAMAÑO CARTA	Pqte.	entrada	2018-11-11 12:11:29
1	321001	PE PAPEL BOND TAMAÑO CARTA	Pqte.	entrada	2018-11-14 00:00:34
3	321003	PE PAPEL BOND TAMAÑO CARTA DE COLOR CELESTE	unidad	entrada	2018-11-12 10:50:15

Sistema de Inventario en Almacen © 2020 esfmthea "Escuela Superior de Formación de Maestros El Alto"

Fuente: (Elaboración Propia)

**Figura Nº 3. 32 Interfaz de Reportes – Reporte por Persona (Administrador)**

ADMINISTRACION

Reporte por Persona

Para generar un Reporte Seleccione una fecha inicial y una fecha final

Fecha Inicial: 05 / 08 / 2018 Fecha Final: 06 / 08 / 2020 Funcionario: (2799344) GERARDO BUENO QUISPE

PDF

Item	Codigo	Mateial Suministro	Unidad	Solicitado	Entregado	Fecha de Entrega
1	321001	PAPEL BOND TAMAÑO CARTA	Pqte.	10	10	2018-11-10 22:13:14
2	321001	PAPEL BOND TAMAÑO CARTA	Pqte.	5	5	2018-11-29 19:16:55

Sistema de Inventario en Almacen © 2020 esfmthea "Escuela Superior de Formación de Maestros El Alto"

Fuente: (Elaboración Propia)

**Figura N° 3. 33 Interfaz de FUNCIONARIO**



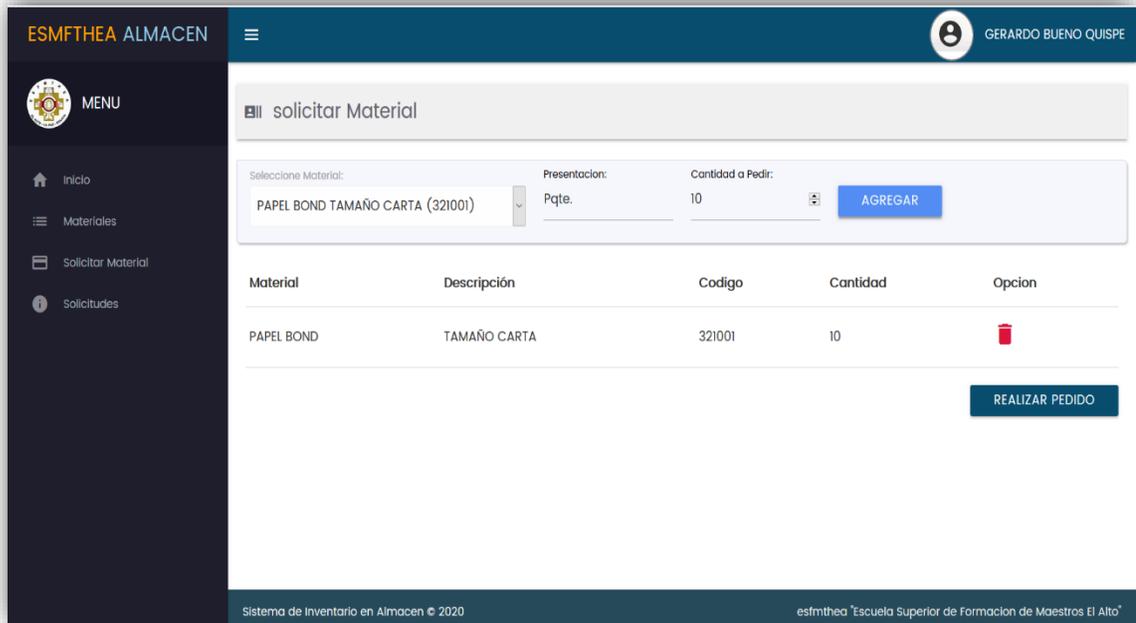
**Fuente:** (Elaboración Propia)

**Figura N° 3. 34 Interfaz de Materiales– (Funcionario)**

Codigo	Nombre de Material	Descripcion	Presentación	C/A	Disponible
321001	PAPEL BOND	TAMAÑO CARTA	Pqte.	10	✓
321006	PAPEL BOND	TAMAÑO OFICIO	Pqte.	10	ⓘ
321003	PAPEL BOND	TAMAÑO CARTA DE COLOR CELESTE	unidad	10	✓
321004	PAPEL BOND	TAMAÑO CARTA DE COLOR FUXIA	unidad	10	ⓘ
321005	PAPEL	PARA FAX	unidad	10	ⓘ

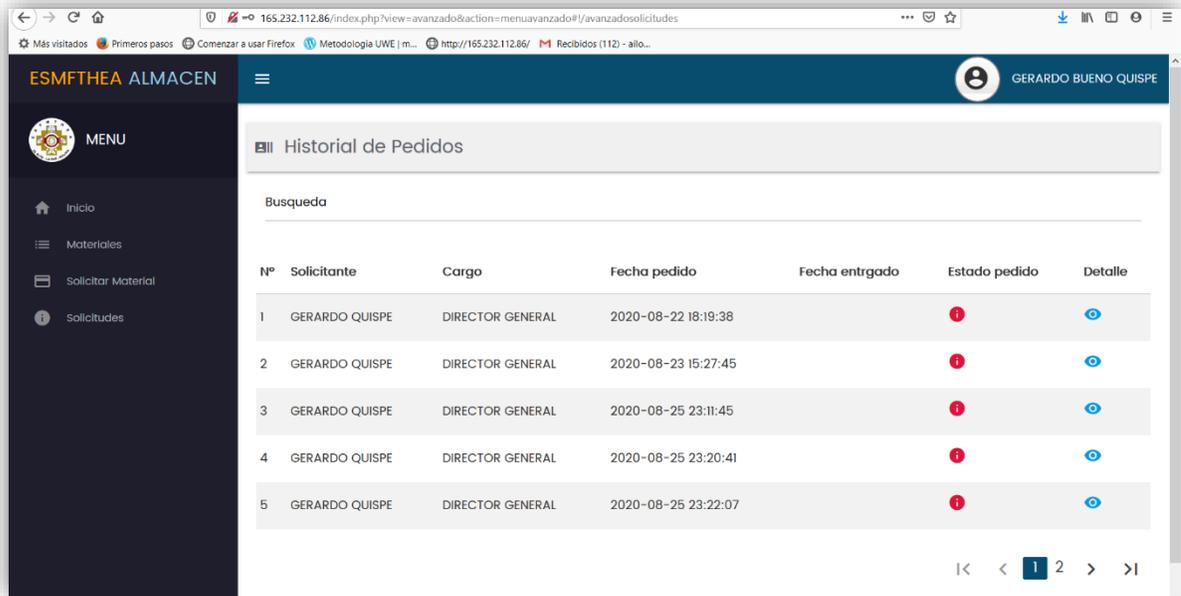
**Fuente:** (Elaboración Propia)

**Figura Nº 3. 35 Interfaz de Solicitar Material– (Funcionario)**

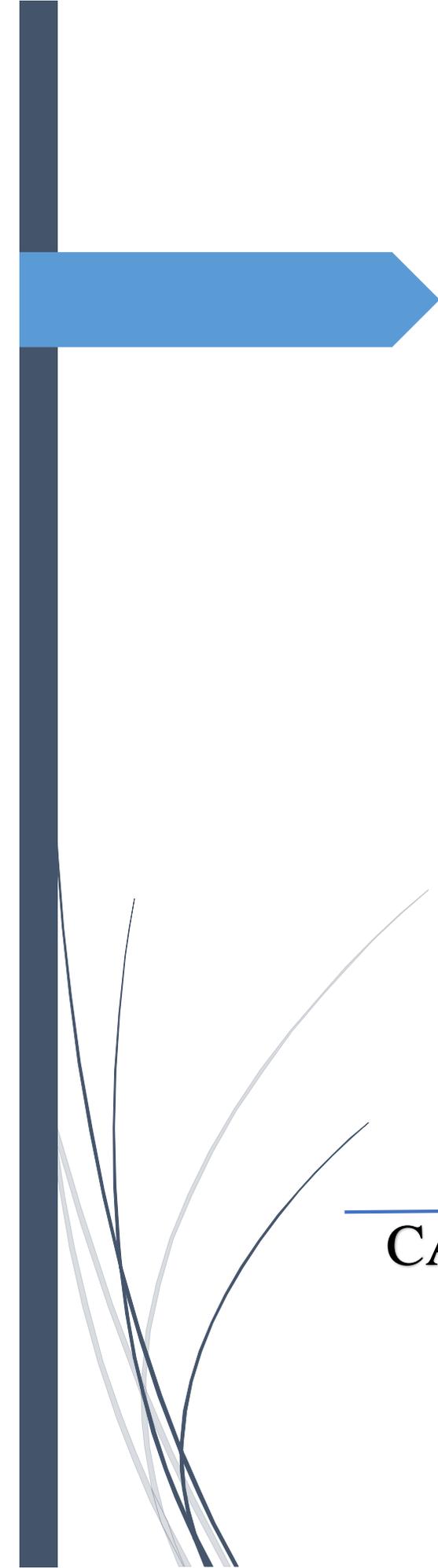


**Fuente:** (Elaboración Propia)

**Figura Nº 3. 36 Interfaz de Solicitud– (Funcionario)**



**Fuente:** (Elaboración Propia)



# CAPITULO IV

---

## CALIDAD Y SEGURIDAD

*INGENIERIA DE SISTEMAS - UNIVERSIDAD PÚBLICA EL ALTO*

## **4. CALIDAD Y SEGURIDAD**

### **4.1. PRUEBAS DE SOFTWARE**

Para Pressman, las pruebas de software son elementos críticos para la garantía de calidad de software y representa una revisión de las especificaciones, del diseño y la codificación. Los objetivos de las pruebas son:

La prueba es un proceso de ejecución de un programa con la intención de descubrir un error. Un buen caso de prueba es aquel que tiene una alta probabilidad de mostrar un error no descubierto hasta entonces.

Una prueba de error tiene éxito si descubre un error no detectado hasta entonces. La siguiente lista de comprobación proporciona un conjunto de características que llevan a un software fácil de probar:

- Operabilidad, “cuando mejor funcione, más eficiente se puede probar”.
- Observabilidad, “lo que se ve, se prueba”.
- Controlabilidad, “cuanto mejor podamos controlar el software, más se puede automatizar y optimizar”.
- Capacidad de descomposición, “controlando el ámbito de prueba, podemos aislar más rápidamente los problemas y llevar a cabo mejores pruebas de regresión”.
- Simplicidad, “cuanto menos haya que probar, más rápidamente podemos probarlo”.
- Estabilidad, “Cuantos menos cambios, menos interrupciones a las pruebas”.
- Facilidad de comprensión, “cuanta más información tengamos, más inteligentes serán las pruebas”. de los usuarios.

#### **4.1.1 Pruebas de Caja Blanca**

Según Pressman (citado por Yali, 2011), las pruebas de caja blanca es un método de diseño de casos de prueba que usa la estructura de control del diseño procedimental.

La prueba de caja blanca del software se basa en un minucioso examen de los detalles procedimentales. Se comprueban los caminos lógicos del software proponiendo casos de prueba que ejerciten conjuntos específicos de condiciones y/o ciclos. Se puede examinar el estado del programa en varios puntos para determinar si el estado real coincide con lo esperado o mencionado.

La prueba del camino básico es una técnica que permite obtener la complejidad lógica de un diseño procedimental y usar esta mediada como guía para la definición de un conjunto básico de caminos de ejecución y los casos de prueba obtenidos del conjunto básico garantizan que durante la prueba se ejecuta por lo menos una vez cada sentencia de programa.

La complejidad ciclomática, se basa en la teoría gráfica y se calcula de tres maneras:

- El número de regiones corresponde a la complejidad ciclomática.
- La complejidad ciclomática,  $V(G)$  de una gráfica de flujo  $G$ , se define como  $V(G) = A - N + 2$ .

Dónde:  $A$  es el número de aristas y  $N$  el número de nodos. 123

- La complejidad ciclomática  $V(G)$ , de una gráfica de flujo  $G$ , también se define como  $V(G) = P + 1$ .

Dónde:  $P$ , es el número de nodos predicado

En este caso para realizar las pruebas de caja blanca determinamos los casos de prueba que permitan la ejecución de cada uno de los caminos independientes y posteriormente ejecutamos cada caso de prueba y comprobamos los resultados esperados. Las pruebas de caja blanca que se muestran a continuación son de las funciones más relevantes del Sistema.

Archivo	Caminos básicos	Resultados
Autenticación php	1. login(), con complejidad ciclomática $V(G) = 3$	
	1. Se ha ingresado datos (usuario, contraseña) incorrectos. 2. El usuario está registrado pero su cuenta no se encuentra activo. 3. El usuario es válido y se encuentra activo.	1. Muestra el mensaje "El usuario y/o contraseña incorrectos". 2. Muestra el mensaje "El usuario no se encuentra activo". 3. Permite el acceso e inicia la sesión correctamente y envía al panel de administración dependiendo del perfil del usuario
USUARIO	crear usuario(), con complejidad ciclomática $V(G) = 5$	
	1. Se ha solicita la función de forma directa, con angular JS. 2. Se solicita por angular JS; se solicita adicionar usuario, pero no se han introducido los datos obligatorios; el nombre de usuario; el usuario ya está registrado. 3. Se solicita por Angular Js; se solicita adicionar usuario; los datos son correctos; el usuario es válido y único. 4. Se solicita por Angular JS; se solicita editar usuario, pero no se han introducido los datos obligatorios; el nombre usuario no es válido; el nombre usuario ya está registrado. 5. Se solicita por Angular JS; se solicita editar usuario; los datos son correctos; el usuario es válido y único.	1. Muestra el mensaje de error 404. 2. Muestra los mensajes de "Email inválido" o "el campo x es obligatorio". 3. Registra correctamente los datos y se muestra el mensaje de confirmación. 4. Se muestran los mensajes de error. 5. La actualización se hace con éxito.

	get_usuario(), con complejidad ciclomática $V(G) = 3$	
	<ol style="list-style-type: none"> <li>1. La consulta no se hace por medio de Angular JS</li> <li>2. La consulta se hace por medio de Angular JS; pero el id del usuario no es válido.</li> <li>3. La consulta se hace por Angular JS y el id del usuario es válido.</li> </ol>	<ol style="list-style-type: none"> <li>1. Muestra el mensaje de error 404.</li> <li>2. Muestra el mensaje "El usuario no existe"</li> <li>3. Retorna los datos del usuario seleccionado</li> </ol>
<b>mod_Funcionario.php</b>	listar funcionario (), con complejidad ciclomática $V(G) = 3$	
	<ol style="list-style-type: none"> <li>1. No se introduce una palabra clave para la búsqueda.</li> <li>2. El usuario introduce una palabra clave, pero no es una que se encuentra registrada en la base de datos del sistema.</li> <li>3. El usuario introduce una palabra clave; es correcta y la consulta se realiza correctamente.</li> </ol>	<ol style="list-style-type: none"> <li>1. Muestra el mensaje de error "introduce una palabra clave".</li> <li>2. Muestra el mensaje "No existen resultados, intente nuevamente".</li> <li>3. Muestra todas las coincidencias con la palabra clave.</li> </ol>
	guardarDatos(), con complejidad ciclomática $V(G) = 4$	

	<ol style="list-style-type: none"> <li>1. La consulta se hace por Angular JS; la petición es de un nuevo registro; los datos obligatorios no fueron introducidos.</li> <li>2. La consulta es realizada por Angular; la petición es registro nuevo; los datos son válidos.</li> <li>3. La consulta se hace mediante Angular JS; la petición envía el ID del cargo; el ID no es válido.</li> <li>4. La consulta se hace mediante Angular JS; la petición envía el ID del cargo; el ID es válido.</li> </ol>	<ol style="list-style-type: none"> <li>1. Muestra el mensaje de error "Llene los campos obligatorios".</li> <li>2. Registra los datos del nuevo animal en la base de datos y se muestra el mensaje de confirmación.</li> <li>3. Muestra el mensaje "Los datos fueron alterados"</li> <li>4. Actualiza los datos exitosamente y se muestra el mensaje de confirmación</li> </ol>
<b>mod_Control de Materiales.php</b>	guardar materiales(), con complejidad ciclomática $V(G) = 4$	
	<ol style="list-style-type: none"> <li>1. No se solicita el envío de datos del formulario.</li> <li>2. El usuario solicita el envío de datos; los datos obligatorios no son introducidos.</li> <li>3. El usuario solicita el envío de datos; los datos enviados son válidos, y el registro es nuevo.</li> <li>4. El usuario solicita el envío de datos; los datos enviados son válidos; se solicita actualización.</li> </ol>	<ol style="list-style-type: none"> <li>1. Re direcciona nuevamente al panel de seguimientos.</li> <li>2. Muestra el mensaje de error "El campo x es obligatorio".</li> <li>3. Registra los datos del seguimiento correctamente.</li> <li>4. Actualiza los datos del seguimiento exitosamente.</li> </ol>
Muestra lista existe, con complejidad ciclomática $V(G) = 2$		

	<ol style="list-style-type: none"> <li>1. El usuario no envía alguno de los parámetros requeridos</li> <li>2. El usuario envía los parámetros requeridos y válidos.</li> </ol>	<ol style="list-style-type: none"> <li>1. Muestra el error, show_404().</li> <li>2. Muestra toda la información solicitada por el usuario.</li> </ol>
<b>Entrada de materiales.php</b>	<b>function listarpedidode(), con complejidad ciclomática <math>V(G) = 3</math></b>	
	<ol style="list-style-type: none"> <li>1. El usuario no solicita envío de datos desde el formulario</li> <li>2. El funcionario no digita los datos obligatorios.</li> <li>3. El usuario envía todos los datos obligatorios</li> </ol>	<ol style="list-style-type: none"> <li>1. Re direcciona al panel del voluntario.</li> <li>2. Muestra el mensaje "Llene los campos obligatorios."</li> <li>3. Registra los datos y muestra el mensaje de confirmación</li> </ol>
	<b>Function edit_materiales(), con complejidad ciclomática <math>V(G) = 3</math></b>	
<ol style="list-style-type: none"> <li>1. El usuario no solicita detalles editar materiales del Listado.</li> <li>2. el usuario no digita los datos requeridos.</li> <li>3. Solicita envío de formulario; el usuario digita los datos requeridos.</li> </ol>	<ol style="list-style-type: none"> <li>1. Re direcciona al formulario de edición.</li> <li>2. Muestra el mensaje "El campo x es obligatorio ó Digite dato admitido".</li> <li>3. Registra y actualiza.</li> </ol>	
<b>modsalida de materiales.php</b>	<b>Function listarsalida(), con complejidad ciclomática <math>V(G) = 4</math></b>	

	<ol style="list-style-type: none"> <li>1.El usuario solicita registrar</li> <li>2.Solicita envío de formulario; el usuario no digita los datos requeridos.</li> <li>3.Solicita envío de formulario; el usuario digita los datos requeridos; solicita registro de nuevo formulario.</li> <li>4.Solicita envío de formulario; el usuario digita los datos requeridos; solicita actualización de un reporte de pedido.</li> </ol>	<ol style="list-style-type: none"> <li>1. Re direcciona al panel del Nota de pedido.</li> <li>2. Muestra el mensaje “El campo x es obligatorio”.</li> <li>3. Registra el nuevo pedido de material.</li> <li>4. Actualiza los datos del Nuevo reporte de pedido.</li> </ol>
<p>Funcion detallarsalida, con complejidad ciclomática <math>V(G) = 4</math></p>		
	<ol style="list-style-type: none"> <li>1. El usuario solicita ver datos; no envía el dato requerido.</li> <li>2. El usuario solicita ver datos; envía el dato requerido.</li> <li>3. El usuario solicita ver datos; envía el dato requerido; no existen los datos de material.</li> <li>4. El usuario solicita ver Datos; envía el dato requerido</li> </ol>	<ol style="list-style-type: none"> <li>1. Muestra mensaje de error, “el campo X es obligatorio”.</li> <li>2. Verifica si datos existe.</li> <li>3. Muestra el mensaje “No se puede generar el registro”.</li> <li>5. 4. Se registra exitosamente los datos de persona y el servicio solicitado.</li> </ol>
<b>reportes.php</b>	<p>Funcion construirtablapdfingreso con complejidad ciclomática <math>V(G) = 2</math></p>	
	<ol style="list-style-type: none"> <li>1. El usuario solicita listar el estado de movimiento de material.</li> <li>2. El usuario solicita cambiar estado de la registro de material.</li> </ol>	<ol style="list-style-type: none"> <li>1. Muestra todos los estados de movimiento de material.</li> <li>2. Cambia el estado de “Pendiente” a “Entregar” y de “Entregar” a “modificar”, previo respaldo de recibo</li> <li>3.</li> </ol>

	Function construirtablapdfentrega, con complejidad ciclomática $V(G) = 3$	
	<ol style="list-style-type: none"> <li>1. El usuario solicita reporte por fechas de movimiento de material.</li> <li>2. El usuario digita rango de fecha inexistente</li> <li>3. El usuario digita datos requeridos para el reporte</li> </ol>	<ol style="list-style-type: none"> <li>1. Muestra ó solicita el rango de fecha</li> <li>2. Muestra el mensaje “No existe datos para esta entrada”.</li> <li>3. Se genera el reporte, con los parámetros</li> </ol>
<b>proveedor.php</b>	crearproveedor, con complejidad ciclomática $V(G) = 2$	
	<ol style="list-style-type: none"> <li>1. El usuario solicita guardar servicio, no envía los datos requeridos.</li> <li>2. El usuario solicita guardar ficha, envía los datos requeridos</li> </ol>	<ol style="list-style-type: none"> <li>1. Muestra el mensaje “el campo x es requerido”.</li> <li>2. Registra los datos del Servicio.</li> </ol>
	listarproveedor, con complejidad ciclomática $V(G) = 2$	
	<ol style="list-style-type: none"> <li>3. El usuario solicita guardar, no envía los datos requeridos.</li> <li>4. El usuario solicita guardar y mostrar, envía los datos requeridos de proveedor.</li> </ol>	<ol style="list-style-type: none"> <li>3. Muestra el mensaje “el campo x es requerido”.</li> <li>4. Registra los datos de proveedor.</li> </ol>
<b>categoria.php</b>	crearcategoria, con complejidad ciclomática $V(G) = 3$	

	<ol style="list-style-type: none"> <li>1. El usuario solicita el registro de categoría</li> <li>2. El usuario solicita el envío de datos; los datos requeridos no son enviados.</li> <li>3. El usuario solicita el envío de datos; los datos requeridos son enviados</li> </ol>	<ol style="list-style-type: none"> <li>1. Re direcciona al panel del de reservas.</li> <li>2. Muestra el mensaje de error “el campo x es obligatorio.”</li> <li>3. Registra la datos de dirección de material según la categoría</li> </ol>
--	---	---

#### 4.1.2. Pruebas De Caja Negra

Las pruebas de caja negra se centran en los requisitos funcionales del software, es decir intenta encontrar errores de las siguientes categorías:

- Funciones incorrectas o ausentes.
- Errores de interfaz.
- Errores de estructura de datos en accesos de base de datos externos.
- Errores de rendimiento.
- Errores de inicialización y de terminación.

