

# UNIVERSIDAD PÚBLICA DE EL ALTO

## CARRERA INGENIERÍA DE SISTEMAS



## PROYECTO DE GRADO

### “SISTEMA DE INFORMACIÓN WEB DE COSTOS DE PRODUCCIÓN Y DETERMINACIÓN DE UTILIDADES”

CASO: EMPRESA UNIPERSONAL DE ROPA DEPORTIVA KUBIK TEXTILES

Para Optar al Título de Licenciatura en Ingeniería de Sistemas  
MENCIÓN: INFORMÁTICA Y COMUNICACIONES

Postulante: Willian Alanoca Espejo

Tutor Metodológico: Lic. Ing. Helen Fanny Suntura Escobar

Tutor Especialista: Lic. Ing. Yolanda Escobar Mancilla

Tutor Revisor: Lic. Ing. Edwin Mamani Viscarra

EL ALTO – BOLIVIA

2024

## DECLARACIÓN JURADA DE AUTENTICIDAD Y RESPONSABILIDAD

Yo, **Willian Alanoca Espejo**, estudiante con **C.I. 9199444 LP** mediante la presente **declaro** de manera pública que la propuesta del **TRABAJO DE GRADO** titulada “**SISTEMA DE INFORMACIÓN WEB DE COSTOS DE PRODUCCIÓN Y DETERMINACIÓN DE UTILIDADES**” **CASO: EMPRESA UNIPERSONAL DE ROPA DEPORTIVA KUBIK TEXTILES** es original, siendo resultado de mi trabajo personal y no constituye una copia o replica de trabajos similares elaborados,

Autorizo la publicación del resumen de mi propuesta en internet y me comprometo a responder a todos los cuestionamientos que se desprenden de su lectura.

Asimismo, me hago responsable ante la universidad o terceros, de cualquiera irregularidad o daño que pudiera ocasionar, por el incumplimiento de lo declarado.

De identificarse falsificación, plagio, fraude, o que el **TRABAJO DE GRADO** haya sido publicado anteriormente; asumo las consecuencias y sanciones que de mi acción se deriven, responsabilizándome por todas las cargas legales que se deriven de ello sometiéndome a las normas establecidas y vigentes de la Carrera de Ingeniería de Sistemas de la Universidad Pública de El Alto.

El Alto, noviembre de 2024

Univ. Willian Alanoca Espejo  
C.I. 9199444 LP  
willianalanocaespejo@gmail.com

## DEDICATORIA

*Este proyecto de grado está dedicado:*

*A Dios, por ser el inspirador y darme fuerza para continuar este proceso y obtener uno de los anhelos más deseados.*

*A mis padres quienes con su amor han sabido guiarme, gracias a ustedes lo he logrado llegar hasta aquí y convertirnos en lo que soy.*

*Ha todas estas personas que nos ha apoyado y han hecho posible que pueda cumplir mis objetivos.*

### **AGRADECIMIENTOS**

*Gracias a Dios por permitirme tener a mi familia, gracias a ellos por apoyarme en todo en especial en mi etapa universitaria, agradezco a la universidad por formar parte de ella.*

*Agradezco también a mi tutor metodológico: Lic. Ing. Helen Fanny Suntura Escobar, tutor especialista Lic. Ing. Yolanda Escobar Mancilla, Tutor revisor Lic. Ing. Edwin Mamani Viscarra por haberme brindado la oportunidad de recurrir a su capacidad y conocimiento, así como también tenerme paciencia para guiarme en el desarrollo de mi proyecto.*

## RESUMEN

En la actualidad la empresa unipersonal "KUBIK Textiles" se controla la documentación interna y externa mediante unas planillas improvisadas en Excel lo cual la información generada no es del todo correcta y es limitado a falta de funciones del control rutinario que se hace. Por otro lado, la empresa no cuenta con un sistema de costos lo cual nos permita realizar reportes con datos correctos para la toma de decisiones.

Entre estos problemas se encuentran la ausencia de un sitio web, lo cual facilite el registro de los pedidos y ayude a calcular los costos incurridos en la producción, la falta de planillas sistematizados para registrar los pedidos lo que ocasiona: Demoras en la publicación de reportes, inconsistencias y errores en los datos, ineficiencia en la gestión de documentos.

Este proyecto tiene como objetivo desarrollar un Sistema Web moderno y optimizado para "Kubic Textiles," incorporando registro de costos de producción y potenciar oportunidades de negocio, generando un crecimiento significativo optimizando la producción de la empresa.

El presente proyecto de grado propone solucionar mediante certificación de profesionales el desarrollo de un " Sistema de información web de costos de producción ", a través de una metodología de la metodología UWE para el diseño y también la UML (Lenguaje de Modelado Unificado). En el proceso de evaluación de calidad de software, también se tomará en cuenta la ISO 9126, la estimación de costo del software se la realizará con COCOMO II basándonos en puntos de función, se busca no solo abordar los problemas identificados, sino también posicionar a la empresa en el epicentro de la innovación digital.

**Palabras Claves:** Costos, Producción, Cálculo, Sistematización, Reportes.

## ABSTRACT

Currently, the sole proprietorship "KUBIK Textiles" controls internal and external documentation using improvised spreadsheets in Excel; the information generated is not entirely correct and is limited due to the lack of routine control functions. On the other hand, the company does not have a cost system which allows us to make reports with correct data for decision making.