Las pruebas de caja negra tienen un enfoque complementario que tiene probabilidades de descubrir una clase de diferente de errores de los que se descubrirían con los métodos de caja blanca. Las pruebas de caja negra fueron realizadas en dos niveles:

- a) Pruebas unitarias, estas pruebas fueron realizadas en la fase diseño de casos de usos.
- b) Pruebas de integración, esta prueba fue realizada al finalizar la iteración del desarrollo de todos los módulos, con las pruebas de aceptación

### 4.1.3 Prueba de Carga

La velocidad con la que cargan nuestros sitios web, y aunque en cierta medida depende de muchos factores externos, ninguno de ellos genera tanto impacto como la falta de optimización de las páginas que componen el sitio, cuya responsabilidad recae únicamente sobre nosotros como diseñadores y desarrolladores

Un sitio que tarda mucho tiempo en cargar terminará por cansar al usuario, quien, sin pensarlo dos veces, cerrará la pestaña del navegador y de manera casi segura nunca volverá a visitarlo. Además de esto, los buscadores más importantes de la red toman en cuenta la velocidad con la que cargan los sitios como parte del SEO (Search Engine Optimization o Posicionamiento en Buscadores), por lo que, de manera obvia, también se ve afectada la posición que tendrán nuestras páginas en los resultados de búsqueda. A continuación, se realizó la prueba con la dirección <http://www.webpagetest.org> y el resultado el siguiente:

**Figura N°4. 1 Resultados de Análisis de Navegación de Sistema**

Tester: VM03-02-192.168.199.197 [View JSON result](#)  
 First View only [Raw page data](#) - [Raw object data](#)  
 Test runs: 3 [Export HTTP Archive \(.har\)](#)  
Re-run the test [View Test Log](#)

**Performance Results (Median Run - SpeedIndex)**

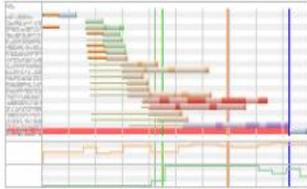
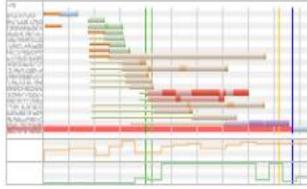
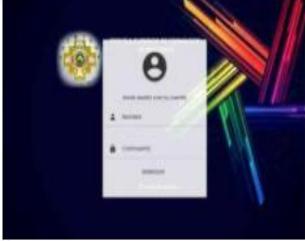
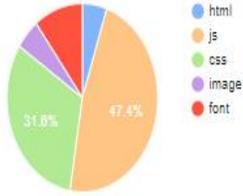
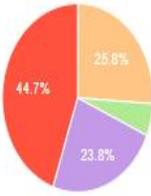
						Web Vitale		Document Complete			Fully Loaded			
	First Byte	Start Render	First Contentful Paint	Speed Index	Lat Painted Hero	Largest Contentful Paint	Cumulative Layout Shift	Time	Requets	Byte In	Time	Requets	Byte In	Cost
First View (Run 3)	0.259s	0.800s	0.840s	1.979s	2.000s	1.507s	0.024	1.936s	19	658 KB	2.068s	20	658 KB	\$5---

[Plot Full Results](#)

**Fuente:** (Elaboración Propia)

**Figura N°4. 2 Datos de Navegación**

**Test Results**

Run 1:	Waterfall	Screenshot
First View (1.845s)		
Run 2:	Waterfall	Screenshot
First View (4.475s)		
Run 3:	Waterfall	Screenshot
First View (1.936s)		
<u>Content Breakdown</u>	<p><b>Requests</b></p>  <ul style="list-style-type: none"> <li>● html</li> <li>● js</li> <li>● css</li> <li>● image</li> <li>● font</li> </ul>	<p><b>Bytes</b></p>  <ul style="list-style-type: none"> <li>● html</li> <li>● js</li> <li>● css</li> <li>● image</li> <li>● font</li> </ul>

**Fuente:** (Elaboración Propia)

## 4.2 MÉTRICAS DE CALIDAD DE SOFTWARE

En este capítulo se verá el desarrollo de la medición de la calidad de software mediante la métrica iso-9126 la cual es un estándar internacional para la evaluación de software, que establece que cualquier componente de la calidad pueda ser descrito por las características de Funcionalidad, Confiabilidad, Mantenibilidad, Usabilidad y Portabilidad.

### 4.2.1 Funcionabilidad

La funcionabilidad no se puede medir directamente por esta razón corresponde derivar medidas directas como es el punto función que cuantifica el tamaño y la complejidad del sistema en términos de las funciones del usuario. Para la funcionalidad o medición del sistema, se debe determinar las siguientes características.

**Tabla N° 4. 1 Características de la Funcionabilidad**

CARACTERISTICAS	DESCRIPCION
<b>Número de Entradas de Usuario</b>	Se origina al usuario, cuando realiza el ingreso de datos orientado a la aplicación.
<b>Número de Salidas de Usuarios</b>	Se cuenta cada salida que proporciona información orientada a la aplicación.
<b>Número de peticiones de Usuario</b>	En una entrada en línea que lleva a la generación de algunas respuestas inmediatas por parte del software.
<b>Número de archivos</b>	Se cuenta cada archivo lógico maestro, cada archivo lógico interno que es un agrupamiento lógico de datos como ser parte de una base de datos o archivo independiente.
<b>Número de interfaces externas</b>	Se cuenta todas las interfaces legibles por máquina.

**Fuente:** (Elaboración Propia)

➤ **Número de Entradas de Usuario**

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

**Tabla Nº 4. 2 Número de Entradas de Usuario**

ENTRADAS DE USUARIO		
1	Módulo Área	3
2	Módulo Cargo	3
3	Módulo Usuario	4
4	Módulo Funcionario	3
5	Módulo proveedores	3
6	Módulo Control de Materiales	9
7	Módulo Entrada y salida de material	5
8	Módulo reportes	2
	<b>TOTAL</b>	<b>32</b>

**Fuente:** (Elaboración Propia)

➤ **Número de Salidas de Usuarios**

Se cuenta cada salida que proporciona al usuario información orientada a la aplicación. Este contexto la salida se refiere a informes, pantallas, mensajes de error, etc. Los elementos de datos particulares dentro de un informe no se cuentan de forma separada.

**Tabla Nº 4. 3 Número de Salidas de usuario**

ENTRADAS DE USUARIO		
1	Módulo Área	2
2	Módulo Cargo	3
3	Módulo Usuario	3
4	Módulo Funcionario	3
5	Módulo proveedores	1
6	Módulo Control de Materiales	8
7	Módulo Entrada y salida de material	8
8	Módulo reportes	2
	<b>TOTAL</b>	<b>30</b>

**Fuente:** (Elaboración Propia)

➤ **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 separada.

**Tabla N° 4. 4 Número de Peticiones de Usuario**

ENTRADAS DE USUARIO		
1	Módulo Área	1
2	Módulo Cargo	1
3	Módulo Usuario	1
4	Módulo Funcionario	1
5	Módulo proveedores	1
6	Módulo Control de Materiales	4
7	Módulo Entrada y salida de material	2
8	Módulo reportes	2
	<b>TOTAL</b>	<b>12</b>

**Fuente:** (Elaboración Propia)

➤ **Numero de archivos**

Se cuenta archivo maestro lógico

**Tabla N° 4. 5 Número de Archivos**

ENTRADAS DE USUARIO		
7	Módulo Entrada y salida de material	2
8	Módulo reportes	2
	<b>TOTAL</b>	<b>4</b>

**Fuente:** (Elaboración Propia)

➤ **Numero de interfaces externas**

Se cuenta todas las interfaces legibles por la maquina

**Tabla N° 4. 6 Número de Interfaces Externas**

NUMERO DE INTERFACES EXTERNAS		
1	Login	1
	TOTAL	1

**Fuente:** (Elaboración Propia)

Para realizar el cálculo de la cuenta total con factores de ponderación se debe tomar en cuenta las siguientes: Tabla N° 3. 10 al Tabla N° 3. 14

**Tabla N° 4. 7 Factores de Ponderación**

PARAMETROS DE MEDIDA	CUENTA	FACTOR simple	TOTAL
1 Nro. de Entradas de Usuarios	32	3	96
2 Nro. de Salidas de usuarios	30	4	120
3 Nro. de Peticiones de Usuarios	12	3	36
4 Nro. de Archivos	4	7	28
5 Nro. de Interfaces Externas	1	5	5
<b>CUENTA TOTAL</b>			285

**Fuente:** (Elaboración Propia)

Los valores de ajuste de complejidad según las respuestas a las siguientes preguntas que se muestra en las siguientes tablas:

**Tabla N° 4. 8 Valores de Ajuste de Complejidad**

Nro.	FACTOR DE COMPLEJIDAD	Fi						
		Sin Influencia 0	Incidental 1	Moderada 2	Medio 3	Significativa 4	Esencial 5	
1	¿Requiere el sistema copia de seguridad y recuperación?						X	5
2	¿Requiere comunicación de datos?						X	5
3	¿Existen funciones de procesos distribuidos?					x		4

Nro.	FACTOR DE COMPLEJIDAD						Fi	
		Sin Influencia 0	Incidental 1	Moderada 2	Medio 3	Significativa 4		Esencial 5
4	¿El rendimiento es crítico				X		3	
5	¿Sera ejecutado el sistema en entorno existente y fuertemente utilizado?						X	5
6	¿Entrada de datos EN LINEA?						X	5
7	¿Requiere la entrada de datos interactiva que las transiciones de entrada se llevan a cabo sobre múltiples pantallas o variadas opciones?						X	5
8	¿Se actualizan los archivos maestros de forma interactiva?						X	5
9	¿Son complejas de las entradas de salidas de archivos?				X			3
10	¿Lógica del proceso Interno Compleja?				X			3
11	¿Se diseña el código para ser reutilizable?						X	5
12	¿Están incluidas en el diseño conversiones de instalación?					x		4
13	¿Instalaciones Múltiples?			X				2
14	¿Facilidad de Cambios?						X	5
Factor de Complejidad Total (FCT)							59	

**Fuente:** (Elaboración Propia)

Para calcular los puntos función (PF), utilizaremos la relación siguiente:

$$PF = \text{Cuenta Total} * (0.65 + 0.01 * \sum Fi)$$

Donde:

Cuenta Total: Nivel de complejidad del sistema con respecto al usuario

**(0.65+0.01\*∑Fi):** Ajuste de complejidad según el dominio de la información.

**0.01:** Factor de conversión, es decir un error de 1%

**0.65:** Valor mínimo de ajuste.

Calculando el punto función según la ecuación:

$$PF=285*[0.65+0.01*59]$$

$$PF=353.4$$

Si calculamos al 100% el nivel de confianza consideramos la sumatoria de  $F_i=70$  como el máximo valor de ajuste de complejidad se tiene:

$$PF \text{ max}=\text{Cuenta Total} *[1+0.01*\sum F_i]$$

$$PF \text{ max}=285*[0,65+0.01*70]$$

$$PF=384.75$$

Con los máximos valores de ajuste de complejidad se tiene que la funcionalidad real es:

$$\text{Funcionalidad} = \frac{353.4}{384.75} = 0,91$$

$$\text{Funcionalidad}=0,91*100=91\%$$

Entonces la funcionalidad del sistema es un 90% esto quiere decir que el sistema tiene un 90% que funcione sin riesgos de fallo y operatividad constante y 10% de colapso del sistema.

#### 4.2.2 Confiabilidad

La confiabilidad del sistema se define como la probabilidad de operación libre de fallos de programa de computación en un entorno determinado y durante un tiempo específico.

Para calcular la confiabilidad del sistema se toma en cuenta el periodo de tiempo en el cual se ejecuta y se obtiene muestras.

$$F(t) = f^{*(-\mu*t)}$$

En el inicio de ejecución  $t_0=0$  lo que significa el tiempo inicial en el cual dará inicio el funcionamiento del sistema.

$$F(0) = f^{*(-\mu*t_0)}$$

Se observa el trabajo del sistema hasta que produce una falla en el instante T, el cual se aproxima a una variable aleatoria continua.

Como se aproxima a variables aleatorias a variables continuas, la confiabilidad ha sido obtenida en términos probabilísticos.

Entonces el término en el cual el sistema trabaja sin falla está dado por la ecuación (2) y tiempo en el cual no falla el sistema está dado por ecuación (30).

**$P(T \leq t) = F(t)$**  (2) probabilidad de fallas

**$P(T \leq t) = 1 - F(t)$**  (3) Probabilidad de trabajo sin falla

En un periodo de 20 días como tiempo de prueba se define de cada 10 ejecuciones 1 falla

Conociendo la funcionalidad del 90% del sistema calculamos para el periodo establecido

$$\begin{aligned} P(T \leq t) &= 1 - F(t) \\ F(t) &= e^{-\lambda t} \\ (t) &= 0.91 \times e^{-\frac{1}{10} \times 20} \\ (t) &= 0.1231 \times 100 \\ F(t) &= 12.31 \\ \text{Con falla: } &1 - F(t) \\ P(T \leq t) &= 1 - 0.1231 \\ &= 0.8769 \times 100 \\ &= 87.69 \end{aligned}$$

La confiabilidad del sistema es del 87% en un periodo de 20 días como tiempo de prueba.

#### 4.2.3 Mantenibilidad

El mantenimiento se desarrolla para mejorar el sistema en respuesta a los nuevos requerimientos que la empresa tenga y los reglamentos que está regida por la misma.

El estándar IEE94 sugiere un índice de madurez del software (IMS) que proporciona un indicador en la estabilidad de un producto, se lo determina con la siguiente fórmula.

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

**MT**=Números de módulos la versión actual.

**Fa**=Números de Módulos en la versión actual que se ha añadido.

**Fc**=Numero de Módulos en la versión actual que se han Cambiado.

**Fd**=Numero de Módulos en la versión anterior que se han borrado en la versión actual.

Calculando el IMS

$$\text{IMS} = [8 - (0 + 0 + 0)] / 9$$

$$\text{IMS} = 0,88$$

Con lo que podemos decir que el nuevo sistema tiene una estabilidad de 83% es la facilidad de mantenimiento el 17% restantes es el margen de error correspondiente a los cambios y modificaciones efectuados desde el prototipo de la versión actual.

Puesto que es un sistema diseñado con los requerimientos actuales con el tiempo surgirán nuevos requerimientos los cuales cambiarán el valor índice de madurez del software.

Mantenimiento Correctivo, el sistema presenta diseño modular y es por eso que se tolera variaciones en su corrección.

Mantenimientos Adaptativo, se realiza cuando en la organización se produce algún cambio haciendo que el sistema sufra modificaciones.

El sistema por su programa modular permitirá fácilmente hacer modificaciones en sus módulos o integrar nuevos sistemas.

#### 4.2.4 Usabilidad

La usabilidad es lo mismo decir la factibilidad de uso, esta métrica nos muestra el costo de aprender a manejar el producto, lo cual se calcula con la siguiente formula:

$$FU=[(\text{Sum}(xi)/n)*100]$$

Donde:

**Xi:** es la sumatoria de valores

**n:** es el número de preguntas

para responder a las preguntas se debe considerar la siguiente tabla

**Tabla Nº 4. 9 Ajustes de Preguntas**

ESCALA	VALOR
Muy bueno	5
Bueno	4
Regular	3
Malo	2
Pésimo	1

**Fuente:**(Elaboración Propia)

**Tabla Nº 4. 10 Ajuste de Preguntas**

PREGUNTAS	Respuestas		Ponderación %
	SI	NO	
¿Puede Utilizar con facilidad el sistema?	4	1	80%
¿Puede Controlar operaciones que el sistema solicita?	4	1	80%
¿Las Respuestas del sistema son complicadas?	1	4	80%
¿El Sistema permitió la retroalimentación de información?	5	0	100%
¿El sistema cuenta con interface agradable a la vista?	5	0	100%

PREGUNTAS	Respuestas		Ponderación %
	SI	NO	
¿La respuesta del sistema es satisfactoria?	4	1	80%
¿Le parece complicada las funciones del sistema?	1	4	80%
¿Se hace difícil o dificultoso aprender a manejar el sistema?	1	4	80%
¿Los resultados que proporciona el sistema facilitan el trabajo?	5	0	100%
¿Durante el uso del sistema se produjo errores?	1	4	80%
<b>USABILIDAD</b>			<b>86%</b>

**Fuente:**(Elaboración Propia)

Existe un 86% de comprensión o entendimiento de los usuarios con respecto a la capacidad del sistema.

#### **4.2.4 Portabilidad**

Para la portabilidad del sistema se tomará en cuenta dos aspectos como ser a nivel aplicación, nivel hardware la portabilidad la dividimos en dos secciones del lado del servidor y del lado del cliente.

##### **Portabilidad lado del Cliente**

A nivel sistema de software. El sistema porta bajo los siguientes sistemas operativos de la familia Microsoft Windows 2000, Windows milenio, Windows XP, Windows Vista, Windows 7, Windows 8 y todo el sistema operativo libre de Linux.

##### **Portabilidad del lado del Servidor**

A nivel de base de datos se utiliza base de datos creada en MySQL, la portabilidad se muestra que la base de datos puede ser migrada a SQL server, Oracle, Postgres y MariaDB.

## 4.3 SEGURIDAD INFORMATICA

### 4.3.1 Sistema De Gestión De Seguridad De La Información Iso-27002

La ISO-27002 evalúa y rectifica la implementación mediante el cumplimiento de normas, para que exista una continuidad de mejoramiento de un conjunto de controles que permitan reducir el riesgo de sufrir incidentes de seguridad en el funcionamiento de la empresa en cuanto a la seguridad de la información a continuación, se describe los puntos que se tomaran en cuenta.

#### 4.3.1.1 Seguridad Lógica

- **Gestión de Comunicación y Operaciones**

Los respaldos (Back-up) de la base de datos del sistema, se deberá realizar las precauciones de acuerdo a la siguiente Tabla N° 4.11

**Tabla N° 4. 11 Gestión de Comunicación y Operaciones**

DESCRIPCION	DURACION
En periodos de registro de funcionarios y solicitud e materiales	1 vez por semana
En periodo de registro de mantenimientos	1 vez por semana
En periodo de registro de Documento de ingreso de materiales	1 ves por semana
En periodo de registro de proveedores	1 vez por mes
En periodo registro usuarios	1 vez por mes

**Fuente:** (Elaboración Propia)

Al usuario Administrador se recomienda cambiar periódicamente el password.

#### 4.3.1.2 Seguridad Física

Seguridad Física y del Entorno

Se recomienda realizar, Back-up más de tres copias, estos sean almacenados en distintos lugares.

Los back-up de la base de datos, deberán estar protegidas y tener acceso al personal autorizado.

#### **4.3.1.3 Seguridad Organizativa**

La información del sistema debe tener un nivel de protección, como ser. El manejo de los back-up de acuerdo a las fechas en el que se realizan las copias.

#### **4.3.1.4 Seguridad del Sistema**

Para la seguridad del sistema se consideran las siguientes precauciones, los cuales son:

- Autenticación de los usuarios
- Manejo de privilegios y tipos de usuarios en el sistema.
- Manejo de vistas.
- Integridad y control de datos por url's
- Manejo de sesiones y validación de campos.
- Encriptación con el algoritmo SHA1 en las contraseñas o passwords.

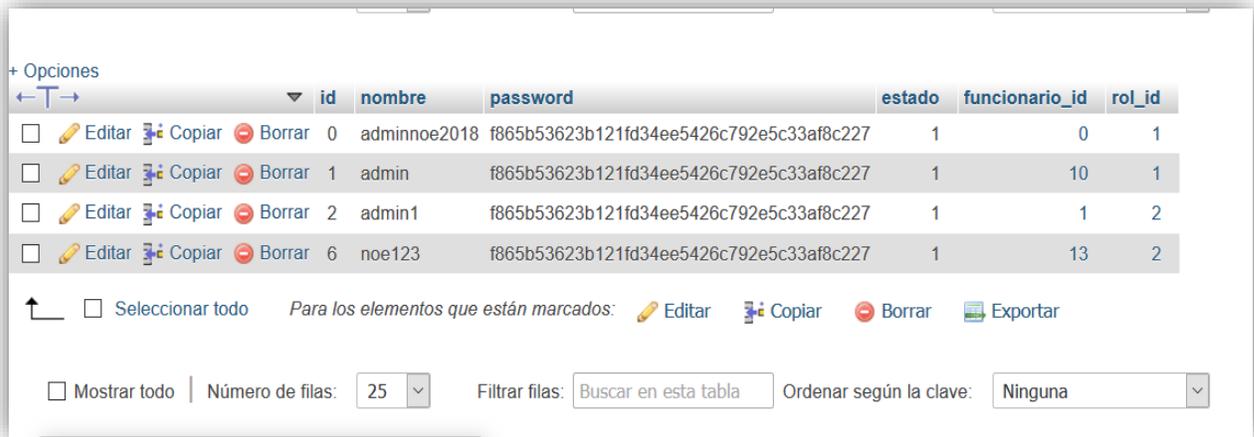
El SHA (Secure Hash Algorithm, Algoritmo de Hash Seguro) es una familia de funciones hash de cifrado publicadas por el Instituto Nacional de Estándares y Tecnologías (NIST). La primera versión del algoritmo fue creada en 1993 con el nombre SHA, producen una salida con un resumen de 160 bits (20 bytes) de un mensaje que puede tener un tamaño máximo de  $2^{64}$  bits. (Soliz, 2012)

#### **4.3.1.5 Políticas de Registros de Eventos**

Los logs o eventos realizados por los usuarios en el sistema de información permite el seguimiento y monitoreo de los accesos de los usuarios a las diferentes funciones que ofrece el sistema como: Ingresos al sistema, actualizaciones o eliminaciones que el usuario haya realizado, las peticiones al sistema.

Se diseñó una tabla exclusiva en la base de datos para realizar las auditorias correspondientes. En la siguiente figura se muestra la tabla para estos eventos.

**Figura N°4. 3Tabla de registro de eventos**



	id	nombre	password	estado	funcionario_id	rol_id
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	0	adminnoe2018	f865b53623b121fd34ee5426c792e5c33af8c227	1	0	1
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	1	admin	f865b53623b121fd34ee5426c792e5c33af8c227	1	10	1
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	2	admin1	f865b53623b121fd34ee5426c792e5c33af8c227	1	1	2
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	6	noe123	f865b53623b121fd34ee5426c792e5c33af8c227	1	13	2

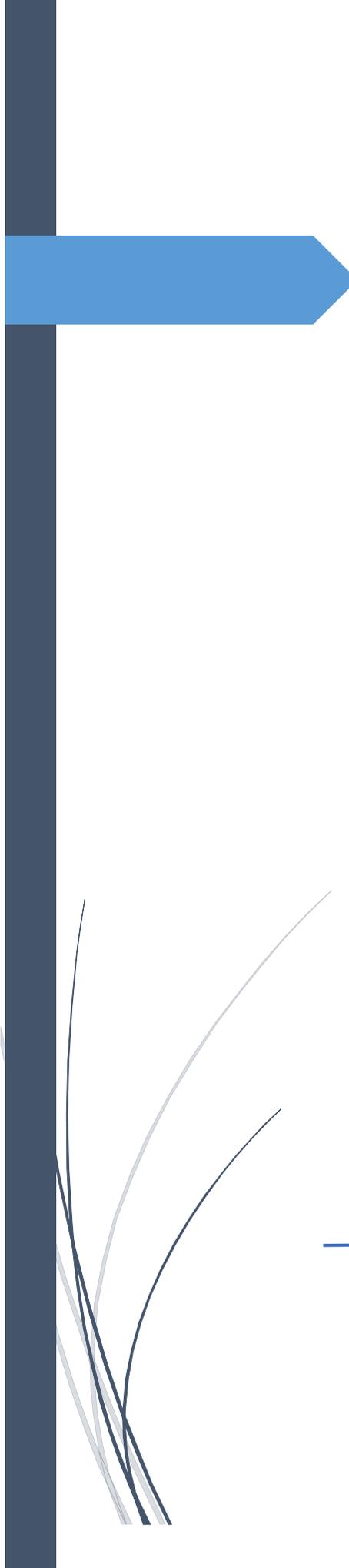
Seleccionar todo    Para los elementos que están marcados:  Editar  Copiar  Borrar  Exportar

Mostrar todo | Número de filas: 25 | Filtrar filas:  Ordenar según la clave: Ninguna

Fuente: Elaboración propia

Para el buen uso del Sistema se recomienda lo siguiente:

- Se recomienda al administrador principal del Sistema Web, asignar o aumentar ciertos privilegios, solo a usuarios capacitados para el manejo el mismo.
- Se recomienda a los usuarios el cambio periódico de sus contraseñas para la seguridad de sus cuentas y principalmente del sistema.
- Se recomienda realizar el mantenimiento preventivo y correctivo del sistema para su buen funcionamiento y prevención de fallas.