Among these problems are the absence of a website, which facilitates the registration of orders and helps calculate the costs incurred in production, the lack of systematized forms to record orders, which causes: Delays in the publication of reports, inconsistencies and errors in data, inefficiency in document management.

This project aims to develop a modern and optimized Web System for "Kubic Textiles," incorporating production cost registration and enhancing business opportunities, generating significant growth by optimizing the company's production.

This degree project proposes to solve the development of a "production cost web information system" through professional certification, through a methodology of the UWE methodology for design and also the UML (Unified Modeling Language). the software quality evaluation process, ISO 9126 will also be taken into account, the software cost estimate will be carried out with COCOMO II based on function points, the aim is not only to address the identified problems, but also to position the company in the epicenter of digital innovation.

Keywords: Costs, Production, Calculation, Systematization, Reports.

## CONTENIDO

<b>1</b>	<b>CAPÍTULO I MARCO PRELIMINAR</b> .....	<b>1</b>
1.1	<b>INTRODUCCIÓN</b> .....	<b>1</b>
1.2	<b>ANTECEDENTES</b> .....	<b>2</b>
1.2.1	Antecedentes Institucionales.....	2
1.2.1.1	Visión .....	2
1.2.1.2	Misión .....	2
1.2.1.3	Organigrama.....	3
1.2.2	Antecedentes Afines al Proyecto de Grado .....	3
1.2.2.1	Antecedentes Internacionales.....	3
1.2.2.2	Antecedentes Nacionales .....	4
1.2.2.3	Antecedentes Locales .....	5
<b>1.3</b>	<b>PLANTEAMIENTO DEL PROBLEMA</b> .....	<b>5</b>
1.3.1	Problema Principal.....	5
1.3.2	Problemas Secundarios .....	6
1.3.3	Formulación del Problema.....	6
<b>1.4</b>	<b>OBJETIVOS</b> .....	<b>6</b>
1.4.1	Objetivo General.....	6
1.4.2	Objetivos Específicos .....	6
<b>1.5</b>	<b>JUSTIFICACIÓN</b> .....	<b>7</b>
1.5.1	Justificación Técnica.....	7
1.5.2	Justificación Económica .....	7
1.5.3	Justificación Social .....	8
<b>1.6</b>	<b>METODOLOGÍA DE DESARROLLO</b> .....	<b>8</b>
1.6.1	Método Costos de Producción .....	9

1.6.1.1	Costos Directos e Indirectos De Producción .....	9
1.6.1.2	Costos Fijos y Variables .....	10
1.6.2	Métricas de Calidad .....	11
1.6.3	Costos .....	12
1.6.4	Seguridad ISO/IEC 25001 .....	12
1.6.5	Pruebas de Software .....	13
1.6.5.1	Pruebas de Caja Blanca .....	13
1.6.5.2	Pruebas de Caja Negra.....	13
1.6.5.3	Pruebas de Estrés.....	14
1.6.5.4	Pruebas de Unitarias.....	14
<b>1.7</b>	<b>HERRAMIENTAS .....</b>	<b>15</b>
1.7.1	Apache (servidor Web HTTP) .....	15
1.7.2	CodeIgniter (Framework).....	15
1.7.3	Visual Studio Code (VS Code).....	15
1.7.4	Lenguaje de Programación .....	16
1.7.5	Gestor de Base de Datos .....	16
<b>1.8</b>	<b>ALCANCES Y LIMITES .....</b>	<b>16</b>
1.8.1	Alcances.....	16
1.8.2	Limites.....	17
<b>1.9</b>	<b>APORTES.....</b>	<b>18</b>
<b>2</b>	<b>CAPÍTULO II MARCO TEÓRICO .....</b>	<b>20</b>
<b>2.1</b>	<b>INTRODUCCIÓN .....</b>	<b>20</b>
<b>2.2</b>	<b>SISTEMA .....</b>	<b>20</b>
<b>2.3</b>	<b>SISTEMA DE INFORMACIÓN .....</b>	<b>20</b>
<b>2.4</b>	<b>WEB .....</b>	<b>21</b>

2.4.1	Web 2.0.....	22
2.4.2	Diferencias.....	22
<b>2.5</b>	<b>COSTOS DE PRODUCCIÓN.....</b>	<b>22</b>
2.5.1	Costo.....	23
2.5.2	Tipos de Costos.....	23
2.5.3	Costo y Gasto.....	25
2.5.4	Utilidades.....	25
<b>2.6</b>	<b>METODOLOGÍA WEB BASADA EN UML(UWE).....</b>	<b>25</b>
2.6.1	Metodología UWE (UML-Based Web Engineering).....	26
2.6.2	Características de UWE.....	26
2.6.3	Modelos de la Metodología UWE.....	27
2.6.3.1	Modelo de Requisitos “Casos de Uso”.....	27
2.6.3.2	Modelo de Contenido “Conceptual”.....	28
2.6.3.3	Modelo de Navegación.....	29
2.6.3.4	Modelo de Presentación.....	29
2.6.3.5	Modelo de Proceso.....	31
2.6.4	Fases de la Metodología UWE.....	32
2.6.4.1	Captura, Análisis y Especificación de Requisitos.....	32
2.6.4.2	Diseño del Sistema.....	33
2.6.4.3	Codificación del Software.....	33
2.6.4.4	Pruebas.....	33
2.6.4.5	La Instalación o Fase de Implementación.....	33
2.6.4.6	El Mantenimiento:.....	33
<b>2.7</b>	<b>ARQUITECTURA DE SOFTWARE (MVC).....</b>	<b>34</b>