# CAPITULO V

---

## COSTOS Y BENEFICIO

*INGENIERIA DE SISTEMAS - UNIVERSIDAD PÚBLICA EL ALTO*

## 5. ANALISIS DE COSTO DE SOFTWARE

Existen distintos métodos para la estimación de costo de desarrollo de software, estos métodos no son otra cosa que establece una relación matemática entre el esfuerzo y el tiempo de desarrollo.

Para la evaluación de costos del sistema se emplea el modelo COCOMO (COConstructive COst MOdel).

### 5.1 Método De Estimación Cócono

La estimación de costos del sistema ha sido desarrollada bajo la KLDC (Kilo-líneas de código) como se detalla a continuación.

Para calcular el esfuerzo, necesitamos hallar la variable KLDC (kili-lineas de código)

Este proyecto se implementa 9868 líneas de código en el lenguaje PHP

Aplicando la Conversión se tiene

$$KLCD=(LCD)/1000$$

$$KLCD=9.868/1000$$

$$KLCD=9.868 \text{ KLC}$$

Es un modelo orgánico ya que el proyecto no supera las 50 kldc y es el más apropiado en este caso:

**Tabla N° 5. 1 Aplicación del modelo Intermedio**

PROYECTO SOFTWARE	A	b	c	d
<b>Orgánico</b>	2,4	1,05	2,5	0,38
<b>Semicopado</b>	3,0	1,12	2,5	0,35
<b>Empotrado</b>	2,8	1,20	2,5	0,32

Fuente: (COCOMO, 2013)

A continuación, presentamos las ecuaciones que nos permitirán calcular el costo total de software.

**Tabla Nº 5. 2 Ecuación de Modelo COCOMO**

Variable	Ecuación	Tipo/Unidad
<b>Esfuerzo requerido por el proyecto</b>	$E = a \times (KLDC)^b \times FAE$	Personas/mes
<b>Tiempo requerido por el proyecto</b>	$T = c \times (E)^n$	Meses
<b>Número de personas requeridas para el proyecto</b>	$NP = \frac{E}{T}$	Personas
<b>Costo total</b>	$CT = \text{Sueldo Mes} \times NP \times T$	Bs

**Fuente:** (Prentice-Hall, 1981)

Para hallar los valores de FAE, se utilizará la tabla de atributos multiplicadores

**Tabla Nº 5. 3 Calculo de Atributos FAE**

Atributos	Valor					
	Muy bajo	Bajo	Nominal	Alto	Muy alto	Extra alto
<b>Atributos de software</b>						
Fiabilidad	0,75	0,88	1,00	1,15	1,40	
Tamaño de Base de datos		0,94	1,00	1,08	1,16	
Complejidad	0,70	0,85	1,00	1,15	1,30	1,65
<b>Atributos de hardware</b>						
Restricciones de tiempo de ejecución			1,00	1,11	1,30	1,66
Restricciones de memoria virtual			1,00	1,06	1,21	1,56
Volatilidad de la máquina virtual		0,87	1,00	1,15	1,30	
Tiempo de respuesta		0,87	1,00	1,07	1,15	
<b>Atributos de personal</b>						
Capacidad de análisis	1,46	1,19	1,00	0,86	0,71	
Experiencia en la aplicación	1,29	1,13	1,00	0,91	0,82	
Calidad de los programadores	1,42	1,17	1,00	0,86	0,70	
Experiencia en la máquina virtual	1,21	1,10	1,00	0,90		
Experiencia en el lenguaje	1,14	1,07	1,00	0,95		

Atributos	Valor					
	Muy bajo	Bajo	Nominal	Alto	Muy alto	Extra alto
Atributos del proyecto						
Técnicas actualizadas de programación	1,24	1,10	1,00	0,91	0,82	
Utilización de herramientas de software	1,24	1,10	1,00	0,91	0,83	
Restricciones de tiempo de desarrollo	1,22	1,08	1,00	1,04	1,10	
Total			0,87			

**Fuente:** (Elaboración Propia)

Por lo tanto, nuestro factor de ajuste será.  
FAE=0,87

Aplicando y reemplazando valores a la fórmula de esfuerzo, se tiene:

$$E = a * KLCD^b * FAE \text{ (Personas/Mes)}$$

$$E = 2,4 * 9.868^{1,05} * 0,87$$

$$E = 23.1 \text{ (Personas Mes)}$$

### Calculo del Tiempo

$$T = c * Esfuerzo^d \text{ (Meses)}$$

$$T = 2,5 * 23.1^{0,32}$$

$$T = 6.82 \text{ (Meses)}$$

### Calculo de Productividad

$$PR = \frac{LCD}{Esfuerzo} \text{ (Meses)}$$

$$PR = \frac{9.868}{23.1} \text{ (Meses)}$$

$$PR = 428.5 \text{ (LCD/Personas Mes)}$$

### **Calculo del personal requerido**

$$P = \frac{E}{T} \text{ (Personas)}$$

$$P = \frac{23.1}{6.82} \text{ (Personas)}$$

P=3.39 equivalente a 3 personas

### **Costo Total del Proyecto**

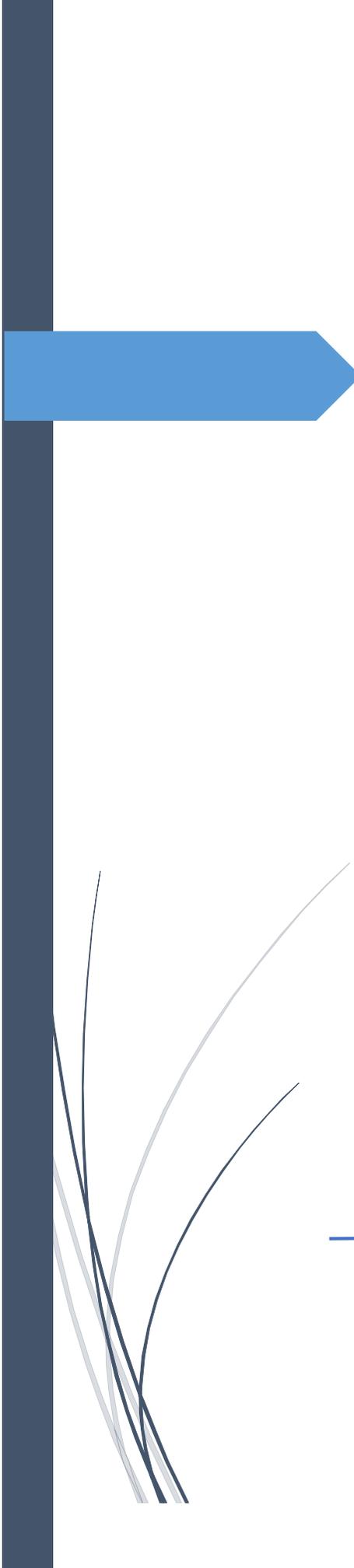
(Coste Mes) = P\* Salario medio entre los programadores y analistas

**Costo Persona Mes Bs. 2,800**

Costo Mes=3\*2800= Bs. 8,400

**Costo Total=8.400\*3= Bs. 25,200**

En resumen, se requiere 3 personas estimado un trabajo de 6 meses y un costo total de Bs. 25,200.



# CAPITULO VI

---

## CONCLUSIONES Y RECOMENDACIONES

*INGENIERIA DE SISTEMAS - UNIVERSIDAD PÚBLICA EL ALTO*

## **6. CONCLUSIONES Y RECOMENDACIONES**

### **6.1 CONCLUSIONES**

A la finalización del presente proyecto, entre las conclusiones que se puede resumir son:

1. Se logró desarrollar e implementar el software para el proceso de adquisición de materiales y control de almacenes, el mismo permite realizar un control adecuado de la distribución de materiales desde el registro de solicitud y entrega de materiales, mediante formularios de ingreso y salida que respaldan el movimiento.
2. Con la implementación del software se obtuvieron importantes beneficios como la generación de formularios con información importante para la administración de la unidad correspondiente y el conjunto de las áreas de la Institución como ser: reducción notable del tiempo que se emplea en realizar una solicitud y posterior entrega, la elaboración y emisión de reportes por parte del área de almacén mediante un sistema oportuno y confiable.
3. A partir del diseño y desarrollo de un sistema para el control de inventario en almacén, el movimiento de ingresos y salidas de materiales está reflejado en el reporte de kardex.
4. La implementación del software agilizará todo el proceso relacionado al área de almacenes, así como la toma de decisiones por parte de la dirección de la Institución generando reportes ágiles, dinámicos y actualizados.
5. Se logró el diseño de un módulo de centralización de información de los proveedores, hecho que genera un beneficio en cuanto al tiempo y la toma de decisiones.
6. Se aplicó la metodología ISO 9126 para la evaluación de la calidad del sistema, llegando a obtener resultados a satisfacción del usuario en la interacción con el sistema. Por tanto, el sistema desarrollado cumple con los

requisitos identificados, la interfaz de usuario es adecuada para el manejo y administración del almacén, cumpliendo su objetivo general, mejorando el control sobre la distribución y movimiento de materiales desde el ingreso hasta la entrega del mismo al personal de la Institución.

## **6.2 RECOMENDACIONES**

El desarrollo e implementación del software y de acuerdo a lo descrito en las conclusiones, se realiza las siguientes recomendaciones:

- Elaborar el reglamento administrativo sobre el manejo y administración del proceso de adquisición de bienes e insumos que contemple los flujos y procedimientos para la utilización del sistema y realizar la difusión respectiva a los funcionarios autorizados de la Institución que lleguen a aplicar el software.
- Realizar un mantenimiento periódico del software, servidores y redes implementados en el presente proyecto.
- Se sugiere para posteriores versiones del sistema realizar la automatización del código de material, ya que actualmente el sistema lo realiza manualmente a requerimiento del jefe administrador de almacén.

## BIBLIOGRAFÍA

### Referencias Bibliográficas:

(s.f.).

Alegsa, L. (2018).

Alegsa, L. (2018).

Aranguren, M. (S.F).

Arias, M. (2006).

Arias, M. (2006).

*COCOMO*. (2013).

*Estimacion de costos de desarrollo de software* . (2014).

Gutierrez Vargas, G. (2015). *Sistema de control de ventas e inventarios para almacenes de aluminios utilizando dispositivos moviles*. La Paz.

Hernanadez Trasobares, A. (2000).

HERNANDEZ TRASOBARES, A. (2000).

Maidana. (2014). *Seguridad Informatica*.

Maidana. (2014). *Seguridad Informatica*.

Mamani, J. J. (2014). *Sistema de informacion seguimiento academico para el control de registro de notas*. la paz: tesis.

Porto, J. P. (2012). Definicion de php.

Prentice-Hall, S. (1981).

Pressman, R. S. (2010). *INGENIERIA DE SOFTWARE Y* . Mexico.

R., Pressman. (2006). *Ingenieria de Software SEPTIMA EDICION*.

➤ **Disponible en:**

- (s.f.).
- A. Goñi, J. I. (1 de 10 de 2018). *Ingenieria del Software II*. Recuperado el 03 de 08 de 2020, de Verificacion del Software:  
[https://ocw.ehu.es/pluginfile.php/16680/mod\\_resource/content/6/Pruebas-OCW.pdf](https://ocw.ehu.es/pluginfile.php/16680/mod_resource/content/6/Pruebas-OCW.pdf)
- Adriana Gómez, M. d. (18 de febrero de 2009). *UN MODELO DE ESTIMACION DE PROYECTOS DE SOFTWARE*. Obtenido de  
<https://blogadmi1.files.wordpress.com/2010/11/cocom011full.pdf>
- AIGP2 Calidad Informatica*. (2011). Obtenido de  
<https://www.um.es/docencia/barzana/IAGP/IAGP2-Calidad-informatica.html>
- Alegsa, L. (27 de 08 de 2018). *alegsa.com.ar*. Obtenido de  
<http://www.alegsa.com.ar/Dic/sistema.php>
- Alvarado, V. V. (15 de 01 de 2012). *Técnicas efectivas para la toma de requerimientos*. Recuperado el 05 de 08 de 2020, de <https://www.northware.mx/>:  
[https://www.northware.mx/2012/01/15/tecnicas\\_efectivas\\_toma\\_requerimientos/](https://www.northware.mx/2012/01/15/tecnicas_efectivas_toma_requerimientos/)
- Alvarez, M. A. (1 de enero de 2001). *desarrolloweb.com*. Obtenido de [desarrolloweb.com](http://www.desarrolloweb.com):  
<http://www.desarrolloweb.com/articulos/332.php>
- ARSYS. (2019). QUE NOS APORTA MATERIALIZE CSS EN EL DESARROLLO? ARSYS. Obtenido de <https://www.arsys.es/blog/programacion/nos-puede-aportar-materialize-css/>
- Calidad de Software Metricas y Fiabilidad de Aplicaciones*. (s.f.). Obtenido de  
[https://www.aprenderaprogramar.com/index.php?option=com\\_content&view=article&id=198:calidad-del-software-metricas-y-fiabilidad-de-aplicaciones-1a-parte-dv00103a&catid=45&Itemid=164](https://www.aprenderaprogramar.com/index.php?option=com_content&view=article&id=198:calidad-del-software-metricas-y-fiabilidad-de-aplicaciones-1a-parte-dv00103a&catid=45&Itemid=164)
- Callisaya Apaza, W. (2017). SOFTWARE DE CONTROL DE INVENTARIOS. Obtenido de <http://rupmetodologia.blogspot.com/>
- Copyright. (2012). *E.S.F.M.T.H.E.A.* . Obtenido de El Alto - Distrito 12 - Zona Alto Chijini Av. Buenos Aires S/N: <http://www.esfmthea.edu.bo>
- de Software, I. (2011). *Ingenieria de Software*. Obtenido de  
<https://ingenieriasoft.webcindario.com/control-de-calidad-del-software/tecnicas-de-prueba-del-software/prueba-de-caja-negra.html>
- Ecured*. (2012). Obtenido de <https://www.ecured.cu/CSS3>
- Estimacion de costos de desarrollo de software* . (2014). Obtenido de  
<https://www.gestiopolis.com/estimacion-de-costos-de-desarrollo-de-software/>
- Flores, D. (13 de mayo de 2009). *Monografias.com S.A*. Obtenido de  
<http://www.monografias.com/trabajos70/disenio-sistema-informacion-proceso-inscripcion/disenio-sistema-informacion-proceso-inscripcion.shtml>
- Gutierrez Vargas, G. (2015). *Sistema de control de ventas e inventarios para almacenes de aluminio utilizando dispositivos moviles*. La Paz.
- Herrera, L. F. (20 de MAYO de 2014). *ISO 9126*. Recuperado el 02 de junio de 2016, de <http://es.slideshare.net/luisfarinango5682/norma-iso-9126-34931370>
- Juan., R. (25 de MARZO de 2005). *Definición de JavaScript*. Recuperado el 22 de ABRIL de 2016, de <http://www.gestiopolis.com/definicion-javascript/>

*Las Normas Iso 9000 (PDF)*. (s.f.). Obtenido de [https://www.ucongreso.edu.ar/grado/carreras/lsi/2006/ele\\_calsof/MaterialComplemen-ISO9000%20A.pdf](https://www.ucongreso.edu.ar/grado/carreras/lsi/2006/ele_calsof/MaterialComplemen-ISO9000%20A.pdf)

Lopez, D. (02 de agosto de 2006). *PHP4 y PHP5: ¿Cuál elegir? ¿Migrar o no Migrar? El advenimiento de PHP6*. Recuperado el 18 de mayo de 2016, de <http://www.maestrosdelweb.com/php4y5/>

Lopez, G. A. (2005). *pdf*. Obtenido de pdf: <http://javeriana.edu.co/biblos/tesis/ingenieria/Tesis212.pdf>  
<http://repositorio.uis.edu.co/jspui/bitstream/123456789/2446/2/112708.pdf>

*Modelo de estimacion de proyectos de software*. (s.f.). Obtenido de <https://es.slideshare.net/elgalin/modelos-de-estimacion-de-software>

*Modeloe de estimacion*. (2012). Obtenido de <https://unpocodejava.com/2012/02/07/modelos-de-estimacion-un-poco-sobre-cocomo-ii/>

Montesinos, R. (24 de JULIO de 2014). *BLOG I2B*. Obtenido de I2B: <http://www.i2btech.com/blog-i2b/?pag=3>

*norma evaluacion iso 9126*. (2013). Obtenido de <http://actividadreconocimiento-301569-8.blogspot.com/2013/03/norma-de-evaluacion-isoiec-9126.html>

*PHP5 Orientado a Objetos*. (07 de noviembre de 2012). Recuperado el 18 de mayo de 2016, de <https://www.codejobs.biz/es/blog/2012/11/07/php5-orientado-a-objetos>

*Proyecto fin de master (pdf)*. (2016). Obtenido de <https://repositorio.comillas.edu/xmlui/bitstream/handle/11531/9389/TFM000345.pdf?sequence=1>

Quiroga, A. (24 de marzo de 2015). *BLOG unad*. Obtenido de <http://proyectogradoingenieriasistemas.blogspot.com/2015/03/metodologia-uwe-uml-uml-based-web.html>

Rainer, R. K. (19 de junio de 2009). *wikipedia*. Obtenido de [https://es.wikipedia.org/wiki/Sistema\\_de\\_informaci%C3%B3n](https://es.wikipedia.org/wiki/Sistema_de_informaci%C3%B3n)

Robles, V. (2017). Obtenido de <https://victorroblesweb.es/2017/08/05/que-es-angular-y-para-que-sirve/>

Soliz, A. (2012). *Tropiccode.net*. Obtenido de <https://tropiccode.net/seguridad/que-es-y-como-funciona-sha1/>

spcgroup. (2014). Obtenido de <https://spcgroup.com.mx/que-es-un-almacen/>

Valdés, D. P. (03 de JULIO de 2007). *¿Qué es Javascript?* Recuperado el 18 de MAYO de 2016, de <http://www.maestrosdelweb.com/que-es-javascript/>

villaroel. (1999). *sistema de informacion del seguimiento academico en el pre grado de la uto*. la paz: proyecto.

*wiki*. (7 de mayo de 2015). Obtenido de <https://maquetando.com/css/materializecss-el-framework-basado-en-material-design-de-google/>

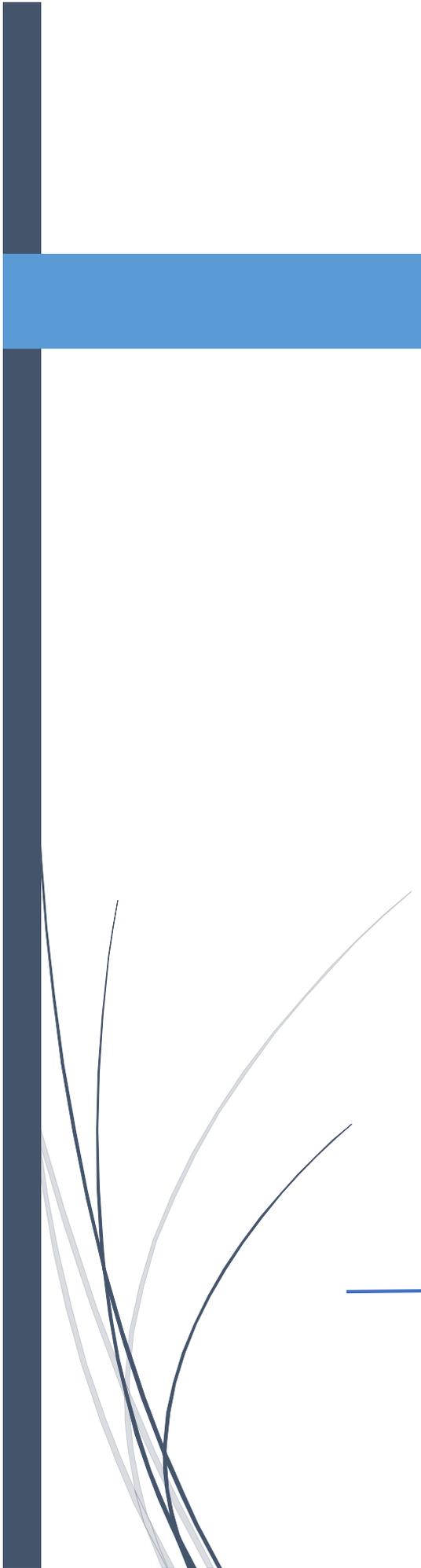
*wiki*. (6 de mayo de 2016). Obtenido de [https://es.wikipedia.org/wiki/Twitter\\_Bootstrap](https://es.wikipedia.org/wiki/Twitter_Bootstrap)

*wikipedia*. (12 de junio de 2012). Obtenido de <https://es.wikipedia.org/wiki/MySQL>

*wikipedia*. (2 de mayo de 2016). Obtenido de <https://es.wikipedia.org/wiki/XAMPP>

*wikipedia*. (20 de mayo de 2016). Obtenido de [https://es.wikipedia.org/wiki/Hoja\\_de\\_estilos\\_en\\_cascada](https://es.wikipedia.org/wiki/Hoja_de_estilos_en_cascada)

*wikipedia*. (8 de marzo de 2016). Obtenido de [https://es.wikipedia.org/wiki/Sublime\\_Text](https://es.wikipedia.org/wiki/Sublime_Text)



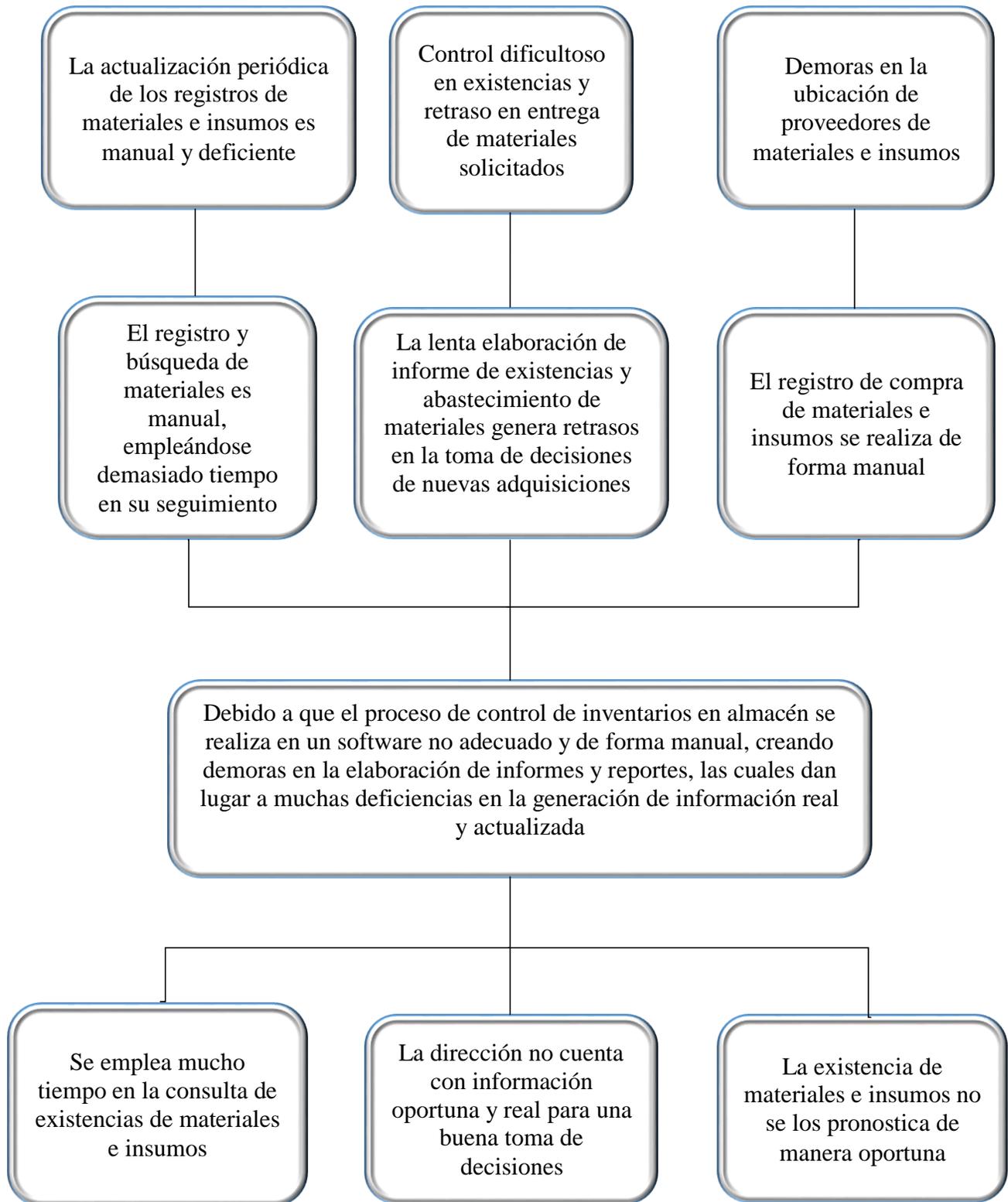
# ANEXOS

---

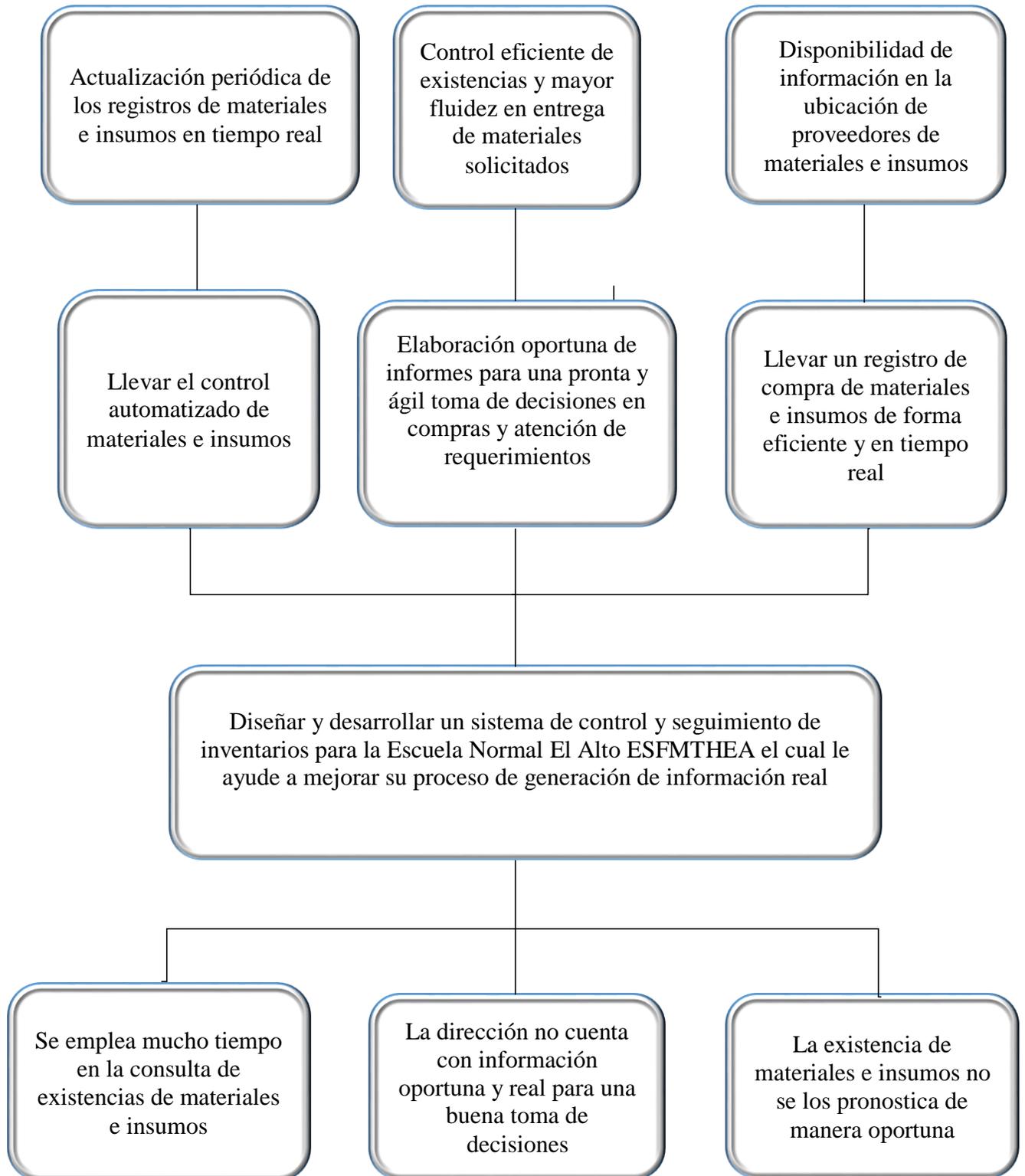
*INGENIERIA DE SISTEMAS - UNIVERSIDAD PÚBLICA EL ALTO*

## ANEXOS

### Anexo A ARBOL DE PROBLEMAS



## Anexo B ARBOL DE OBJETIVOS



# Entrevista

- 1) ¿Cómo se maneja la información registrada en la empresa?
  
- 2) ¿Cómo se realiza el registro de nuevos asociados a la empresa (choferes y camiones) y quienes se encargan?
  
- 3) ¿Cómo se realiza la coordinación de con el cliente?
  
- 4) ¿Cómo se procede a la búsqueda de los datos de los servicios?
  
- 5) ¿De qué manera se almacenan los datos de los datos de la empresa y en dónde?
  
- 6) ¿Qué problemas se tiene cuando un documento de la empresa se caduca?

## Anexo D CUESTIONARIO

# Cuestionario

1) El proceso de registro de funcionario:

- a) Malo      b) Regular      c) Bueno      d) Excelente

2) Las herramientas con las que se realiza registro de usuario es:

- a) Malo      b) Regular      c) Bueno      d) Excelente

3) Como califica la coordinación de los servicios y la disponibilidad de materiales en almacén:

- a) Malo      b) Regular      c) Bueno      d) Excelente

4) Las búsquedas y consultas de la información de los registros de proveedores, control de stock, de manera rápida y oportuna:

- a) Si      b) No

5) Contar con un software a medida que trabaje de acuerdo a la medida de la institución seria de utilidad:

- c) Si      d) No

## **Anexo E MODELO DE CUESTIONARIO UTILIZADO PARA LA MEDICIÓN DE CALIDAD**

Responda la siguiente pregunta con las características: SI NO

### **EJE DE OPERACIÓN DEL PRODUCTO**

#### **FACTOR DE CORRECCIÓN**

1) ¿El sistema cumple con las expectativas de los requerimientos planteados?

**SI**

**NO**

2) ¿En los atributos del sistema se implementa completamente las funciones requeridas y planteadas?

**SI**

**NO**

3) ¿Es posible solucionar errores que se cometan dentro del sistema?

**SI**

**NO**

#### **FACTOR DE FIABILIDAD**

1) ¿Existe precisión en las funciones y/o cálculos que desempeña el sistema?

**SI**

**NO**

2) ¿Existe tolerancia frente a fallos (que el sistema siga en funcionamiento aun en fallos de hardware, software, etc.)?

**SI**

**NO**

3) ¿El sistema tiene la exactitud requerida al momento de realizar los cálculos o funciones?

**SI**

**NO**

### **FACTOR DE EFICIENCIA**

1) ¿La ejecución del sistema se realiza de manera eficiente (tiempo de procesamiento)?

**SI**

**NO**

2) ¿Los datos almacenados son utilizados de manera eficiente para el funcionamiento del sistema?

**SI**

**NO**

3) ¿Los datos almacenados son utilizados de manera eficiente para el funcionamiento del sistema?

**SI**

**NO**

### **FACTOR DE INTEGRIDAD**

1) ¿Se cuenta con un control seguro de accesos al sistema?

**SI**

**NO**

2) ¿Se tiene cierto nivel de acceso a los datos manejados por el sistema?

**SI**

**NO**

3) ¿Existe mecanismos de seguridad para el acceso y manejo de los datos dentro del sistema?

**SI**

**NO**

### **FACTOR DE FACILIDAD DE USO**

1) ¿Es sencillo el manejo y la interacción con el Sistema?

**SI**

**NO**

2) ¿Se cuenta con ayuda necesaria para el manejo adecuado del sistema?

**SI**

**NO**

3) ¿La información manejada como de entrada y de salida en el sistema son de fácil entendimiento?

**SI**

**NO**

### **EJE DE REVISIÓN DEL PRODUCTO**

#### **FACTOR DE FACILIDAD DE MANTENIMIENTO**

1) ¿Se puede realizar un mantenimiento preventivo y correctivo si el sistema presenta fallos?

**SI**

**NO**

2) ¿Está planeado realizar algún tipo de mantenimiento del sistema?

**SI**

**NO**

## **FACTOR DE FACILIDAD DE PRUEBAS**

1) ¿Se puede evaluar de manera fácil los resultados esperados por el sistema?

**SI**

**NO**

2) ¿Es posible verificar de manera fácil la emisión de información del sistema?

**SI**

**NO**

3) ¿Se puede validar los requerimientos planificados para el sistema?

**SI**

**NO**

## **FACTOR DE FLEXIBILIDAD**

1) ¿El sistema es flexible a los cambios que se requiera?

**SI**

**NO**

2) ¿Es posible la adición o expansión de nuevas funciones dentro del sistema?

**SI**

**NO**

3) ¿Se requirió algún cambio para el funcionamiento adecuado del sistema?

**SI**

**NO**

## **EJE DE TRANSICIÓN DEL PRODUCTO**

### **FACTOR DE PORTABILIDAD**

1) ¿El sistema es portables?

**SI**

**NO**

2) ¿El sistema puede ejecutarse en distintos sistemas operativos?

**SI**

**NO**

3) ¿El sistema puede funcionar con otras características de hardware y software?

**SI**

**NO**

4) ¿El sistema no depende del sistema operativo en el cual se ejecuta?

**SI**

**NO**

### **FACTOR DE REUSABILIDAD**

1) ¿El sistema puede ejecutarse en cualquier tipo de software?

**SI**

**NO**

2) ¿Son necesarios ciertos requisitos en el equipo para el funcionamiento del sistema?

**SI**

**NO**

### **FACTOR DE FACILIDAD DE INTEROPERACIÓN**

1) ¿El sistema está diseñado para poder compartir información requerida?

**SI**

**NO**

2) ¿La información del sistema se encuentra centralizada?

**SI**

**NO**



# MANUAL DE USUARIOS

---

## “SISTEMA DE INFORMACIÓN PARA EL CONTROL DE INVENTARIO EN ALMACÉN”

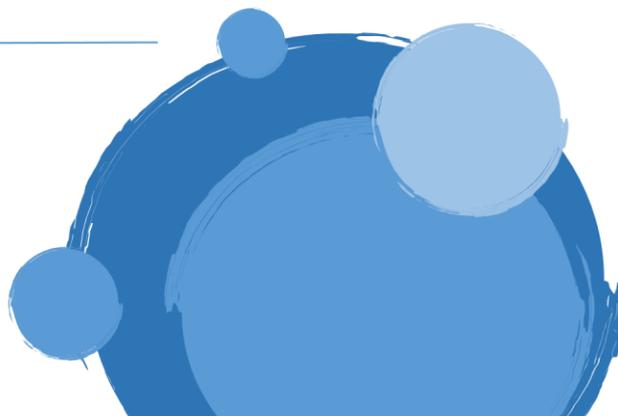
**E.S.F.M.T.H.E.A.**

“Escuela Superior de Formación de Maestros Tecnológico Humanístico El Alto”

---

**Nolia Chamairo Espejo**

**LA PAZ – BOLIVIA**



# INDICE

PAG.

MANUAL DEL USUARIO.....	i
1. INTRODUCCION.....	1
2. OBJETIVO .....	1
3. REQUERIMIENTOS DEL SISTEMA .....	1
4. TIPOS DE USUARIO .....	1
4.1 INGRESO USUARIO ADMINISTRADOR .....	3
4.1.1 MÓDULO ÁREA .....	3
4.1.2 MÓDULO CARGOS .....	5
4.1.3 MÓDULO FUNCIONARIOS.....	6
4.1.4 MÓDULO USUARIO.....	8
4.1.5 MÓDULO PROVEEDOR .....	10
4.1.6 MÓDULO CONTROL DE MATERIALES .....	12
4.1.7 MÓDULO ENTRADA Y SALIDA DE MATERIALES.....	17
4.1.8 MÓDULO REPORTES.....	23
4.2 INGRESO USUARIO FUNCIONARIO.....	25
4.2.1 MÓDULO MATERIALES .....	26
4.2.2 MÓDULO SOLICITUD DE MATERIALES .....	27
4.2.3 MÓDULO SOLICITUDES .....	28

## INDICE DE FIGURAS

	PAG.
<i>FIGURA N° 1. 2 INTERFAZ INGRESO ADMINISTRADOR.....</i>	<i>3</i>
<i>FIGURA N° 1. 3 INTERFAZ DE MÓDULO ÁREA .....</i>	<i>4</i>
<i>FIGURA N° 1. 4 INTERFAZ MÓDULO ÁREA – BOTÓN NUEVA ÁREA .....</i>	<i>4</i>
<i>FIGURA N° 1. 5 INTERFAZ MÓDULO CARGO .....</i>	<i>5</i>
<i>FIGURA N° 1. 6 INTERFAZ CARGO – BOTÓN NUEVO CARGO.....</i>	<i>6</i>
<i>FIGURA N° 1. 9 MÓDULO DE USUARIO .....</i>	<i>8</i>
<i>FIGURA N° 1. 10 MÓDULO USUARIO – BOTÓN NUEVO USUARIO.....</i>	<i>9</i>
<i>FIGURA N° 1. 11 MÓDULO DE PROVEEDOR.....</i>	<i>10</i>
<i>FIGURA N° 1. 13 MÓDULO DE CONTROL DE MATERIALES - CATEGORÍA .....</i>	<i>12</i>
<i>FIGURA N° 1. 14 MÓDULO PROVEEDOR– BOTÓN NUEVO PROVEEDOR.....</i>	<i>13</i>
<i>FIGURA N° 1. 15 MÓDULO CONTROL DE MATERIALES – REGISTRO DE MATERIALES.....</i>	<i>14</i>
<i>FIGURA N° 1. 16 MÓDULO REGISTRO DE MATERIAL – NUEVO MATERIAL.....</i>	<i>15</i>
<i>FIGURA N° 1. 17 INTERFAZ DE SUB CANTIDAD AUTORIZADA.....</i>	<i>16</i>
<i>FIGURA N° 1. 18 INTERFAZ SUB MÓDULO STOCK EN ALMACÉN .....</i>	<i>17</i>
<i>FIGURA N° 1. 19 INTERFAZ SUB MÓDULO SOLICITUD DE MATERIAL .....</i>	<i>18</i>
<i>FIGURA N° 1. 20 INTERFAZ SUB MÓDULO SOLICITAR POR FUNCIONARIO.....</i>	<i>19</i>
<i>FIGURA N° 1. 23 INTERFAZ DE SUB MÓDULO REGISTRO DE SALIDA .....</i>	<i>22</i>
<i>FIGURA N° 1. 24 INTERFAZ SUB MÓDULO REPORTE GENERAL.....</i>	<i>23</i>
<i>FIGURA N° 1. 25 REPORTE PDF.....</i>	<i>24</i>
<i>FIGURA N° 1. 27 INTERFAZ REPORTE POR FUNCIONARIO – PDF FUNCIONARIO.....</i>	<i>25</i>
<i>FIGURA N° 1. 28 INTERFAZ USUARIO FUNCIONARIO .....</i>	<i>25</i>
<i>FIGURA N° 1. 29 INTERFAZ MATERIALES.....</i>	<i>26</i>
<i>FIGURA N° 1. 32 MÓDULO SOLICITUD .....</i>	<i>28</i>

## 1. INTRODUCCION

El Sistema de Información para el Control de Inventarios en Almacén, es un programa creado para registrar, analizar, actualizar los datos relacionados con el movimiento de materiales en el Almacén de la Institución E.S.F.M.T.H.A.

En el presente manual se explicará el uso y funcionamiento del Sistema de Información para el control de inventario en el área de almacén.

El administrador del sistema cuenta con todos los privilegios y permisos, es decir que el administrador será el único que puede administrar los usuarios que tendrá acceso al sistema.

## 2. OBJETIVO

- ✚ Permitir al personal administrativo de la institución el manejo adecuado de los procesos de información y adquisición de materiales en área de almacén.
- ✚ Guiar a los usuarios mediante imágenes y capturas de pantalla del sistema para el uso correcto.

## 3. REQUERIMIENTOS DEL SISTEMA

### ✚ Requerimientos de Hardware

- ✓ Conexión a internet
- ✓ Computadora personal

### ✚ Requerimientos de Software

- ✓ Sistema Operativo Windows
- ✓ Mozilla Firefox o Navegador Chrome

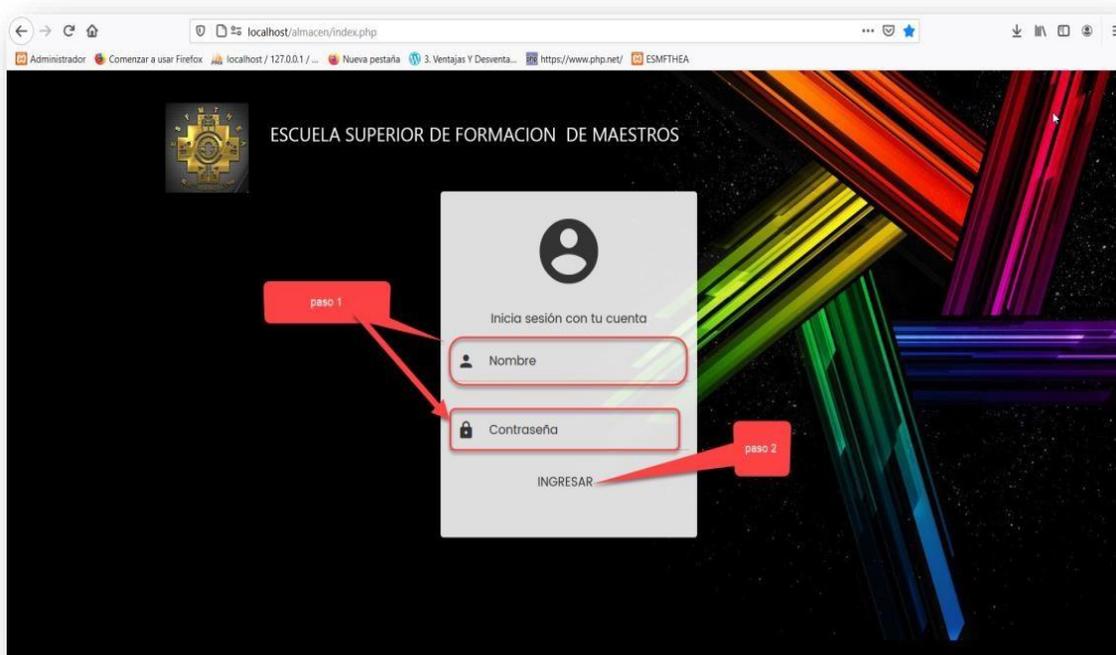
## 4. TIPOS DE USUARIO

El sistema contiene dos tipos de usuario, por tanto, el administrador General puede asignar los privilegios o rol de usuarios de acuerdo al Área del personal

- ✓ Administrador (Jefe de Almacén)
- ✓ Encargado (Funcionarios) según organigrama

Para utilizar el sistema se debe acceder a través de los navegadores Mozilla y Chrome, para acceder al sistema deberá ingresar su nombre de usuario y contraseña como se ve en la figura luego ingresar al sistema.

**Figura N° 1. 1 Login: Iniciar Sesión**



Fuente: Elaboración propia

- Paso1: Ingrese los datos de Usuario y Contraseña.
- Paso 2: Digite en Ingresar para acceder al sistema.

## 4.1 INGRESO USUARIO ADMINISTRADOR

**Figura N° 1. 2 Interfaz Ingreso Administrador**



**Fuente:** Elaboración propia

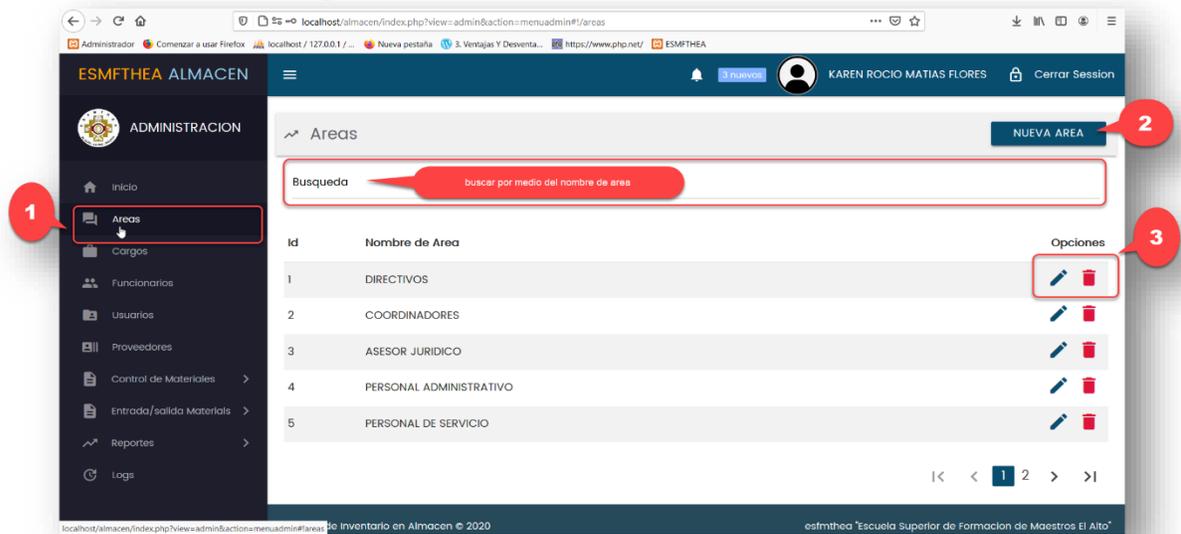
- Se mostrará el menú con los respectivos módulos
- Notificaciones solicitadas
- Datos del usuario administrador, pulsar en nombre usuario para cambiar la contraseña.
- Pulsar en cerrar sesión para cerrar la pantalla administrador

Ingrese al sistema **Administrador** podrá trabajar en la interfaz que se muestra. Asimismo, tendrá acceso a todos los menús y sub menús, como administrador es el único que tiene acceso a agregar, cambiar, los privilegios, y dar de baja a los usuarios en el sistema.