2.7.1	Modelo .....	34
2.7.2	Vista.....	35
2.7.3	Controlador .....	36
<b>2.8</b>	<b>HERRAMIENTAS .....</b>	<b>37</b>
2.8.1	PHP (Hipertext Pre-processor).....	37
2.8.2	Apache.....	38
2.8.3	Gestor de Base de Datos .....	39
2.8.4	Bootstrap.....	39
2.8.5	Visual Studio Code .....	39
2.8.6	Framework CodeIgniter .....	40
<b>2.9</b>	<b>ESTIMACIÓN DE COSTOS .....</b>	<b>41</b>
2.9.1	Cocomo II .....	41
2.9.1.1	Ecuaciones del Modelo COCOMO .....	42
2.9.1.2	Modelos de COCOMO II .....	44
2.9.1.3	Ventajas de COCOMO II.....	46
<b>2.10</b>	<b>MÉTRICAS DE CALIDAD.....</b>	<b>46</b>
2.10.1	ISO/IEC 25010.....	47
2.10.1.1	Funcionalidad .....	47
2.10.1.2	Confiabilidad.....	50
2.10.1.3	Usabilidad.....	52
2.10.1.4	Eficiencia .....	52
2.10.1.5	Mantenibilidad .....	53
2.10.1.6	Portabilidad.....	53
<b>2.11</b>	<b>SEGURIDAD.....</b>	<b>54</b>
2.11.1	ISO 27001.....	55

2.11.2 Seguridad Física.....	56
2.11.3 Seguridad Lógica.....	57
2.11.4 Seguridad de la Base de Datos.....	57
2.11.5 Seguridad del Framework CodeIgniter.....	58
<b>2.12 PRUEBAS DE SOFTWARE.....</b>	<b>58</b>
2.12.1 Pruebas de Caja Blanca.....	59
2.12.2 Pruebas de Caja Negra.....	59
2.12.3 Pruebas de Estrés.....	59
2.12.4 Pruebas de Accesibilidad.....	60
2.12.5 Pruebas de Unitarias.....	61
<b>3 CAPÍTULO III MARCO APLICATIVO.....</b>	<b>63</b>
<b>3.1 INTRODUCCIÓN.....</b>	<b>63</b>
<b>3.2 DESARROLLO DE LA METODOLOGÍA.....</b>	<b>63</b>
3.2.1 Requerimientos Funcionales.....	64
3.2.2 Requerimientos No Funcionales.....	65
3.2.3 Descripción de Actores.....	66
3.2.4 Diagramas de Caso de Uso General.....	67
3.2.5 Diagrama de Caso de Uso Específico.....	68
3.2.6 Modelo Conceptual.....	80
3.2.7 Modelo de Presentación.....	81
3.2.8 Modelo de Proceso.....	88
<b>4 CAPÍTULO IV CALIDAD, SEGURIDAD Y COSTOS.....</b>	<b>99</b>
<b>4.1 INTRODUCCIÓN.....</b>	<b>99</b>
<b>4.2 MÉTRICAS DE CALIDAD - ESTÁNDAR ISO/IEC 25000.....</b>	<b>99</b>
4.2.1 Funcionalidad.....	99
4.2.1.1 Punto Función.....	100

4.2.2	Confiabilidad .....	105
4.2.3	Usabilidad .....	107
4.2.4	Eficiencia.....	109
4.2.5	Mantenibilidad.....	111
4.2.6	Portabilidad.....	112
<b>4.3</b>	<b>SEGURIDAD DEL SISTEMA – ISO 27001.....</b>	<b>114</b>
4.3.1	Identificación y Autenticación. ....	114
4.3.2	Encriptación .....	115
4.3.3	Seguridad Física.....	116
4.3.4	Seguridad Organizativa .....	116
<b>4.4</b>	<b>PRUEBAS DE SOFTWARE.....</b>	<b>116</b>
4.4.1	Prueba de Caja Blanca.....	116
4.4.2	Prueba de Caja Negra.....	119
4.4.3	Pruebas de Estrés .....	124
4.4.4	Pruebas de Accesibilidad .....	126
4.4.5	Pruebas Unitarias .....	127
<b>4.5</b>	<b>ESTIMACIÓN DE COSTO COCOMO II.....</b>	<b>128</b>
4.5.1	Modelo Orgánico .....	129
4.5.2	Modelo Semi Acoplado.....	129
4.5.3	Modelo Empotrado .....	129
4.5.4	Esfuerzo Requerido del Desarrollo .....	133
4.5.5	Tiempo de Desarrollo Requerido del Sistema .....	134
4.5.6	Número de Desarrolladores .....	135
4.5.7	Costo Total .....	136
<b>5</b>	<b>CAPÍTULO V CONCLUSIONES Y RECOMENDACIONES .....</b>	<b>139</b>
<b>5.1</b>	<b>CONCLUSIONES.....</b>	<b>139</b>
<b>5.2</b>	<b>RECOMENDACIONES.....</b>	<b>140</b>

**ANEXOS..... 145**

**ÍNDICE DE FIGURAS**

<b>Figura 1</b> Organigrama de la Empresa KUBIK.....	3
<b>Figura 2</b> Modelo de Casos de Uso.....	28
<b>Figura 3</b> Modelo de Contenido .....	29
<b>Figura 4</b> Modelo de Navegación .....	30
<b>Figura 5</b> Modelo de Presentación .....	32
<b>Figura 6</b> Modelo Vista Controlador.....	36
<b>Figura 7</b> Arquitectura de la Aplicación.....	40
<b>Figura 8</b> ISO/IEC 25010 .....	47
<b>Figura 9</b> Funciones de la Pruebas Unitarias.....	61
<b>Figura 10</b> Caso de Uso General del Sistema .....	67
<b>Figura 11</b> Gestión de Usuarios.....	68
<b>Figura 12</b> Gestión de Roles y Permisos.....	69
<b>Figura 13</b> Gestión de Clientes.....	70
<b>Figura 14</b> Gestión de Pedidos.....	71
<b>Figura 15</b> Gestión Proveedor .....	72
<b>Figura 16</b> Gestión Clasificación Costos .....	73
<b>Figura 17</b> Gestión de Producción.....	74
<b>Figura 18</b> Gestión de Utilidades .....	75
<b>Figura 19</b> Gestión Historial de Documentos .....	76
<b>Figura 20</b> Gestión de Almacén de Pedidos.....	77
<b>Figura 21</b> Gestión de Libro Diario .....	78
<b>Figura 22</b> Gestión de Datos Estadísticos.....	79
<b>Figura 23</b> Diseño de la Base de Datos Relacional .....	80
<b>Figura 24</b> Modelo de Presentación - Login .....	81
<b>Figura 25</b> Modelo de Presentación - Administrador.....	81

<b>Figura 26</b> Modelo de Presentación - Usuarios.....	82
<b>Figura 27</b> Modelo de Presentación - Roles y Permisos.....	82
<b>Figura 28</b> Modelo de Presentación - Clientes.....	83
<b>Figura 29</b> Modelo de Presentación - Pedidos.....	83
<b>Figura 30</b> Modelo de Presentación - Registro del Proveedor.....	84
<b>Figura 31</b> Modelo de Presentación – Clasificación de Costos .....	84
<b>Figura 32</b> Gestión de Producción.....	85
<b>Figura 33</b> Modelo de Presentación - Formulario de Determinación de Utilidades.....	85
<b>Figura 34</b> Gestión de Documentos.....	86
<b>Figura 35</b> Modelo de Presentación – Registro de Almacén de Pedidos .....	86
<b>Figura 36</b> Modelo de Presentación – Gestión Libro Diario.....	87
<b>Figura 37</b> Modelo de Presentación – Datos Estadísticos.....	87
<b>Figura 38</b> Página Principal .....	88
<b>Figura 39</b> Catálogos Productos.....	88
<b>Figura 40</b> Datos de la Empresa.....	89
<b>Figura 41</b> Empresa KUBIK TEXTILES.....	89
<b>Figura 42</b> Inicio de Sesión del Sistema.....	90
<b>Figura 43</b> Ingreso al Sistema - Administrador.....	90
<b>Figura 44</b> Listado de Usuarios .....	91
<b>Figura 45</b> Listado de Roles y Permisos .....	91
<b>Figura 46</b> Administración de Clientes.....	92
<b>Figura 47</b> Registro de Clientes.....	92
<b>Figura 48</b> Administración de Pedidos.....	93
<b>Figura 49</b> Formulario Registro de Pedidos .....	93
<b>Figura 50</b> Gestión de Proveedor .....	94
<b>Figura 51</b> Formulario de Clasificación de Costos .....	94

<b>Figura 52</b> Gestión de Producción.....	94
<b>Figura 53</b> Gestión de Utilidades.....	95
<b>Figura 54</b> Comportamiento de Costos.....	95
<b>Figura 55</b> Gestión de Documentos.....	96
<b>Figura 56</b> Gestión de Almacén de Pedidos.....	96
<b>Figura 57</b> Modelo de Presentación – Libro Diario.....	97
<b>Figura 58</b> Gestión de Datos Estadísticos.....	97
<b>Figura 59</b> Identificación y Autenticación.....	115
<b>Figura 60</b> Seguridad de Password.....	115
<b>Figura 61</b> Diagrama del Sistema.....	117
<b>Figura 62</b> Grafo de Flujo del Sistema.....	118
<b>Figura 63</b> Prueba de Caja Negra Iniciar Sesión.....	120
<b>Figura 64</b> Caja Negra Ingreso al Sistema.....	121
<b>Figura 65</b> Prueba de Caja Negra – Agregar Cliente.....	122
<b>Figura 66</b> Prueba de Caja Negra – Registro de cliente.....	123
<b>Figura 67</b> Validación de Campos en el Formulario.....	124
<b>Figura 68</b> Evaluando Pruebas de Estrés.....	125
<b>Figura 69</b> Prueba de Accesibilidad.....	126
<b>Figura 70</b> Resultado del Testeo.....	127
<b>Figura 71</b> Prueba Unitaria.....	128