### 4.1.1 Módulo Área

En el siguiente menú se podrá realizar el registro de las áreas o dependencias que conforma la institución.

**Figura N° 1. 3 Interfaz de Módulo Área**



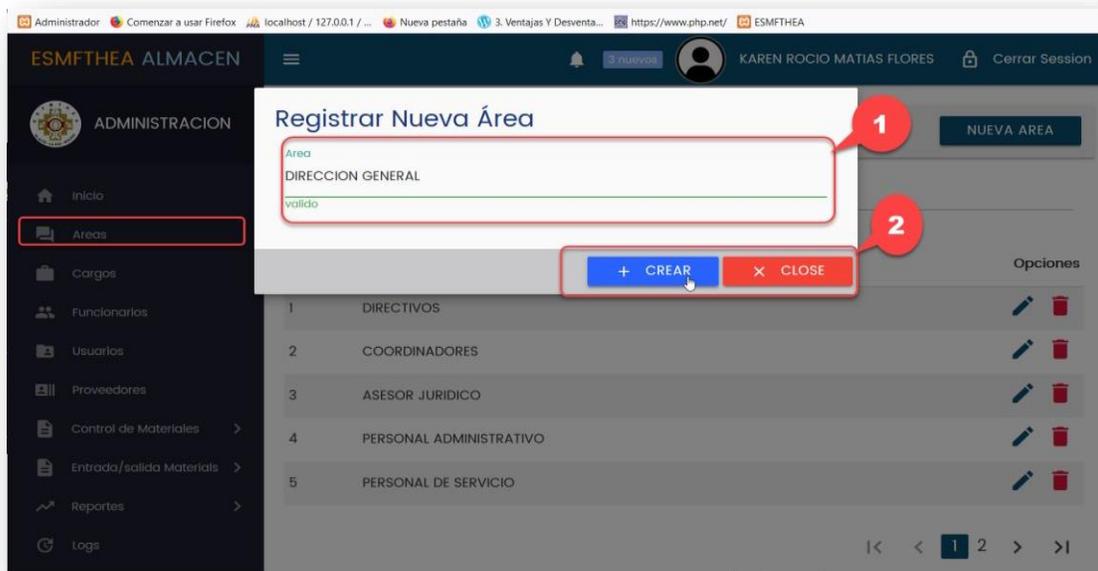
**Fuente:** Elaboración propia

1. Pulsar en el menú Área: el mismo que nos mostrará la ventana del módulo **área**, el cual visualizará una lista de registros de área, asimismo podrá realizar búsqueda llamando el nombre de área.
2. Se muestra el botón **Nueva Área**, pulsar para realizar nuevo registro.
3. Se tiene las siguientes acciones donde se podrán modificar o eliminar.

#### 4.1.1.1 Módulo Área - Botón Nueva Área

Después de pulsar en el menú **Nueva Área** nos envía a la siguiente ventana:

**Figura N° 1. 4 Interfaz Módulo Área – Botón Nueva Área**



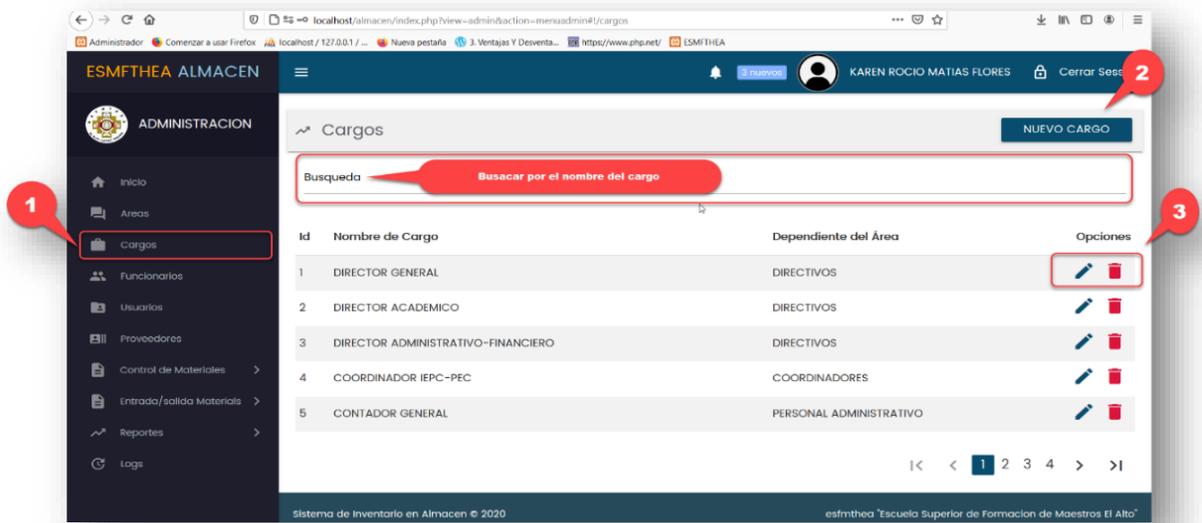
**Fuente:** Elaboración propia

1. Llenar el campo para el registro de área
2. Tenemos el botón **crear** que ayudará a registrar la nueva área, y el botón **close** es para cancelar el registro.

#### 4.1.2 Módulo Cargos

En el siguiente menú se podrá realizar el registro de cargo que esta designado a los funcionarios.

**Figura N° 1. 5 Interfaz Módulo Cargo**



Fuente: Elaboración propia

1. Pulsar en el menú Cargo: el mismo que nos mostrará la ventana del módulo **Cargo**, el cual visualizará una lista de registros de cargos, asimismo podrá realizar búsqueda llamando el nombre de cargos.
2. Se muestra el botón **Nuevo Cargo**, pulsar para realizar un nuevo registro.
3. Se tiene las siguientes acciones donde se podrán modificar datos y eliminar.

##### 4.1.2.1 Módulo Cargo - Botón Nueva Cargo

Después de pulsar en el menú Nueva Cargo nos envía a la siguiente ventana

**Figura N° 1. 6 Interfaz Cargo – Botón Nuevo Cargo**



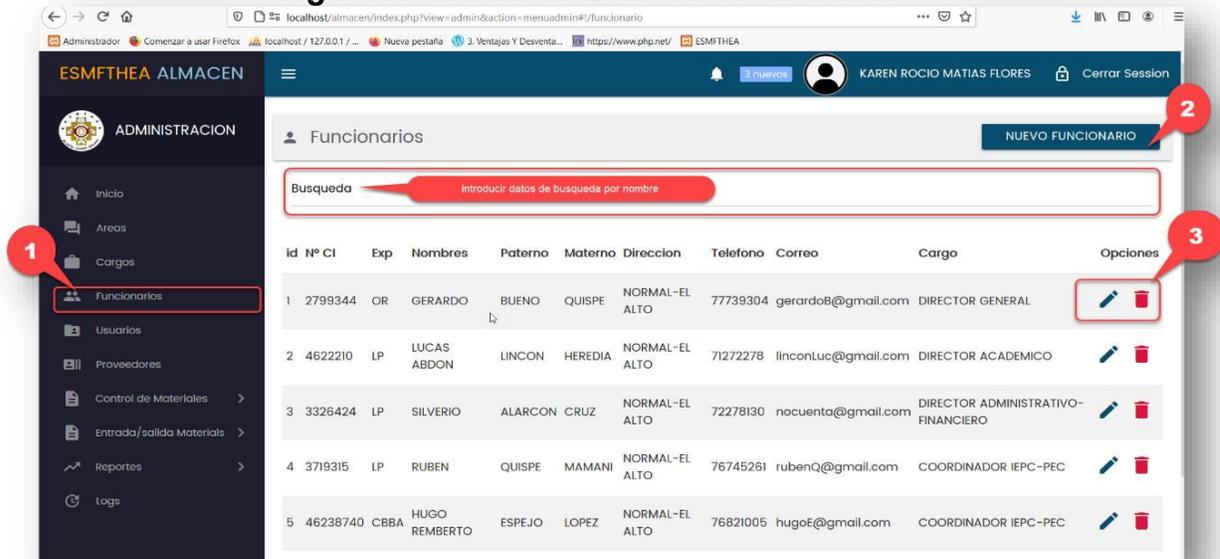
Fuente: Elaboración propia

1. Llenar el campo para el registro de insertar un nuevo cargo
2. Seleccionar el tipo de área que pertenece el cargo registrado
3. Tenemos el botón **crear** que ayudara a registrar el nuevo cargo, y el botón **close** para cancelar el registro.

#### 4.1.3 Módulo Funcionarios

En el siguiente menú se podrá visualizar el Módulo de Registro de Funcionarios.

**Figura N° 1. 7 Interfaz Módulo Funcionario**



Fuente: Elaboración propia

1. Pulsar en el menú Funcionario: el mismo que nos mostrará la ventana del módulo **funcionario**, en el cual se visualizará una lista de registros de funcionarios, asimismo podrá realizar búsqueda llamando el nombre de funcionario.
2. Se muestra el botón Nuevo Funcionario, pulsar para realizar un nuevo registro.
3. Se tiene las siguientes acciones donde se podrás modificar datos y eliminar.

#### 4.1.3.1 Módulo Funcionario - Botón Nuevo Funcionario

Después de pulsar en el menú Nuevo Funcionario nos envía a la siguiente ventana

**Figura N° 1. 8 Interfaz Funcionario – Botón Nuevo Funcionario**

ID	Cedula	Localidad	Nombre	Apellido Paterno	Apellido Materno	Telefono	Correo	Cargo	
4	3719315	LP	RUBEN	QUISPE	MAMANI	ALTO	76745261	rubenq@gmail.com	COORDINADOR IEPC-PEC
5	46238740	CBBA	HUGO REMBERTO	ESPEJO	LOPEZ	NORMAL-EL ALTO	76821005	hugoE@gmail.com	COORDINADOR IEPC-PEC

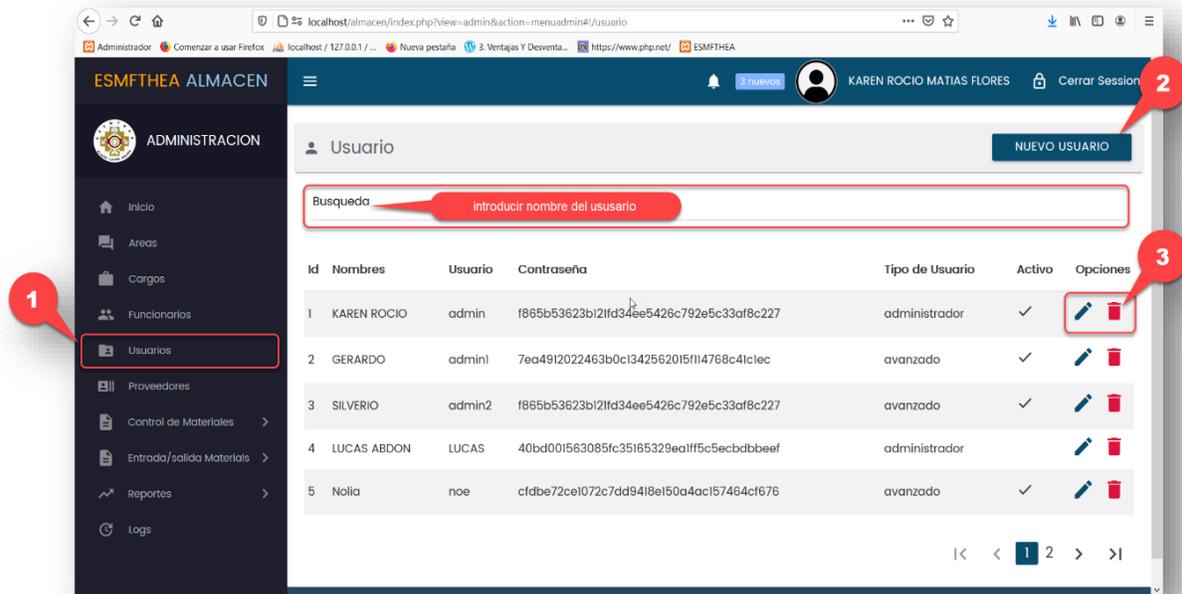
Fuente: Elaboración propia

1. Llenar los campos solicitados para el registro e insertar un nuevo funcionario.
2. Tenemos el botón **crear** que ayudara a registrar el nuevo cargo, y el botón **close** para cancelar el registro.

#### 4.1.4 Módulo Usuario

En el siguiente menú se podrá visualizar el Módulo del Registro de Usuario.

**Figura N° 1. 9 Módulo de Usuario**



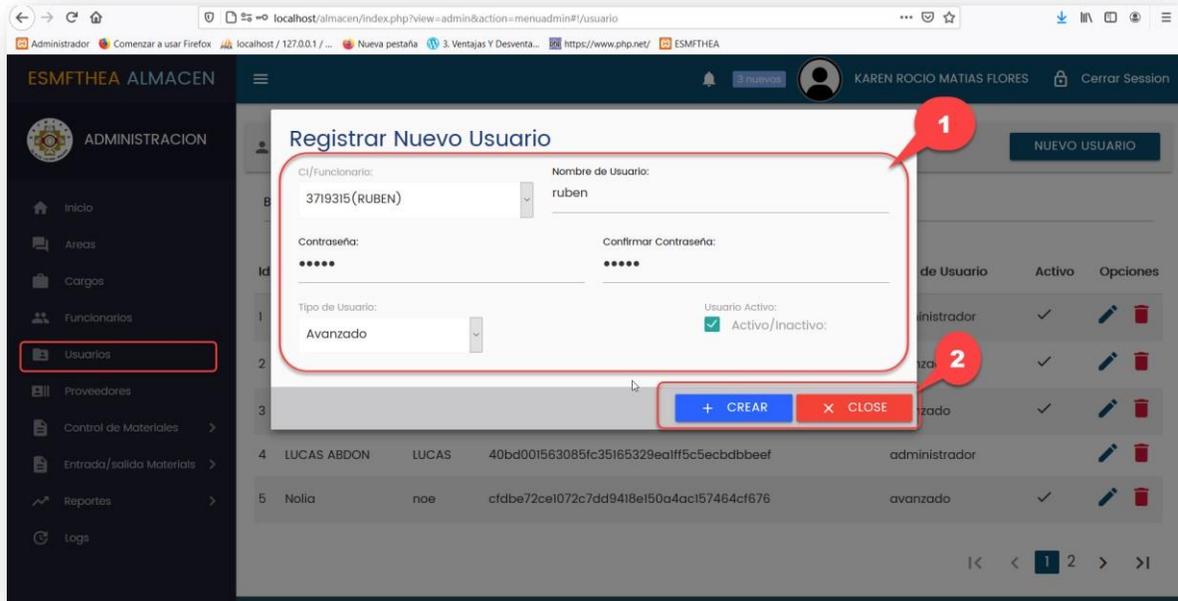
Fuente: Elaboración propia

1. Pulsar en el menú usuario: el mismo que nos mostrará la ventana del módulo funcionario, en el cual se visualizará una lista de registros de usuarios asignados, asimismo podrá realizar búsqueda llamando el nombre de funcionario, al cual se le dará un usuario.
2. Se muestra el botón Nuevo Usuario, pulsar para realizar un nuevo registro.
3. Se tiene las siguientes acciones donde se podrán modificar datos y eliminar, asimismo tendremos el campo para ver si ese usuario está activo.

#### 4.1.4.1 Módulo Usuario - Botón Nueva Usuario

Después de pulsar en el menú Nuevo Usuario nos envía a la siguiente ventana

**Figura N° 1. 10 Módulo Usuario – Botón Nueva Usuario**



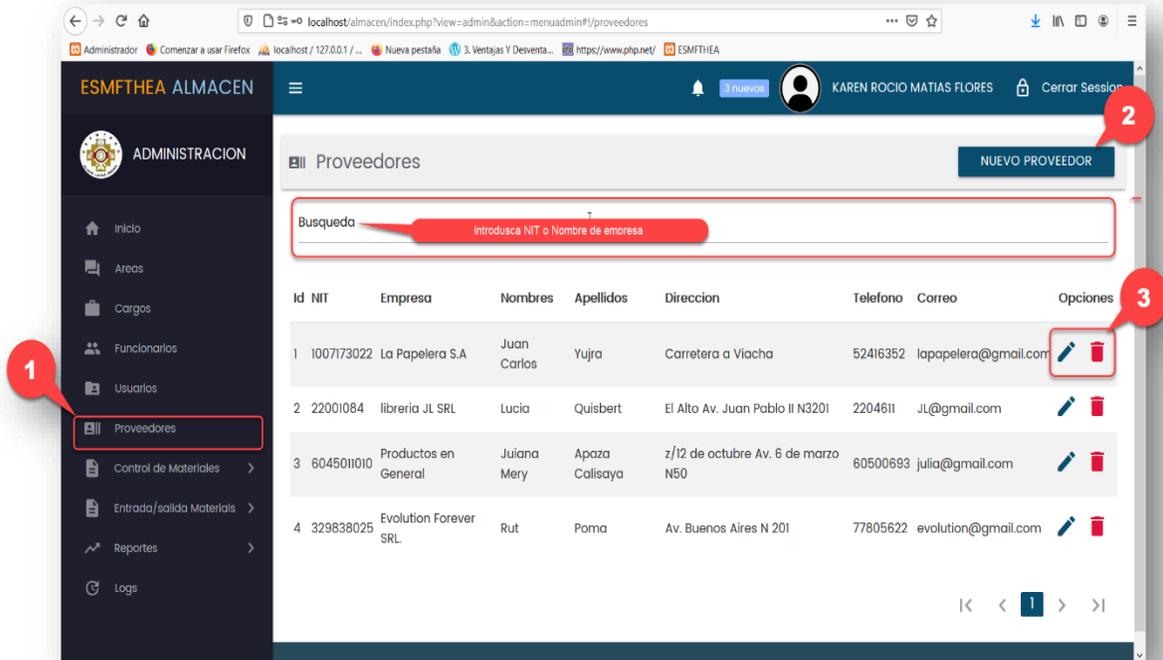
Fuente: Elaboración propia

1. Seleccionar Funcionario al cual se le designara su usuario, registrar el nombre de usuario y designar la contraseña común “**admin**”, asignar el tipo de usuario a todos los funcionarios excepto el administrador serán Usuario “Avanzado” y se les debe activar con el check. Todos los campos son necesarios para habilitar un nuevo usuario
2. Tenemos el botón **crear** que ayudara a registrar el nuevo usuario, y el botón **close** para cancelar el registro.

## 4.1.5 Módulo Proveedor

En el siguiente menú se podrá visualizar el Módulo del proveedor según detalle:

**Figura N° 1. 11 Módulo de Proveedor**



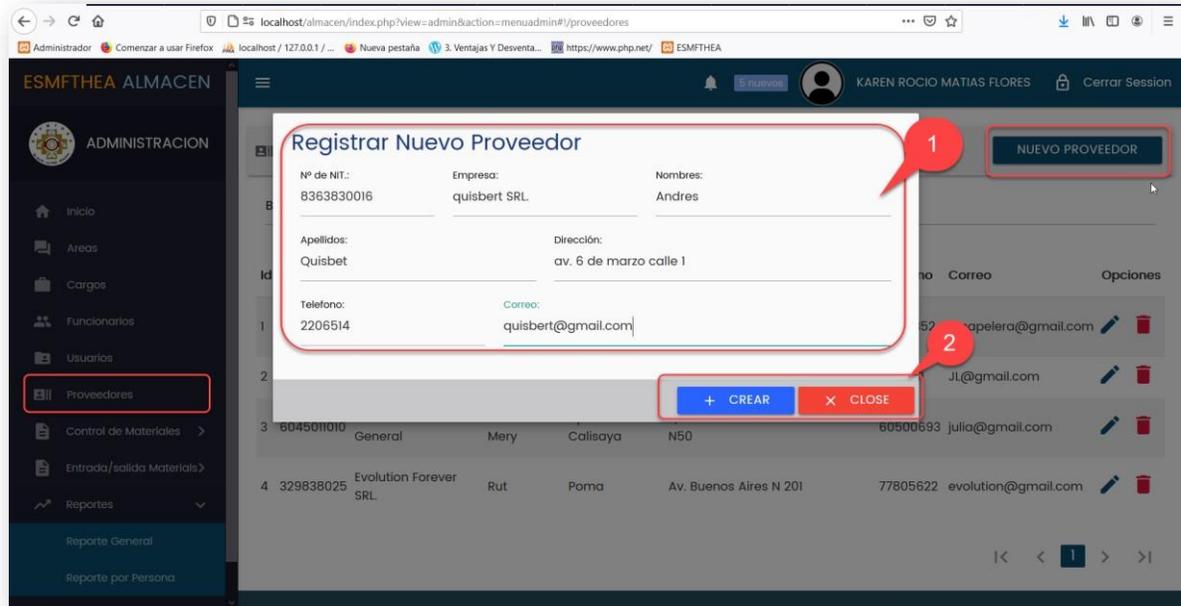
Fuente: Elaboración propia

1. Pulsar en el menú proveedores: el mismo que nos mostrará la ventana del módulo proveedores en el cual se visualizará una lista de registros de datos de proveedor, asimismo podrá realizar búsqueda llamando el nombre o NIT.
2. Se muestra el botón Nueva Proveedor, pulsar para realizar un nuevo registro.
3. Se tiene las siguientes acciones donde se podrá modificar datos y eliminar

#### 4.1.5.1 Módulo Proveedor - Botón Nuevo Proveedor

Después de pulsar en el menú Nuevo Proveedor nos envía a la siguiente ventana:

Figura N° 1. 12 Módulo Proveedor– Botón Nuevo Proveedor



Fuente: Elaboración propia

1. Llenar los campos solicitados para el registro de nuevo proveedor todos los campos son obligatorios para tener un buen registro.
2. Tenemos el botón **crear** que ayudara a registrar el nuevo cargo, y el botón **close** para cancelar el registro.

## 4.1.6 Módulo Control de Materiales

El módulo de control de materiales al desplegarse se dividirá en 4 submódulos los mismos que nos mostrarán a detalle el movimiento de material en almacén.