**ÍNDICE DE TABLAS**

<b>Tabla 1</b> Tabla de Valores de Modelo Básico.....	44
<b>Tabla 2</b> Indices de Usabilidad.....	48
<b>Tabla 3</b> Herramientas para la Obtención de Requerimientos .....	63
<b>Tabla 4</b> Requerimientos Funcionales .....	64
<b>Tabla 5</b> Requerimientos No Funcionales.....	65
<b>Tabla 6</b> Descripción de Actores .....	66
<b>Tabla 7</b> Detalle del Caso de Uso – Gestión de Usuarios.....	68
<b>Tabla 8</b> Detalle del Caso de Uso – Gestión de Roles y Permisos .....	69
<b>Tabla 9</b> Detalle del Caso de Uso – Gestión de Clientes .....	70
<b>Tabla 10</b> Detalle del Caso de Uso – Gestión de Pedidos .....	71
<b>Tabla 11</b> Detalle de Caso de Uso - Gestión Proveedor .....	72
<b>Tabla 12</b> Detalle de Caso de Uso – Clasificación de Costos.....	73
<b>Tabla 13</b> Detalle de Caso de Uso – Gestión Producción.....	74
<b>Tabla 14</b> Detalle de Caso de Uso - Gestión de Utilidades.....	75
<b>Tabla 15</b> Detalle de Caso de Uso – Gestión Historial de Documentos .....	76
<b>Tabla 16</b> Detalle de Caso de Uso - Gestión de Almacén de Pedidos.....	77
<b>Tabla 17</b> Detalle de Caso de Uso - Gestión Libro Diario .....	78
<b>Tabla 18</b> Detalle de Caso de Uso - Gestión de Datos Estadísticos.....	79
<b>Tabla 19</b> Número de Usuarios del Sistema .....	100
<b>Tabla 20</b> Número de Salidas de los Usuarios .....	101
<b>Tabla 21</b> Número de Peticiones.....	101
<b>Tabla 22</b> Parámetros de Medición .....	102
<b>Tabla 23</b> Indicé de Usabilidad.....	103
<b>Tabla 24</b> Indicadores de Facilidades de Uso.....	107

<b>Tabla 25</b> Usabilidad del Sistema .....	108
<b>Tabla 26</b> Escala de Valores de Eficiencia.....	109
<b>Tabla 27</b> Evaluación de Eficiencia .....	110
<b>Tabla 28</b> Indicadores de Mantenibilidad .....	111
<b>Tabla 29</b> Resultados Obtenidos - Norma ISO 25000 .....	113
<b>Tabla 30</b> Valores de Caja Negra Login .....	120
<b>Tabla 31</b> Prueba de Caja Negra Ingreso el Sistema .....	121
<b>Tabla 32</b> Valores Limite - Registrar Pedido .....	122
<b>Tabla 33</b> Prueba de Caja Negra - Registrar pedido .....	123
<b>Tabla 34</b> Resultado de la Prueba de Accesibilidad .....	127
<b>Tabla 35</b> Complejidad y Eficiencia del Lenguaje.....	130
<b>Tabla 36</b> Tabla de Atributos Factores del Esfuerzo .....	132
<b>Tabla 37</b> Coeficientes Cocomo II – Modo Orgánico .....	133

## ÍNDICE DE ECUACIONES

( 1 ) Ecuación de Esfuerzo .....	42
( 2 ) Ecuación Tiempo de Esfuerzo .....	43
( 3 ) Ecuación Número de Personas .....	43
( 4 ) Ecuación Personas Por Mes Según Modelo .....	44
( 5 ) Ecuación Tiempo de Proyecto Según Modelo .....	44
( 6 ) Ecuación Costo Personal .....	45
( 7 ) Ecuación de Costo Total del Proyecto .....	45
( 8 ) Fórmula de Ajuste .....	49
( 9 ) Punto Función Máxima .....	50
( 10 ) Formula de la Funcionalidad .....	50
( 11 ) Probabilidad de Fallas .....	51
( 12 ) Probabilidad de Trabajo Sin Fallas .....	51
( 13 ) Función de la Confiabilidad del Sistema MT .....	51
( 14 ) Función para Determinar la Usabilidad MT .....	52
( 15 ) Fórmula de la Eficiencia MT .....	52
( 16 ) Cálculo del Nivel de Madurez del Sistema .....	53
( 17 ) Fórmula de la Portabilidad MT .....	54
( 18 ) Ecuación de Ajuste .....	104
( 19 ) Función de la Confiabilidad del Sistema .....	105
( 20 ) Función para Determinar la Usabilidad .....	107
( 21 ) Fórmula Eficiencia .....	110
( 22 ) Ecuación para el Índice de Madurez del Sistema .....	111
( 23 ) Fórmula Portabilidad .....	113
( 24 ) Ecuación Complejidad Ciclomática .....	119

( 25 ) Ecuación Líneas de Código.....	130
( 26 ) Ecuación Esfuerzo .....	134
( 27 ) Ecuación Tiempo Requerido.....	134
( 28 ) Ecuación Número de Desarrolladores .....	135
( 29 ) Ecuación Costo del Software.....	136

## LISTADO DE ABREVIATURAS

**FLDC:** Factor de conversión de líneas de código entregado.

**TDES:** Tiempo empleado en el desarrollo.

**PF:** Puntos de función.

**PHP:** Preprocesador de Hipertexto.

**UWE:** Ingeniería Web basada en UML.

**UML:** Lenguaje de Modelado Unificado.

**HTML:** Lenguaje de Marcado de Hipertexto.

**HTTP:** Protocolo de Transferencia de Hipertexto.

**KLDC:** Líneas de Código.

**IDE:** Entorno de Desarrollo Integrado

**MVC:** Modelo Vista Controlador

**ISO:** Organización Internacional de Normalización

**IEC:** Comisión Electrotécnica Internacional

**RNF:** Requerimientos No Funcionales

**FAE:** Factor de Ajuste de Esfuerzo

# **CAPÍTULO I**

## **MARCO**

### **PRELIMINAR**

**INGENIERÍA**  
**DE SISTEMAS**  
UNIVERSIDAD PÚBLICA DE EL ALTO



## 1 CAPÍTULO I MARCO PRELIMINAR

### 1.1 INTRODUCCIÓN

En la actualidad, las empresas grandes medianas y pequeñas han comprendido la importancia de implementar sistemas de apoyo contable de costos que sirva como soporte para la toma de decisiones y al mismo tiempo permita analizar procesos de producción. Tener el control sobre los datos y la información de la empresa unipersonal por medio de un sistema organizado que permita controlar los costos de producción en la elaboración de ropa deportiva.

La empresa unipersonal "KUBIK Textiles" se controla la documentación interna y externa mediante unas planillas improvisadas en Excel lo cual la información generada no es del todo correcta y es limitado a falta de funciones del control rutinario que se hace. Por otro lado, la empresa no cuenta con un sistema contable de costos lo cual nos permita realizar reportes con datos correctos para la toma de decisiones

Este proyecto de grado propone solucionar mediante certificación de profesionales el desarrollo de un " Sistema de información web de costos de producción ", a través de una metodología de la metodología UWE para el diseño y también la UML (Lenguaje de Modelado Unificado). En el proceso de evaluación de calidad de software, también se tomará en cuenta la ISO 9126, la estimación de costo del software se la realizará con COCOMO II basándonos en puntos de función, se busca no solo abordar los problemas identificados, sino también posicionar a la empresa en el epicentro de la innovación digital.

Desde el análisis de la situación actual hasta la implementación de estrategias de contenido y diseño optimizados, este proyecto tiene como propósito no solo superar los desafíos existentes, sino también se pretende catapultar a la empresa "kubic textiles" como líder en la producción de prendas deportivas.

## 1.2 ANTECEDENTES

A continuación, se describen los antecedentes institucionales:

### 1.2.1 Antecedentes Institucionales

La empresa unipersonal “KUBIK Textiles” dedicada al rubro de la comercialización y elaboración de ropas deportivas tiene como función principal brindar a su distinguida clientela variedad de diseños y modelos.

- Actividad Primaria Fabricación De Prendas Deportivas
- Actividad Especifica Fabricación de prendas deportivas, excepto prendas de piel.
- Actividad venta de ropa deportiva y otra confección de ropa deportiva en general bordados computarizados

#### 1.2.1.1 Visión

La empresa unipersonal “KUBIK Textiles” tiene la misión de brindar una buena atención ofreciendo variedad de productos en alta costura diseños exclusivos y novedosos.

#### 1.2.1.2 Misión

Consolidarnos como una empresa sólida y líder en el rubro de elaboración de ropas deportivas ofreciendo productos de calidad al 100% garantizado.

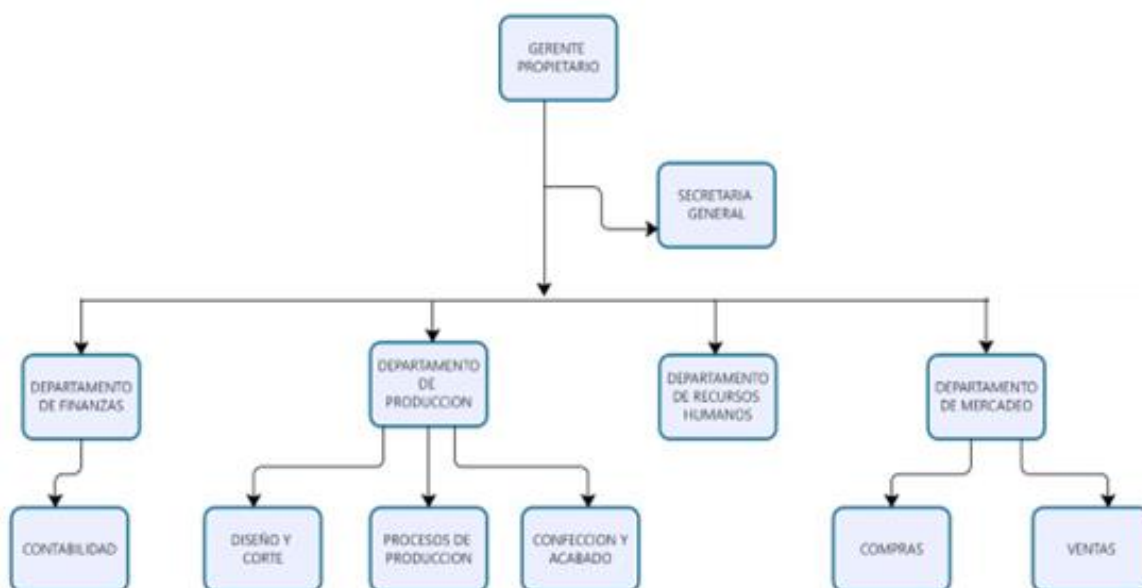
<b>Razón Social:</b>	<b>KUBIK TEXTILES</b>
<b>Gerente Propietario:</b>	Eustaquio Charles Chirinos Aduviri
<b>Número de NIT:</b>	3398469016
<b>Tipo Societario:</b>	Empresa Unipersonal
<b>Licencia de Funcionamiento:</b>	Información no registrada
<b>Departamento:</b>	La Paz