### 4.1.6.1 Sub-Módulo Categoría

En el siguiente menú se podrá visualizar el Sub Módulo categoría según detalle:

**Figura N° 1. 13 Módulo de Control de Materiales - Categoría**

Id	Tipo	Nombre de Categoría	Opciones
1	PE	Papereria de Escritorio	[Edit] [Delete]
2	EO	Material de Escritorio y Oficina	[Edit] [Delete]
3	PAG	Productos de Artes Grafiza Papel y Carton	[Edit] [Delete]
4	ML	MATERIAL DE LIMPIEZA	[Edit] [Delete]
5	MMP	PRODUCTOS MINERALES NO METALICOS Y PLASTICOS	[Edit] [Delete]

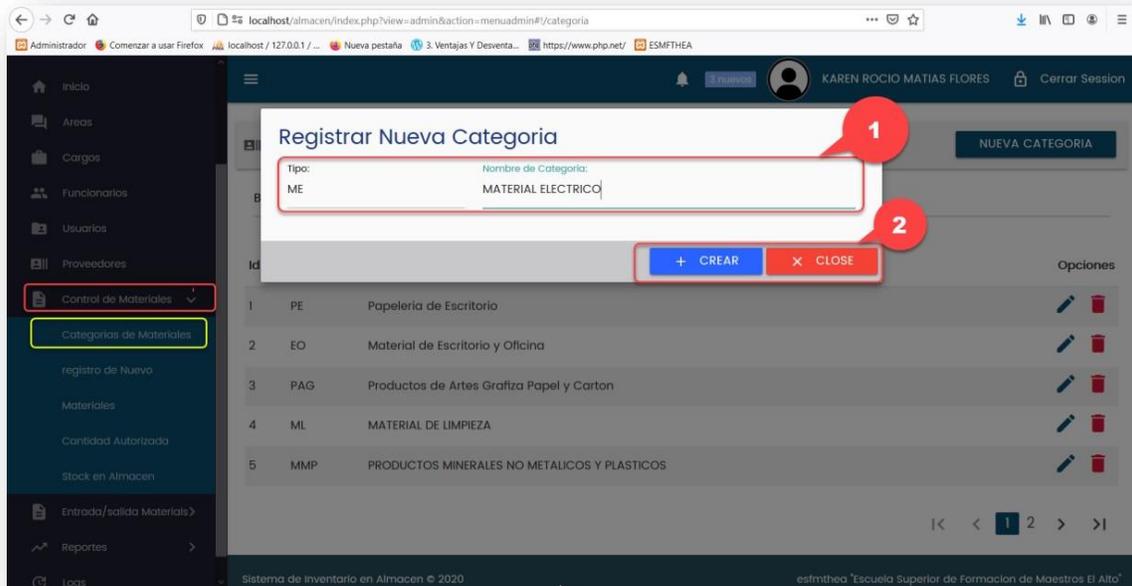
Fuente: Elaboración propia

1. Pulsar en el menú Categoría de Materiales: el mismo que nos mostrará la ventana del módulo categoría de materiales en el cual se visualizará una lista de registros de clasificación de materiales según el tipo, asimismo podrá realizar búsqueda llamando el nombre o tipo de categoría.
2. Se muestra el botón Nueva categoría, pulsar para realizar un nuevo registro.
3. Se tiene las siguientes acciones donde se podrá modificar y eliminar datos

#### 4.1.6.1.1 Sub Módulo Categoría de Materiales - Botón Nueva Categoría.

Después de pulsar en el Botón Nueva Categoría de Materiales nos envía a la siguiente ventana de registro.

**Figura N° 1. 14 Módulo Proveedor– Botón Nuevo Proveedor**



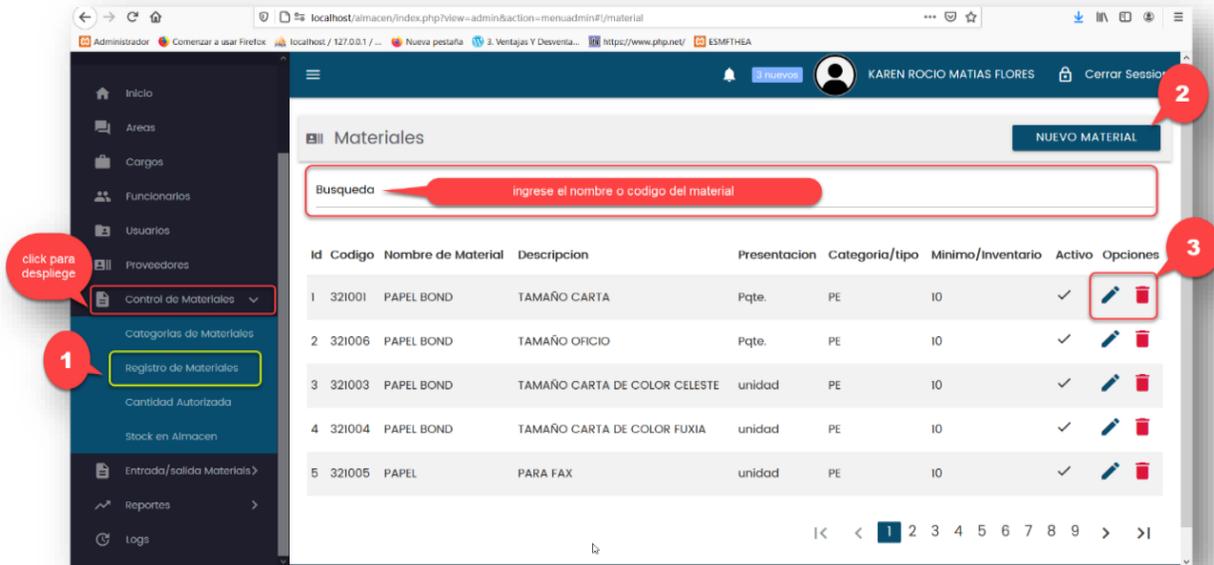
Fuente: Elaboración propia

1. Llenar los campos solicitados para el registro de nueva categoría, los campos son obligatorios para realizar el registro.
2. Tenemos el botón **crear** que ayudara a registrar la nueva categoría, y el botón close para cancelar el registro.

#### 4.1.6.2 Sub-Módulo Registro de Materiales

En el siguiente menú se podrá visualizar el Sub Módulo registro de materiales según detalle:

**Figura N° 1. 15 Módulo Control de Materiales – Registro de Materiales**



Fuente: Elaboración propia

1. Pulsar en el menú Categoría de Materiales: el mismo que se desplegará y mostrará la ventana del módulo registro de materiales en el cual se visualizará una lista de registros de materiales según el detalle, asimismo podrá realizar búsqueda llamando el nombre o código del material asignado.
2. Se muestra el botón Nuevo Material, pulsar para realizar un nuevo registro.
3. Se tiene las siguientes acciones donde se podrá modificar y eliminar datos

#### 4.1.6.2.1 Sub Módulo Registro de Materiales - Botón Nuevo Material

Después de pulsar en el botón Nuevo Material nos envía a la siguiente ventana:

**Figura N° 1. 16 Módulo Registro de Material – Nuevo Material**

The screenshot shows a web application interface for registering new materials. The main form is titled "Registrar Nuevo Material" and contains the following fields:

- Codigo:** 331026
- Tipo/Categoria:** EO (Material de Escritorio y Oficina)
- Nombre de Material:** marcadores color azul
- Descripción:** con alcohol
- Minimo en Inventario:** 10
- Presentación:** Unidad

Below the form are two buttons: "+ CREAR" (blue) and "X CLOSE" (red). A red circle with the number "1" highlights the form fields, and another red circle with the number "2" highlights the buttons.

The background shows a sidebar menu with "Control de Materiales" and "Registro de Materiales" highlighted. Below the form, there is a table of materials with columns for ID, Codigo, Nombre, Tipo/Categoria, Unidad, and Opciones.

ID	Codigo	Nombre	Tipo/Categoria	Unidad	Opciones
35	331025	ARCHIVADOR	TAMAÑO OFICIO	unidad	EO 5

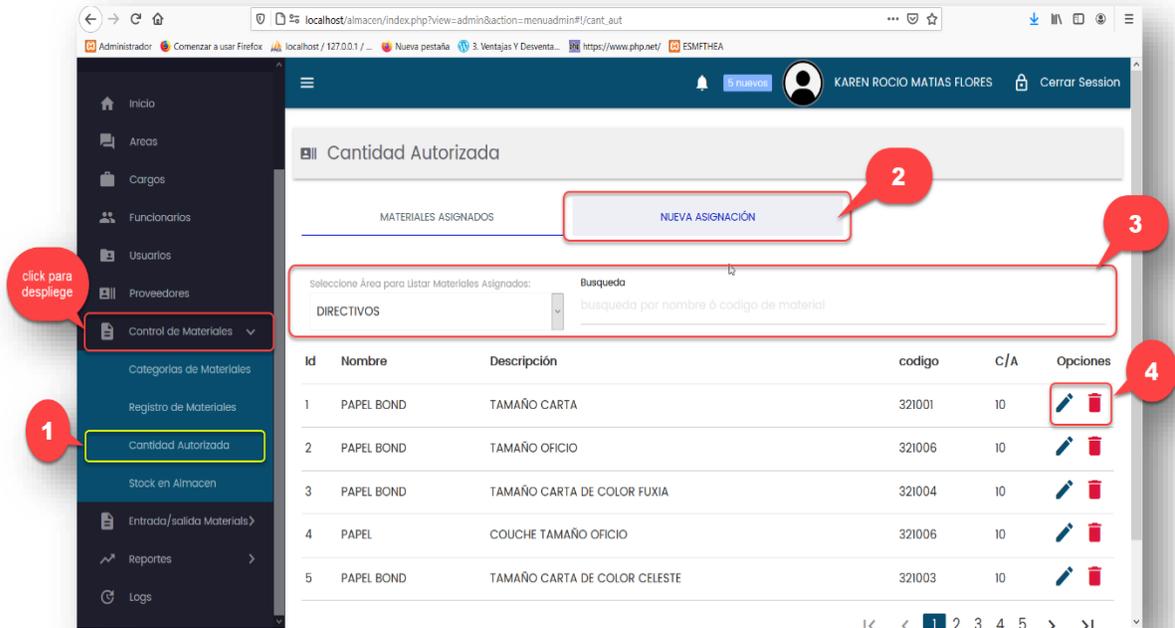
Fuente: Elaboración propia

1. Llenar los campos solicitados para el registro de nuevo material los campos son obligatorios para realizar el registro.
2. Tenemos el botón **crear** que ayudara a registrar la nueva categoría, y el botón **close** para cancelar el registro.

### 4.1.6.3 Sub-Módulo Cantidad Autorizada

Desplegando el Módulo control de materiales, en el siguiente menú se podrá visualizar el Sub Módulo Cantidad Autorizada según detalle:

**Figura N° 1. 17 Interfaz de Sub Cantidad Autorizada**



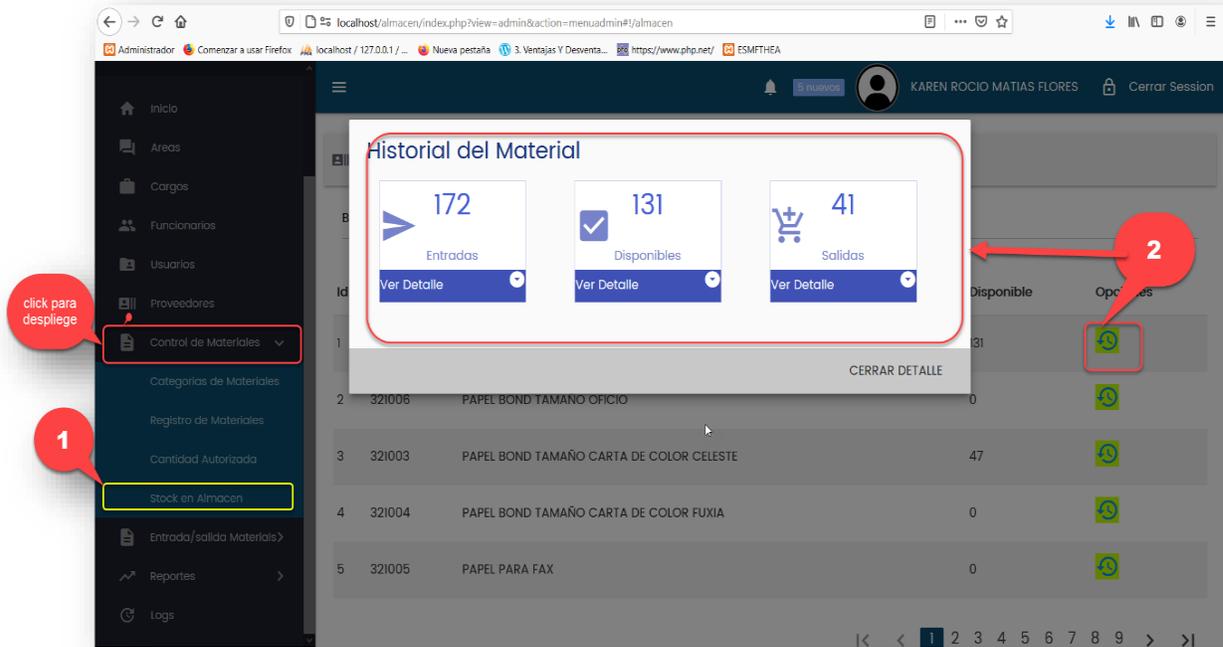
Fuente: Elaboración propia

1. Pulsar en el menú Categoría de Materiales: el mismo que se desplegará y mostrará la ventana del **módulo cantidad autorizada** en el cual se visualizará dos opciones:
2. Nueva asignación (se asignará la cantidad de un nuevo material según el área el mismo y podrá guardar o eliminar para realizar el registro)
3. Material asignado (nos muestra una lista si pulsamos en material asignado y se debe elegir el área)
4. Se tiene las siguientes acciones donde se podrás modificar y eliminar datos

#### 4.1.6.4 Sub-Módulo Stock en Almacén

Desplegando el Módulo control de materiales, en el siguiente menú se podrá visualizar el Sub Módulo Stock en Almacén según detalle:

**Figura N° 1. 18 Interfaz Sub Módulo Stock en Almacén**



Fuente: Elaboración propia

1. Pulsar en el menú Categoría de Materiales: el mismo que se desplegará y mostrará la ventana del **módulo Stock en Almacén**, en el cual se visualizará los detalles de registro de todo el dato de material.
2. Tenemos el icono Opciones si ponchamos en ahí visualizaremos el historial de ese material registrado en almacén.

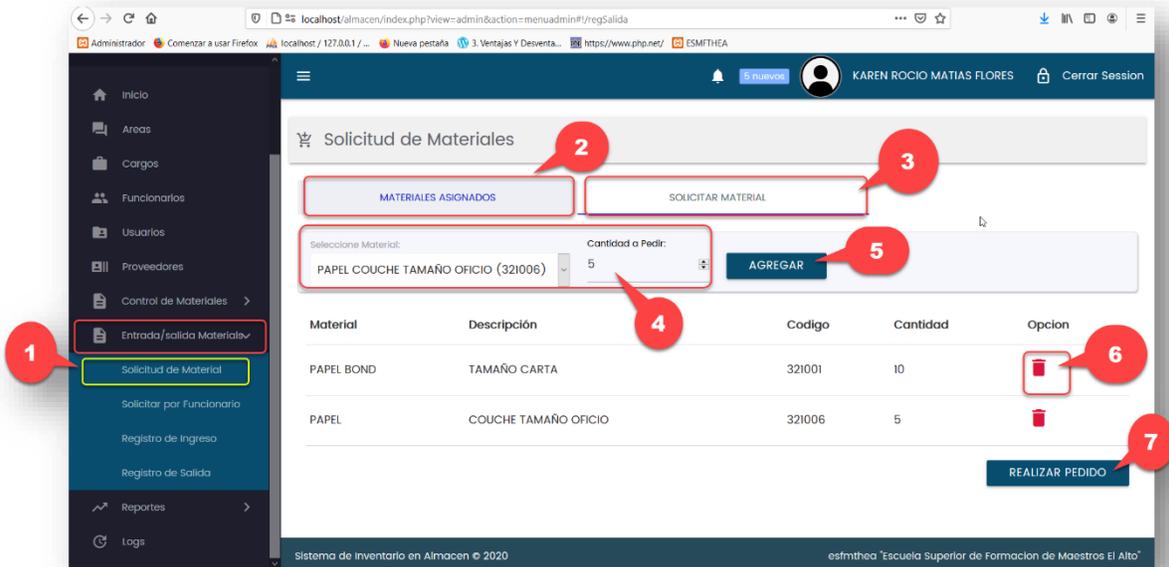
#### 4.1.7 Módulo Entrada y Salida de Materiales

El módulo de Entrada y Salida de Materiales al desplegar se dividirá en 4 sub módulos los mismos que ayudaran al manejo de movimiento de material.

#### 4.1.7.1 Sub-Módulo Solicitud de Material.

En el siguiente menú se podrá visualizar el Sub Módulo Solicitud de material según detalle:

**Figura N° 1. 19 Interfaz Sub Módulo Solicitud de material**



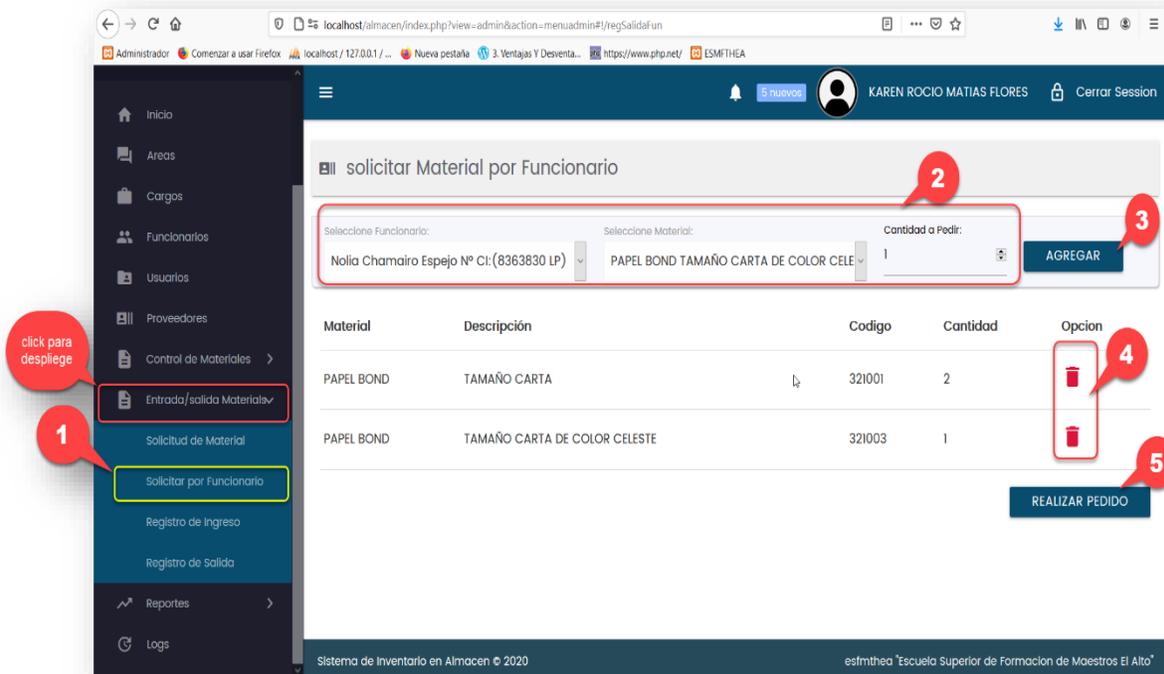
Fuente: Elaboración propia

1. Pulsar en el menú sub Módulo solicitud de material: se visualizará dos opciones: Material Asignado y Solicitud de Material.
2. Material asignado mostrará al administrador que tipo de material registrado y si hay disponible en stock de almacén.
3. Solicitar Material, desde esa pantalla el funcionario administrador podrá realizar su pedido de material,
4. Registrar datos de solicitud seleccionando la opción de material e introducir cantidad.
5. **Agregar** nos permite realizar el requerimiento de materiales, se puede seleccionar diferentes materias en un solo pedido,
6. Asimismo, se cuenta con la opción eliminar en caso de no proceder con el pedido de material.
7. Botón realizar pedido, mismo que confirmara la solicitud de material.

#### 4.1.7.2 Sub-Módulo Solicitar por Funcionario

En el siguiente menú se podrá visualizar el Sub Módulo Solicitar por Funcionario según detalle:

**Figura N° 1. 20 Interfaz Sub Módulo Solicitar por Funcionario**



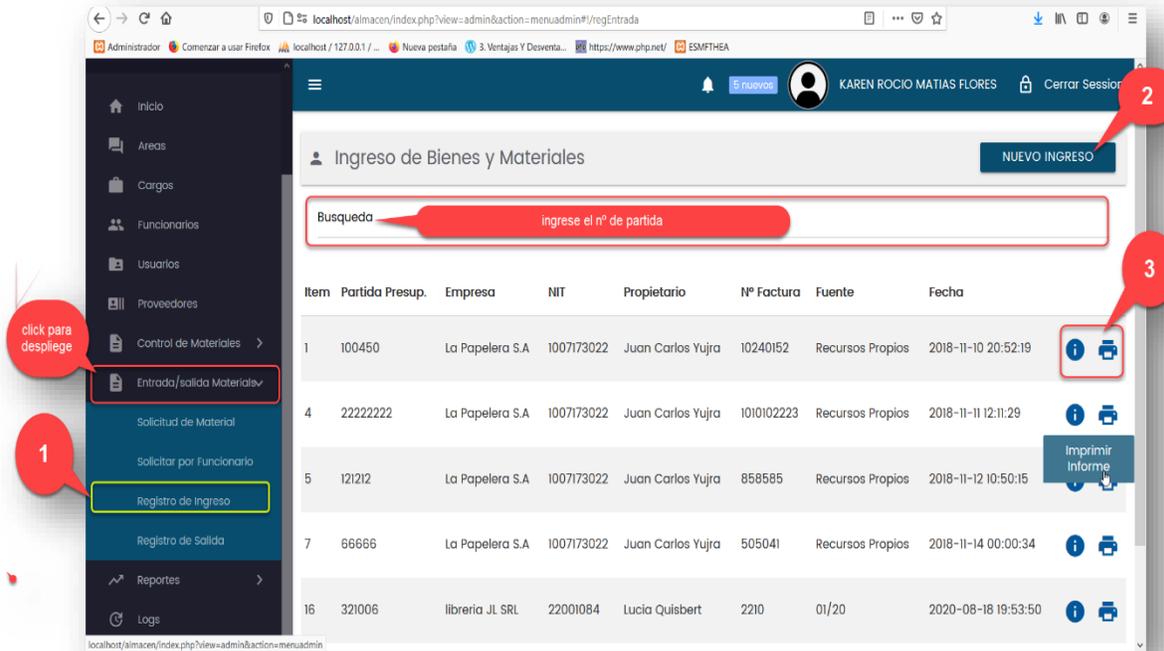
Fuente: Elaboración propia

1. Pulsar en el menú Sub Módulo **Solicitar por Funcionario**: Se visualizará la opción para solicitar materiales por funcionario.
2. Debe seleccionar el funcionario que solicitará, tipo y cantidad de material.
3. Botón **agregar**, se puede realizar varios pedidos de materiales en un solo requerimiento, siendo la función principal de este botón.
4. **Eliminar**, ayuda a anular un requerimiento solicitado preliminarmente elaborado.
5. Botón **realizar pedido**, confirma la solicitud de material.