<b>Municipio:</b>	El Alto
<b>Dirección:</b>	Z/ Villa Bolívar "C" Calle Alonzo Yañes Nro. 5464
<b>Número de teléfono(s):</b>	72514129

### 1.2.1.3 Organigrama

**Figura 1**

*Organigrama de la Empresa KUBIK*



**Nota.** *Organigrama de la empresa unipersonal, 2010, Empresa KUBIK TEXTILES.*

## 1.2.2 Antecedentes Afines al Proyecto de Grado

Haciendo referencia a trabajos realizados a nivel internacional y nacional se puede citar:

### 1.2.2.1 Antecedentes Internacionales

Según Ramírez (2019), "Sistemas web para controlar y gestionar la producción de banano. Implementación de un sistema web de los procesos de ventas y gestión de inventarios en una empresa de calzado". Los Framework de desarrollo en CSS facilitan las tareas de

diseño y estilización de las páginas web. Los CSS son mecanismos simples que describen como se debe mostrar un documento web en la pantalla. Son utilizadas para dar estilo a los documentos HTML y XML separando el contenido de la presentación.

Según Angulo (2021), “Implementación de un sistema web para la gestión de ventas e inventario de una empresa de calzado”, Determinar en qué medida la implementación de un sistema web optimízalos tiempos, la cantidad de reportes disponibles y de tareas manuales de los procesos de ventas y gestión de inventarios en una empresa de calzado. Las herramientas que utilizo es PHP con un motor de base de datos MYSQL. Finalmente, a través de la implementación del sistema web propuesto se demostró la mejora de los procesos principales con la reducción de tiempos, simplificación de tareas y aumento de reportes disponibles para la toma de decisiones.

### **1.2.2.2 Antecedentes Nacionales**

Según Hilaya (2021), “Empresa elaboradora comercializadora de ropa deportiva entela de fibra de bambú “textilbam””, El presente plan de negocios denominado “Textilbam” se constituirá en una empresa de responsabilidad limitada, dirigida principalmente a la población de la ciudad de El Alto con el fin de ofrecer una nueva alternativa en el mercado de prendas deportivas realizadas únicamente en tela de fibra de Bambú, convirtiéndose en prendas 100% biodegradables, ideal para realizar alguna actividad física o deportiva debido a que absorbe la humedad que se produce por la sudoración comprendiendo agentes antibacteriales pero sobre todo se constituyen en prendas que proporcionan suavidad y comodidad de forma natural es por ellos que se la denomina al Bambú la planta de los mil usos.

### **1.2.2.3 Antecedentes Locales**

Para Suri (2020), “Sistema web para la gestión de producción, pedidos y entregas a través de código Qr”. Implementar un sistema web para la gestión de producción, pedidos y entrega a través de código QR en la empresa de fábrica de calaminas aceros Isacar S.R.L. y tener una buena planificación en la producción de calaminas. Las herramientas para el desarrollo del sistema web para la gestión de producción, pedidos y entrega a través de código QR se usarán: Gestor de base de datos MariaDB, Lenguaje de desarrollo PHP, Servidor Web Apache, Vue.Js y Vuetify .

## **1.3 PLANTEAMIENTO DEL PROBLEMA**

La empresa unipersonal **KUBIK Textiles** dedicado a la confección de prendas deportivas da a conocer sus productos y servicios en general brindando confianza y fidelidad en sus clientes, aumentando sus pedidos y beneficios. Sin embargo, presenta varios problemas que afectan su rendimiento y competitividad. Entre estos problemas se encuentran la ausencia de un sitio web, lo cual facilite el registro de los pedidos y ayude a calcular los costos incurridos en la producción, la falta de planillas sistematizados para registrar los pedidos. Actualmente, todos los registros y reportes se realizan de en tablas y planillas improvisadas, lo que ocasiona: Demoras en la publicación de reportes, inconsistencias y errores en los datos, ineficiencia en la gestión de documentos.

### **1.3.1 Problema Principal**

La empresa KUBIC TEXTILES unipersonal se enfrenta a desafíos en los costos de producción de ropas deportivas, deriva en deficiencias que afectan la efectividad y la precisión de las operaciones de la empresa.

### **1.3.2 Problemas Secundarios**

- La ausencia de registros sistematizados y desactualizados en la empresa "kubic textiles", enfrenta desafíos al momento de proporcionar informes de los procesos de producción a los clientes.
- La inexistencia de un sistema web costos de producción, deriva en retrasos y cálculos erróneos en los reportes diarios, especialmente en el área administrativa.
- La falta de coherencia en el llenado de los pedidos, distorsiona la percepción que los clientes potenciales tienen al momento de hacer su cotización.
- La deficiente manipulación de costos manuales, reduce la confianza y percepción que tienen los clientes, elevando el riesgo a que obtén por otras opciones.
- Desorganización en el departamento de producción en cuanto a las planillas de costos, lo que se traduce en búsquedas demorosas de la información requerida.