### 4.1.7.3 Sub-Módulo Registro de Ingresos

En el siguiente menú se podrá visualizar el Sub Módulo **Registro de Ingresos** nos mostrará una lista de detalles de los materiales que ingresan según la compra realizada.

**Figura N° 1. 21 Interfaz Sub Módulo Registro de Ingreso**



**Fuente:** Elaboración propia

1. Pulsar en el menú Sub Módulo **Registro de Ingreso**: se visualizará la opción una lista de ingreso de bienes y materiales según registro realizado, asimismo podremos realizar búsqueda de materiales por la partida presupuestaria o fecha.
2. Botón **Nuevo Registro**, si pulsamos nos mostrara algunos detalles para realizar el registro.
3. También tenemos la opción de **imprimir y generar** un reporte de registro de material ingresado.

#### 4.1.7.3.1 Sub-Módulo Registro de Ingresos – Botón Nuevo Ingreso

**Pulsar** en el botón Nuevo Ingreso de material, nos ayudara a registrar todos los materiales que llegan a la institución.

**Figura N° 1. 22 Interfaz Sub Módulo Registro de Ingresos –Nuevo Ingreso**

Item	Codigo	Material	Presentacion	Disponibles	Cantidad	P/Unitario	Total	Opciones
1	331006	CUADERNO TAMAÑO MEDIO OFICIO CON ASPIRAL	unidad	100	50	2.5	125 Bs.	

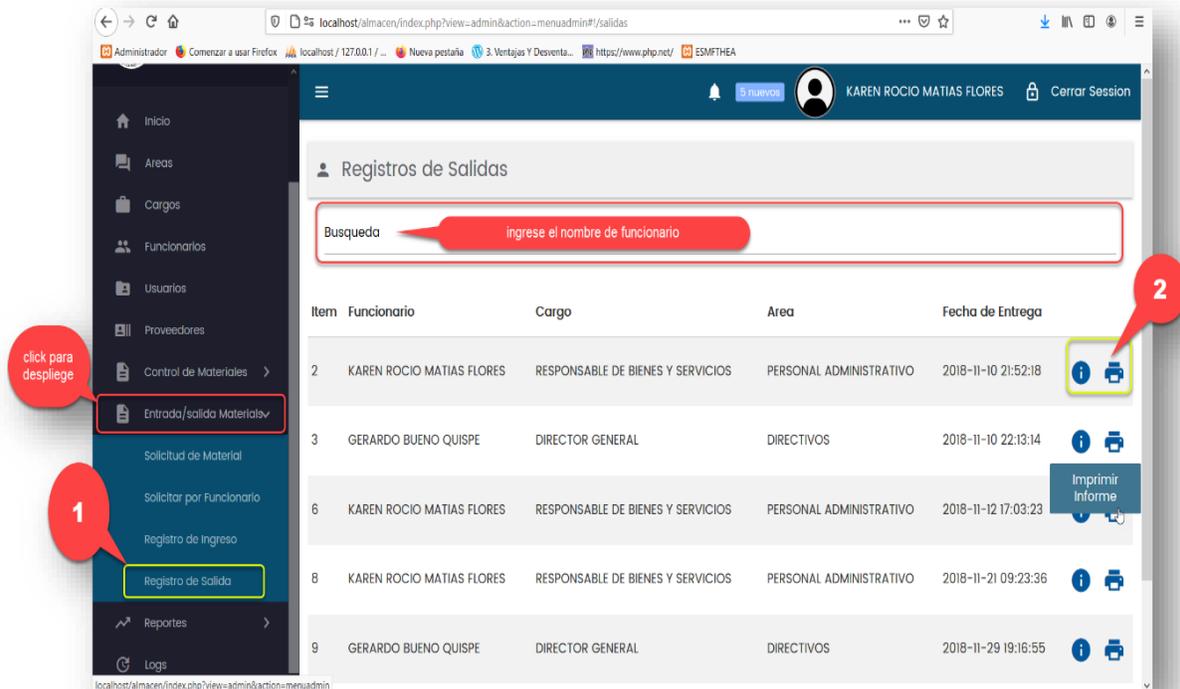
**Fuente:** Elaboración propia

1. Llenar todos los campos que son necesarios, Seleccione el nombre del proveedor y NIT que son datos importantes para registrar el origen del material adquirido, Partida presupuestaria que refleja los fondos previstos para dichas compras, Numero de factura con la que se efectúa la compra y la Fuente, que refleja el origen de los fondos para la compra del material. Se debe llenar todas las opciones siguiendo el detalle del material adquirido adjuntando sus respectivos precios.
2. Botón agregar, nos da la opción de registrar todos los materiales comprados en una sola factura.
3. Eliminar, esta opción se tiene en caso de realizar alguna corrección o anular un registro no deseado.
4. Botón **crear**, ejecuta el registro, el botón **cancelar**, anula el registro de la operación.

#### 4.1.7.4 Sub-Módulo Registro de Salida

En el siguiente menú se podrá visualizar el Sub Módulo **Registro de Salida** que nos muestra un listado detallado de los materiales que salen de almacén.

**Figura N° 1. 23 Interfaz de Sub Módulo Registro de Salida**



**Fuente:** Elaboración propia

1. Pulsar en el menú sub módulo **Registro de Salida**: se visualizará la opción con un detalle de materiales que se solicitó y se entregó ordenando por fecha, también se puede realizar la búsqueda de pedido según el nombre de funcionario.
2. Se podrá realizar la impresión, pulsando en la opción impresora.

## 4.1.8 Módulo Reportes

El módulo de Reportes al desplegarse se dividirá en 2 sub módulos el mismo que nos ayudaran a visualizar el detalle de todos los materiales.

### 4.1.8.1 Sub-Módulo Reporte General

En el siguiente menú se podrá visualizar en la figura el Sub Módulo **Reporte General**.

**Figura N° 1. 24 Interfaz Sub Módulo Reporte General**

The screenshot shows the 'Reporte General' interface. The sidebar menu on the left has 'Reportes' selected, highlighted with a red box and callout 1. The main content area has a header 'Reporte General' and a form for date selection. The form has 'Fecha Inicial' set to '02/06/2018' and 'Fecha Final' set to '07/08/2020', both highlighted with a red box and callout 2. Below the form is an 'EXPLORAR' button and a 'PDF' icon, both highlighted with a red box and callout 3. The table below shows the following data:

Codigo	Tipo	Material Suministro	Unidad	transaccion	Fecha
1	321001	PE PAPEL BOND TAMAÑO CARTA	Pqte.	entrada	2018-11-10 20:52:19
1	321001	PE PAPEL BOND TAMAÑO CARTA	Pqte.	entrada	2018-11-11 12:11:29
1	321001	PE PAPEL BOND TAMAÑO CARTA	Pqte.	entrada	2018-11-14 00:00:34
3	321003	PE PAPEL BOND TAMAÑO CARTA DE COLOR CELESTE	unidad	entrada	2018-11-12 10:50:15

Fuente: Elaboración propia

1. Pulsar en el menú sub Módulo Reporte General: se visualizará las opciones para generar el reporte.
2. Seleccione las fechas de inicio y fechas de finalización para explorar el reporte, el mismo que visualizara la lista solicitada.
3. Pulsar en el icono PDF el cual visualizará y generará el reporte final solicitado la cual puede ser impreso o descargado para su archivo final.

**Figura N° 1. 25 Reporte PDF**

CODIGO	TIPO	MATERIAL SUMINISTRO	UNIDAD	BALDO AL 20-08-2018			ENTRADA		BALDA			BALDO AL 04-08-2020				
				CANTIDAD	PU	IMPORTE	CANTIDAD	PU	IMPORTE	CANTIDAD	PU	IMPORTE	CANTIDAD	PU	IMPORTE	
321001	PE	PAPEL BOND TAMAÑO CARTA	Pqte.	400	20.00	2000					40	20.00	800	50	20.00	1000
321001	PE	PAPEL BOND TAMAÑO CARTA	Pqte.	22	50.00	1100						0	0	22	50.00	1100
321001	PE	PAPEL BOND TAMAÑO CARTA	Pqte.	50	25.00	1250						0	0	50	25.00	1250
321003	PE	PAPEL BOND TAMAÑO CARTA DE COLOR CELESTE	unidad	50	1.50	75					3	1.50	4.5	47	1.50	70.5
TOTAL						4425							804.5			3620.5

Fuente: Elaboración propia

#### 4.1.8.2 Sub-Módulo Reporte por Funcionario

En el siguiente menú se podrá visualizar en la figura el Sub Módulo **Reporte por Funcionario**.

**Figura N° 1. 26 Interfaz de Sub Módulo de Reporte por Funcionario**

click para despliegue

1

2

3

Item	Codigo	Materia Suministro	Unidad	Solicitado	Entregado	Fecha de Entrega
1	321001	PAPEL BOND TAMAÑO CARTA	Pqte.	10	10	2018-11-10 22:13:14
2	321001	PAPEL BOND TAMAÑO CARTA	Pqte.	5	5	2018-11-29 19:16:55

Fuente: Elaboración propia

1. Pulsar en el menú sub Módulo Reporte por Funcionario: se visualizará las opciones para generar el reporte.
2. Seleccione las fechas de inicio y fechas de finalización asimismo seleccionar el funciona del cual se quiera solicitar el reporte.
3. Pulsar en icono PDF el cual visualizará y generará el reporte final solicitado.

**Figura N° 1. 27 Interfaz Reporte por Funcionario – PDF Funcionario**

ESM T H A  
E S M T H A  
EL ALTO - LA PAZ - BOLIVIA

ESCUELA SUPERIOR DE FORMACIÓN DE MAESTROS TECNOLÓGICO HUMANÍSTICO DE EL ALTO  
DIRECCIÓN ADMINISTRATIVA FINANCIERA  
ALMACENES E INVENTARIO  
(DEL 30 de julio del 2018 AL 30 de julio del 2020)

PDF

**REPORTE DE MATERIALES ENTREGADOS**

CI.: 2799344 OR      FECHA: 23-08-2020

SOLICITANTE: GERARDO BUENO QUISPE

CARGO: DIRECTOR GENERAL

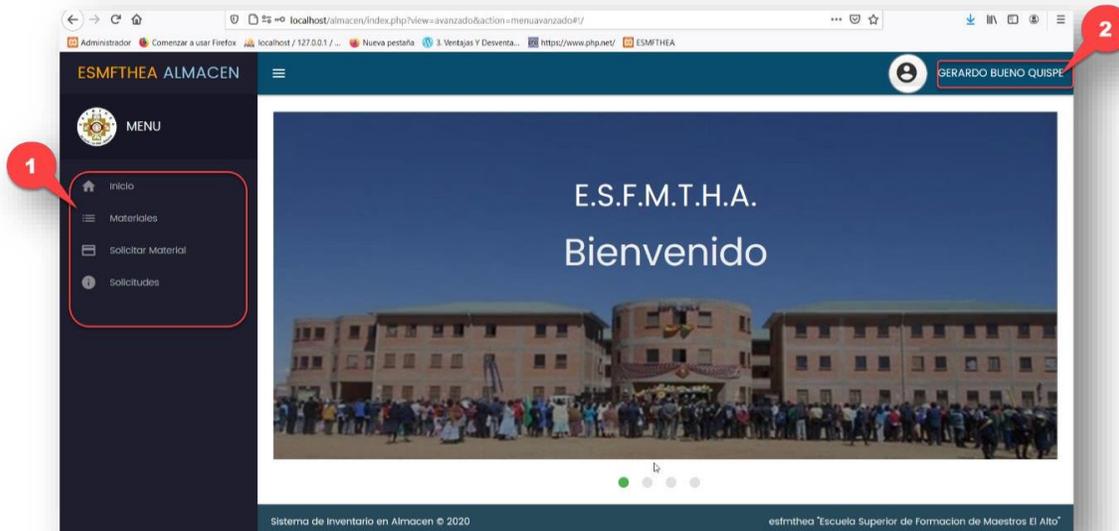
ITEM	CODIGO	MATERIAL SUMINISTRO	UNIDAD	CANTIDAD SOLICITADA	CANTIDAD ENTREGADA	FECHA DE ENTREGA
1	321001	PAPEL BOND TAMAÑO CARTA	Pqte.	10	10	10-11-2018
2	321001	PAPEL BOND TAMAÑO CARTA	Pqte.	5	5	29-11-2018

Fuente: Elaboración propia

## 4.2 INGRESO USUARIO FUNCIONARIO

Ingrese al sistema **Usuario Funcionario**, en la interfaz se visualizará el menú habilitado para ese tipo de usuario.

**Figura N° 1. 28 Interfaz Usuario Funcionario**



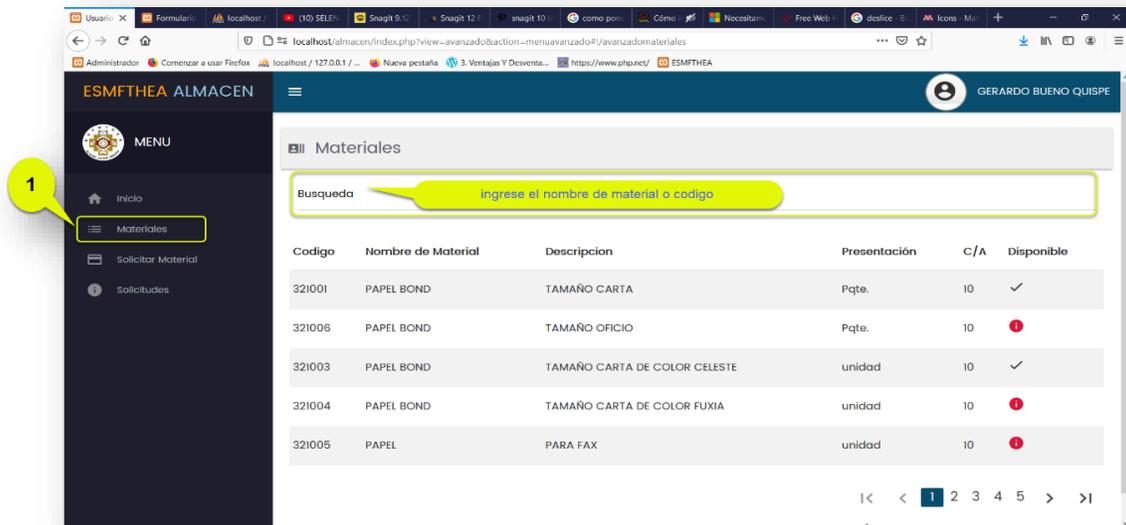
Fuente: Elaboración propia

1. Se mostrará el acceso a los siguientes menús, para realizar el requerimiento de materiales.
2. Pulsar en el nombre de usuario te habilitara dos opciones **mi perfil** y **cerrar sesión**.

## 4.2.1 Módulo Materiales

En el presente Módulo ayudara al funcionario a saber la existencia de material en almacén.

**Figura N° 1. 29 Interfaz Materiales**



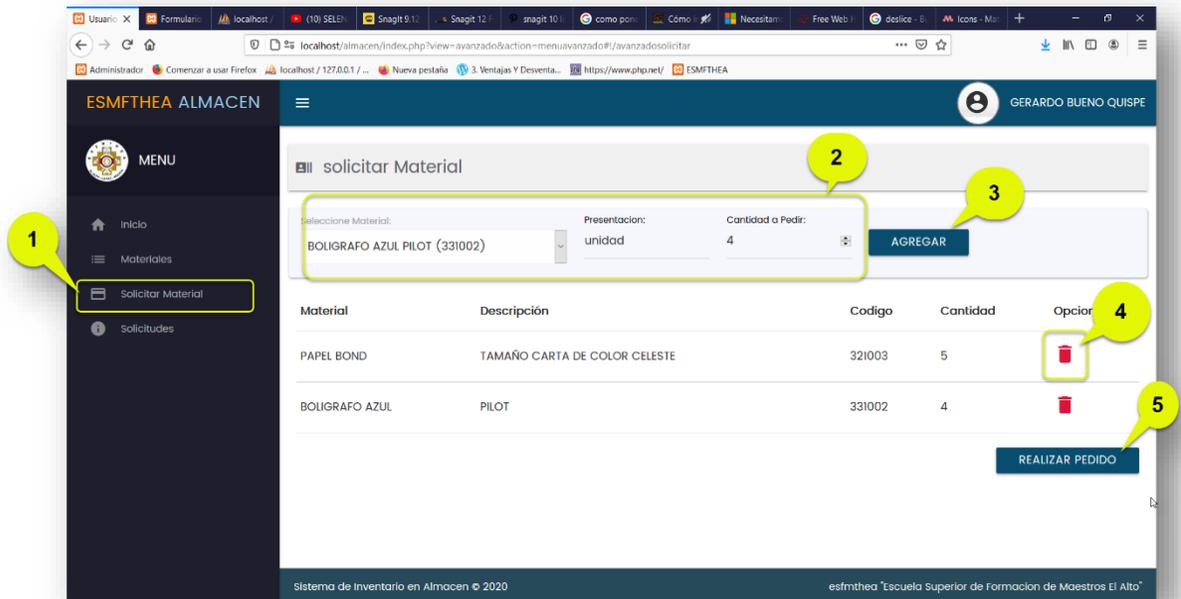
Fuente: Elaboración propia

1. Pulse en Menú Material, el mismo que nos mostrara una lista de disponibilidad de materiales los cuales solicita el funcionario desde allí también puede realizar una búsqueda rápida de material ingresando y el nombre de dicho material.

## 4.2.2 Módulo Solicitud de Materiales

En el presente Módulo ayudara al funcionario a solicitar los materiales requeridos.

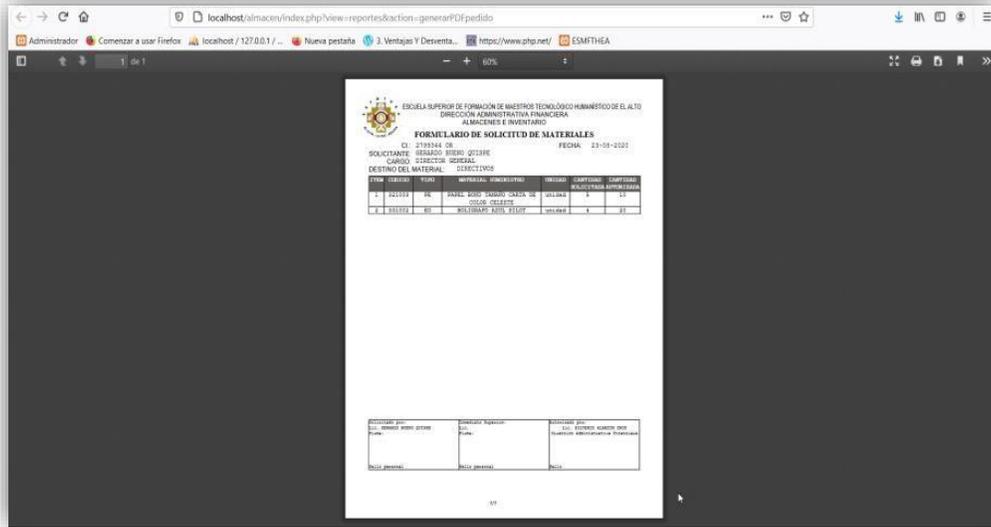
**Figura N° 1. 30 Interfaz Módulo Solicitud de Materiales**



Fuente: Elaboración propia

1. Pulsar en Menú **Solicitar Materiales**: se visualizara una la pantalla para realizar el requerimiento según detalle.
2. Llenar los campos, seleccionar el material y la cantidad
3. Botón **agregar** te da la opción de solicitar varios materiales, si pulsa en él va generando una lista según se vaya registrando los materiales a solicitar.
4. Se tiene la opción a eliminar en caso de que se decida no pedir algún material.
5. Botón **realizar pedido**, genera la solicitud de los materiales solicitados por el funcionario, asimismo genera un reporte pdf de nota de pedido según figura.

**Figura N° 1. 31 PDF- Nota de Solicitud**

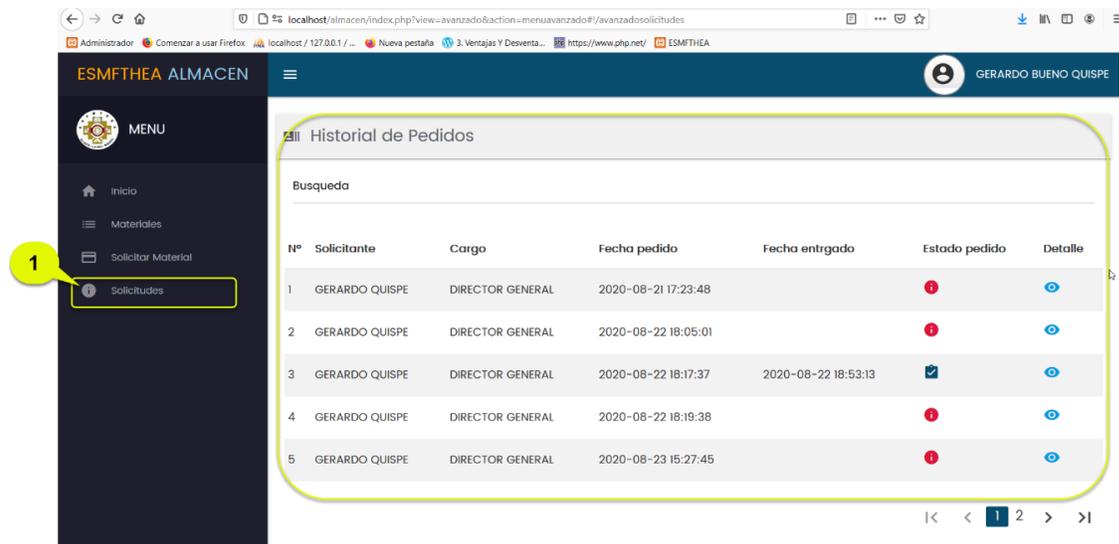


Fuente: Elaboración propia

#### 4.2.3 Módulo Solicitudes

En el presente módulo de solicitudes el **Usuario Funcionario** podrá visualizar una lista de todas las solicitudes que realizó con su usuario, asimismo podrá evidenciar si su solicitud fue aprobada o está pendiente para su aprobación.

**Figura N° 1. 32 Módulo Solicitud**



Fuente: Elaboración propia

1. Pulse en solicitud, y visualizará el historial de sus solicitudes.



# MANUAL TECNICO

---

## “SISTEMA DE INFORMACIÓN PARA EL CONTROL DE INVENTARIO EN ALMACÉN”

**E.S.F.M.T.H.E.A.**

“Escuela Superior de Formación de Maestros Tecnológico Humanístico El Alto”

---

**Nolia Chamairo Espejo**

**LA PAZ – BOLIVIA**



# INDICE MANUAL TÉCNICO

PAG.

## MANUAL TÉCNICO

<b>1. INTRODUCCIÓN</b> .....	<b>1</b>
<b>2. REQUERIMIENTOS TÉCNICOS.</b> .....	<b>1</b>
2.1 REQUERIMIENTOS MÍNIMOS DE HARDWARE .....	1
2.2. REQUERIMIENTOS MÍNIMOS DE SOFTWARE.....	1
<b>3. HERRAMIENTAS UTILIZADAS PARA EL DESARROLLO</b> .....	<b>2</b>
3.1 PHP.....	2
3.2 MYSQL.....	2
3.3 APACHE.....	2
3.4 INSTALACIÓN DE WEB SERVER.....	3
<b>4. CREANDO PROYECTO CON ANGULAR JS</b> .....	<b>7</b>
4.1. CONFIGURAR LA BASE DE DATOS.....	8
4.2 SERVIDOR ACTIVO.....	9
<b>5. CASOS DE USO</b> .....	<b>9</b>
<b>6 MODULO DE ADMINISTRACION</b> .....	<b>10</b>

## INDICE DE FIGURAS

	<b>PÁG.</b>
<i>FIGURA N° 1. 1 XAMPP INSTALACIÓN .....</i>	<i>3</i>
<i>FIGURA N° 1. 2 DIRECCIONAMIENTO PARA LA INSTALACIÓN.....</i>	<i>4</i>
<i>FIGURA N° 1. 3 INICIAR INSTALACIÓN.....</i>	<i>4</i>
<i>FIGURA N° 1. 4 CONFIGURAR FIREWALL .....</i>	<i>5</i>
<i>FIGURA N° 1. 5 FINALIZAR INSTALACIÓN.....</i>	<i>5</i>
<i>FIGURA N° 1. 6 INTERFAZ DEL SERVIDOR CONFIGURABLE.....</i>	<i>6</i>
<i>FIGURA N° 1. 7 INICIANDO EL PUERTO APACHE Y MYSQL.....</i>	<i>7</i>
<i>FIGURA N° 1. 8 CREACIÓN DE PROYECTO.....</i>	<i>8</i>
<i>FIGURA N° 1. 9 CONFIGURACIÓN DE BASE DE DATOS .....</i>	<i>8</i>
<i>FIGURA N° 1. 10 DIAGRAMA CASO USO-ADMINISTRADOR.....</i>	<i>9</i>
<i>FIGURA N° 1. 11 PANTALLA DE INICIO DE SESIÓN.....</i>	<i>10</i>

## OBJETIVOS

Brindar la información necesaria para poder realizar la instalación y configuración del aplicativo.

Detallar la información necesaria para realizar la instalación, configuración y puesta en funcionamiento del aplicativo.

## OBJETIVO ESPECIFICO

- Definir claramente el proceso de instalación del aplicativo.
- Detallar los requerimientos mínimos de hardware y software para el funcionamiento del aplicativo.
- Describir las herramientas utilizadas en el desarrollo del aplicativo.

## 1. INTRODUCCIÓN

En este manual se describa los pasos necesarios para poner en funcionamiento el aplicativo, se requiere que la persona encargada de la instalación, tenga conocimientos básicos de sistemas.

## 2. REQUERIMIENTOS TÉCNICOS.

### 2.1 REQUERIMIENTOS MÍNIMOS DE HARDWARE

*Tabla 1 Datos de Hardware*

<b>Procesador</b>	Core
<b>Memoria RAM: Mínimo</b>	2 Gigabytes (GB)
<b>Disco Duro</b>	500Gb.

Fuente: (Elaboración Propia)

### 2.2. REQUERIMIENTOS MÍNIMOS DE SOFTWARE

 Privilegios de administrador

 **Sistema Operativo** : Windows 8, Windows 10 en adelante.

## **3. HERRAMIENTAS UTILIZADAS PARA EL DESARROLLO**

### **3.1 PHP**

Es un Lenguaje de Programación para trabajar páginas WEB ofreciendo la ventaja de combinar con HTML. Las ejecuciones son realizadas en el Servidor y el cliente es el encargado de recibir los resultados de la ejecución. Si el cliente realiza una petición, se ejecuta el intérprete de PHP y se genera el contenido de manera dinámica. Permite conexión con varios tipos de Bases de Datos como: MySql, SQL Server, etc. permitiendo aplicaciones robustas sobre la WEB. Este lenguaje de programación puede ser ejecutado en la gran mayoría de sistemas operacionales y puede interactuar con Servidores WEB populares

### **3.2 MYSQL**

Es un manejador de Bases de Datos, el cual permite múltiples hilos y múltiples usuarios, fue desarrollado como software libre.

Aunque se puede usar sobre varias plataformas es muy utilizado sobre LINUX. Es libre para uso en Servidores WEB.

Ofrece ventajas tales como fácil adaptación a diferentes entornos de desarrollo,

Interacción con Lenguajes de Programación como PHP, Java Script y fácil Integración con distintos sistemas operativos.

### **3.3 APACHE**

Es un Servidor WEB desarrollado por el grupo Apache. Su código fuente se puede distribuir y utilizar de forma libre. Está disponible para diferentes plataformas de Sistemas Operativos entre otros Windows, Linux, Mac y NetWare.

Ofrece ventajas tales como independencia de plataforma, haciendo posible el cambio de plataforma en cualquier momento; creación de contenidos dinámicos, permitiendo crear sitios mediante lenguajes PHP.

Además de ser libre su soporte técnico es accesible ya que existe una comunidad que está disponible en foros, canales IRC y servidores de noticias, donde hay gran cantidad de usuarios disponibles para cuando surge algún problema.

### 3.4 INSTALACIÓN DE WEB SERVER

Descargar al PC el fichero de instalación xampp-win32-7.2.10-0-VC15-installer - Apache/2.4.34 (Win32) 32b.exe el cuál se encuentra en la página: <https://xampp.uptodown.com/windows> una vez descargado, dar clic sobre él para iniciar la instalación. La instalación se debe hacer con una cuenta de Administrador o con derechos de administrador. Durante la instalación aparecerán las siguientes pantallas:

En la primera ventana aparece la pantalla de inicio del asistente para instalar XAMPP. Para ajustar las configuraciones de la instalación se hace clic en “Next”.

**Figura N° 1. 33 Xampp instalación**



**Fuente:** (Elaboración Propia)

La siguiente ventana presionar el botón **Next** Instalación y administración para continuar.

En este paso se escoge el directorio donde se instalará el paquete. Si se ha escogido la configuración estándar se creará una carpeta con el nombre XAMPP en C:|. **Next** Para continuar.

**Figura N° 1. 34 Direccionamiento para la Instalación**



**Fuente:** (Elaboración Propia).

El asistente extrae los componentes seleccionados y los guarda en el directorio seleccionado en un proceso que puede durar algunos minutos. El avance de la instalación se muestra como una barra de carga de color verde.

**Figura N° 1. 35 Iniciar Instalación.**



**Fuente:** (Elaboración Propia).

Durante el proceso de instalación es frecuente que el asistente avise del bloqueo de Firewall. En la ventana de diálogo puedes marcar las casillas correspondientes

para permitir la comunicación del servidor Apache en una red privada o en una red de trabajo. Recuerda que no se recomienda usarlo en una red pública.

**Figura N° 1. 36 Configurar Firewall**



**Fuente:** (Elaboración Propia).

Una vez extraídos e instalados todos los componentes puedes cerrar el asistente con la tecla “Finish”. Para acceder inmediatamente al panel de control solo es necesario marcar la casilla que pregunta si deseamos hacerlo.

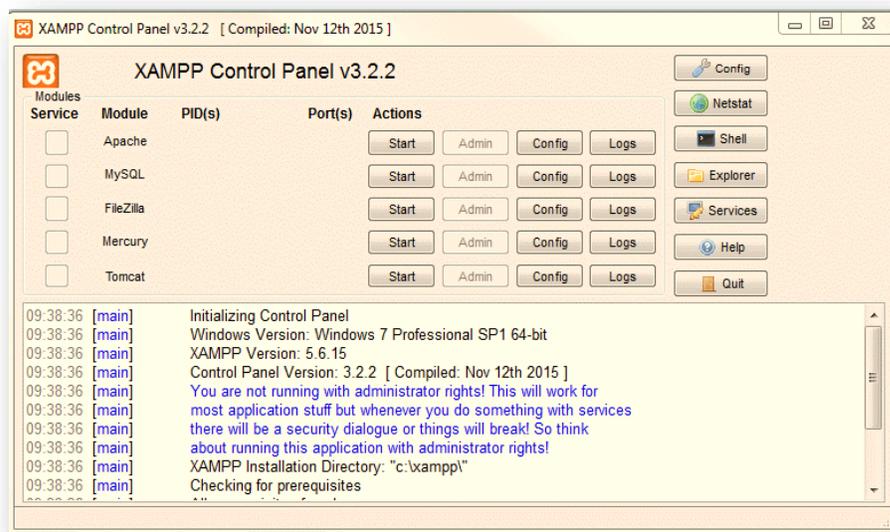
**Figura N° 1. 37 Finalizar instalación**



En la interfaz de usuario del panel de control se protocolan todas las acciones y es posible activar o desactivar los módulos por separado con un simple clic. Además, se dispone de diversas utilidades como:

- ✓ Config: para configurar XAMPP así como otros componentes aislados.
- ✓ Netstat: muestra todos los procesos en funcionamiento en el ordenador local
- ✓ Shell: lanza una ventana de comandos UNIX
- ✓ Explorer: abre la carpeta XAMPP en el explorador de Windows
- ✓ Services: muestra todos los servicios en funcionamiento
- ✓ Help: incluye enlaces a foros de usuarios
- ✓ Quit: se usar para salir del panel de control.

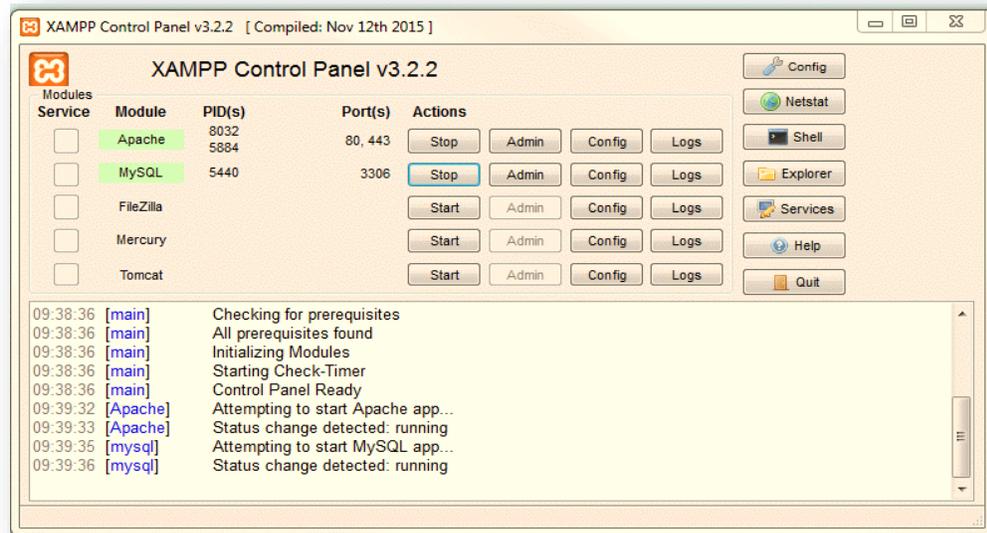
**Figura N° 1. 38 Interfaz del servidor configurable**



**Fuente:** (Elaboración Propia).

En la parte superior se pueden iniciar o interrumpir los módulos de XAMPP por separado mediante los comandos "Start" y "Stop" bajo "Actions". Los módulos que se activaron aparecen marcados en verde.

**Figura N° 1. 39 Iniciando el puerto Apache y MySql**



**Fuente:** (Elaboración Propia).

## 4. CREANDO PROYECTO CON ANGULAR JS

Existen dos formas de crear un proyecto con ANGULAR JS, la primera es descargando el archivo master desde su repositorio oficial de GitHub y la otra es usando Composer desde la consola que es precisamente lo que haremos.

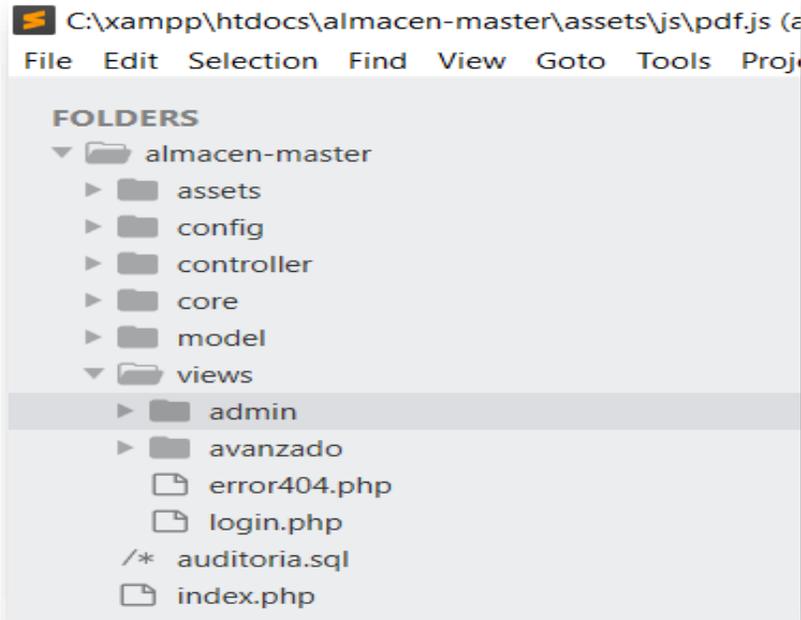
Desde la consola, dirígete al directorio donde guardas tus proyectos web (si usas XAMPP la ruta es C:\xampp\htdocs y teclea lo siguiente:

✓ C:\xampp\htdocs

Ahora crearemos el proyecto AngulaJS realizando los siguientes pasos:

- Bien, ahora que hemos clonado el repositorio e instalado las dependencias, el esqueleto de nuestra aplicación va a tener este aspecto: el cual constituye de controller, model, view.
- En mi caso en un arranque de creatividad llamaré a mi proyecto “Almacén”

**Figura N° 1. 40 Creación de Proyecto**

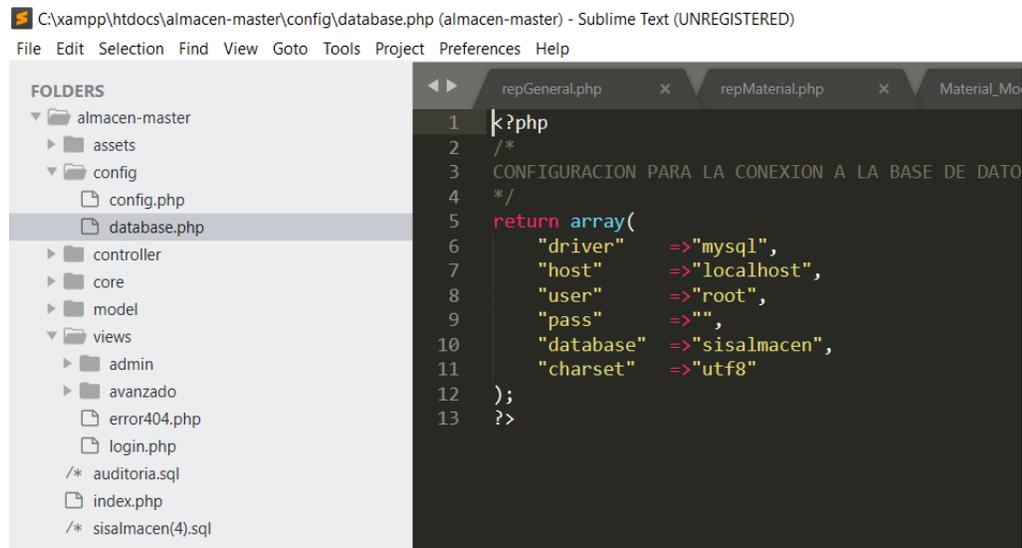


Fuente: (Elaboración Propia)

#### 4.1. CONFIGURAR LA BASE DE DATOS

C:\xampp\htdocs\almacen\sisalamacen\

**Figura N° 1. 41 Configuración de base de datos**



Fuente: (Elaboración Propia)

## 4.2 SERVIDOR ACTIVO

✓ Puerto: 80

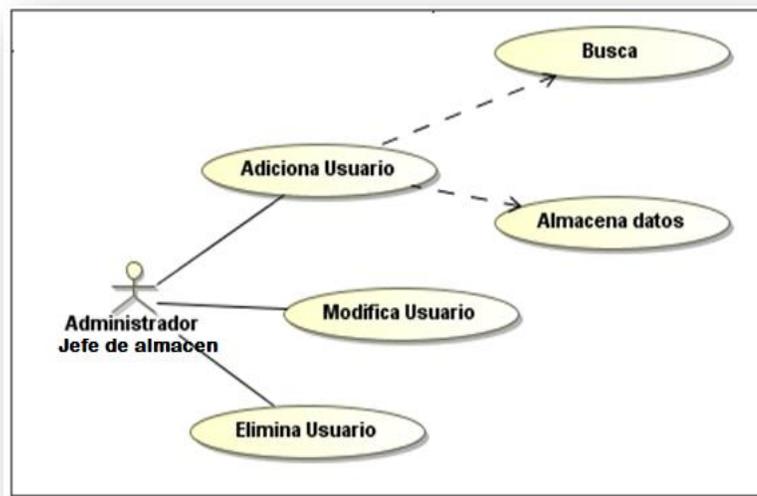
## 5. CASOS DE USO

Para el aplicativo se define dos roles o tipos de usuarios que interactuarán con el sistema de información de diferentes modos.

- ✚ Administrador: Jefe de Almacén
- ✚ Administrador Común: Funcionario

**Administrador:** tendrá la labor de ingresar, eliminar, modificar, copiar y realizar diferentes acciones sobre los registros de las tablas de la base de datos.

**Figura N° 1. 42 Diagrama Caso Uso-Administrador**



**Fuente:** (Elaboración Propia).

**Solicitantes:** el usuario al que le es asignado este rol tiene acceso Ingresado al módulo donde podrá registrar datos, consultar actualizaciones de datos previamente registrados y actualizar información.

**Tabla 2 Creación de registro de usuario**

<b>Nombre: Adicionar o Crear Usuarios</b>
Actor(es): Administrador
Descripción: Se crean cuentas de usuario para ingreso al sistema por su respectivo rol, directamente en la tabla usuarios
Precondiciones:
Flujo Normal: 1. El actor ingresa al módulo de administración. 2. Antes debe registrar personas con sus datos. 3. En la tabla usuarios crea un nuevo registro con los siguientes datos (nombre de funcionario, usuario y contraseña, tipo de ususario) 4. Se le asigna usuario y su password.
Flujo Alternativo: No aplica
Pos condición : 1. Cuenta creada

**Fuente:** (Elaboración Propia).

## 6 MODULO DE ADMINISTRACION

Fue creado con el fin de que facilitar el trabajo al administrador del aplicativo. A través de este módulo se podrán ver, editar, buscar y borrar registros de las tablas.

A continuación, describimos el procedimiento:

1. se ingresa al módulo de apertura al aplicativo

**Figura N° 1. 43 Pantalla de Inicio de Sesión**



**Fuente:** (Elaboración Propia).

Ingresar datos como el correo y el password, luego ingresa a la pantalla principal de Administrador.

El Alto, septiembre de 2020

Señor(a)  
**Ing. Carlos David Mamani Quispe**  
**DIRECTOR**  
**CARRERA INGENIERÍA DE SISTEMAS – UPEA**  
Presente.-

**REF.: AVAL DE CONFORMIDAD**

Distinguido ingeniero:

Tengo a bien dirigirme a su persona para comunicarle mi conformidad del trabajo final del Proyecto de Grado, titulado “**SISTEMA DE INFORMACIÓN PARA EL CONTROL DE INVENTARIO EN ALMACÉN**”, elaborado por la universitaria **Nolia Chamairo Espejo**, con cedula de identidad **8363830 L.P.** y **R.U.11005127:**, para su defensa pública y evaluación correspondiente a la materia de Taller de Licenciatura II, de acuerdo al reglamento vigente de la Carrera de Ingeniería de Sistemas de la Universidad Pública de El Alto.

Sin otro particular, me despido de usted.

Atentamente.



Ing. Marisol Arguedas Balladares  
TUTOR METODOLÓGICO

El Alto, Agosto de 2020

Señora:

**Ing. Marisol Arguedas Balladeras TUTOR METODOLÓGICO  
TALLER II**

Presente. –

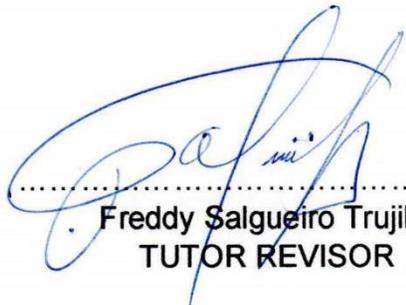
Ref.- **AVAL DE CONFORMIDAD PROYECTO DE GRADO**

Distinguido Ingeniero:

Mediante la presente tengo a bien comunicarle mi conformidad del proyecto de grado “SISTEMA DE INFORMACIÓN PARA EL CONTROL DE INVENTARIO EN ALMACÉN”, que propone la postulante Nolia Chamairo Espejo con CI: 8363830 LP., para su defensa pública y evaluación correspondiente a la materia de Taller de Licenciatura II de acuerdo a reglamento vigente de la carrera de Ingeniería de Sistemas de la Universidad Pública de El Alto.

Sin otro particular, reciba saludos cordiales.

Atentamente,



.....  
Freddy Salgueiro Trujillo  
TUTOR REVISOR

El Alto agosto de 2020

Señora:

**Ing. Marisol Arguedas Balladeras TUTOR METODOLÓGICO  
TALLER II**

Presente. –

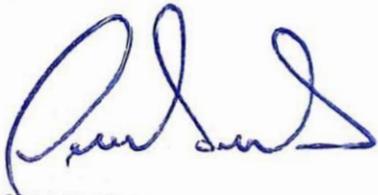
Ref.- **AVAL DE CONFORMIDAD PROYECTO DE GRADO**

Distinguido Ingeniero:

Mediante la presente tengo a bien comunicarle mi conformidad del proyecto de grado “SISTEMA DE INFORMACIÓN PARA EL CONTROL DE INVENTARIO EN ALMACÉN”, que propone la postulante Nolia Chamairo Espejo con CI: 8363830 LP., para su defensa pública y evaluación correspondiente a la materia de Taller de Licenciatura II de acuerdo a reglamento vigente de la carrera de Ingeniería de Sistemas de la Universidad Pública de El Alto.

Sin otro particular, reciba saludos cordiales.

Atentamente,



Ing. Edwin Mamani Viscarra  
**TUTOR ESPECIALISTA**



**ESCUELA SUPERIOR DE FORMACION DE MAESTROS  
TECNOLÓGICO Y HUMANÍSTICO EL ALTO**

*Fundado el 6 de marzo de 2006 por D.S. 28625 y Ley 3441*

El Alto, 27 de Agosto de 2020

Señora:  
**Ing. Marisol Arguedas Balladeras**  
**TUTOR METODOLÓGICO TALLER II**

Presente. –

Ref.- **AVAL DE CONFORMIDAD PROYECTO DE GRADO**

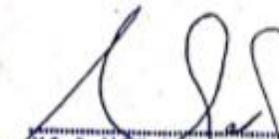
De mi muy Consideración:

Mediante la presente tengo a bien comunicarle mi conformidad del proyecto de grado titulado "SISTEMA DE INFORMACIÓN PARA EL CONTROL DE INVENTARIO EN ALMACÉN", que propone la postulante Nolia Chamairo Espejo con CI: 8363830 LP., para su defensa pública y evaluación correspondiente a la materia de Taller de Licenciatura II, de acuerdo a reglamento vigente de la carrera de Ingeniería de Sistemas de la Universidad Pública de El Alto.

Sin otro particular, reciba saludos cordiales.

Atentamente,



  
M.Sc. David Félix Cordero Arancibia  
DIRECTOR GENERAL  
ESPMTHEA