### **1.3.3 Formulación del Problema**

¿De qué manera se optimizaría los costos de producción en la empresa "KUBIC TEXTILES", a fin de obtener exactitud en los resultados de utilidades en la misma?

## **1.4 OBJETIVOS**

### **1.4.1 Objetivo General**

Desarrollar un sistema de información web de costos de producción empresa unipersonal de ropa deportiva "kubik textiles" con el fin de facilitar los cálculos, reportes, e informes generando así precisión en la determinación de utilidades en cuanto a costos se refiere, para un eficiente control.

### **1.4.2 Objetivos Específicos**

- Realizar pruebas exhaustivas en los formularios requeridos para el registro y control de cada uno de los factores productivos que intervienen en el sistema de costo.

- Optimizar los datos estadísticos de los registros de pedidos y movimientos económicos en “kubic Textiles”
- Reemplazar las planillas de registro manual por registros sistematizados optimizados y específicos para el área de producción.
- Aplicar un módulo para generar reportes que contenga los datos de los costos de producción.
- Implementar un Sistema Web responsivo que se adapta a diferentes tamaños y tipos de pantalla.

## **1.5 JUSTIFICACIÓN**

### **1.5.1 Justificación Técnica**

El proyecto se justifica técnicamente porque para su desarrollo se utilizará tecnologías de última generación asimismo se dispone de equipos de computación en sus diferentes unidades administrativas. La empresa unipersonal “KUBIK Textiles” dispone de:

Una computadora con procesador de Intel Core i5 de 3.30 GHz, 4G de RAM, almacenamiento de 500 Gb. Una impresora Epson M105 de alto tráfico.

La elaboración del presente proyecto se justifica técnicamente por lo que “KUBIK Textiles” cuenta con equipos de computación y está dispuesta a adquirir equipos necesarios para implementar el sistema web para la gestión de producción y pedidos.

### **1.5.2 Justificación Económica**

La elaboración del presente proyecto de gestión de costos de producción y pedidos de la empresa unipersonal “KUBIK Textiles” está dispuesto a invertir en los que se requiera para la implementación del proyecto y su mantenimiento para lo cual cuenta con los recursos

necesarios para dicha implementación. El proyecto será desarrollado con herramientas de software libre<sup>1</sup> por lo cual este proyecto reducirá el costo de desarrollo.

### **1.5.3 Justificación Social**

La implementación del sistema web automatizara todos los procesos manuales, brindando datos actualizados para el beneficio de las actividades diarias que realizan en las áreas de gerencia, producción, almacenes y ventas para tomar decisiones óptimas. Permitirá que los clientes aumenten su nivel de confianza con la microempresa que ofrece sus productos al público en general.

Con el desarrollo del proyecto se beneficiará al propietario, él señor Eustaquio Charles Chirinos Aduviri, y a los usuarios en general, puesto que agilizará el proceso de determinar los costos en los distintos procesos de producción, como herramienta útil para el departamento de producción y administrativo.

## **1.6 METODOLOGÍA DE DESARROLLO**

Para el desarrollo del sistema se utilizará la metodología UWE (Ingeniería Web Basada en UML) nació a finales de la década de los 90 con la idea de encontrar una forma estándar para analizar y diseñar modelos de sistemas web. El objetivo por el cual nació esta metodología fue utilizar un lenguaje común o por lo menos definir un metamodelo basado en el mapeo a lo largo de las diferentes etapas. En esa época UML prometía convertirse en un estándar para el modelamiento de sistemas. Por este motivo, UWE se adhirió a UML y no a otra técnica de modelado. UWE se ha adaptado a las nuevas características de los sistemas web como transacciones, personalizaciones y aplicaciones asíncronas, y por otro lado ha evolucionado para incorporar técnicas de ingeniería de software como el modelamiento orientado a aspectos y nuevos lenguajes de transformación para mejorar la calidad del diseño.

---

<sup>1</sup> «Software libre» significa que el software respeta la libertad de los usuarios y la comunidad.

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 sistemática, basada en las técnicas de UML, la notación de UML y los mecanismos de extensión de UML.

Las fases de la metodología UWE cubre todo el ciclo de vida de este tipo de aplicaciones centrandose además su atención en aplicaciones personalizadas o adaptativas (Evangellys, 2013, p. 5).

- Captura, análisis y especificación de requisitos
- Diseño del sistema
- Codificación del software
- Pruebas
- La Instalación o Fase de Implementación
- El Mantenimiento

### **1.6.1 Método Costos de Producción**

El método costo de la producción se refiere a todos los elementos que participan en la valuación del producto terminado durante el proceso productivo. Uno de estos elementos es el método de costo de la producción. El método de costo se refiere a la estrategia mediante la cual se suman los costos en el producto terminado.

#### **1.6.1.1 Costos Directos e Indirectos De Producción**

Los costos directos tienen una relación mucho más directa con el producto, servicio o actividad de la empresa que los costos indirectos. Este es el principal asgo distintivo entre ambos tipos de gastos.

Por lo tanto, los costos indirectos tienen una conexión más indirecta o tangencial con

las actividades de la empresa.

Los costos directos se establecen en las primeras fases de la producción, no así los indirectos.

Por otro lado, los costos directos casi **siempre** se reflejan en los presupuestos previos o estimaciones de gastos de la empresa, mientras que los indirectos (erróneamente) se suelen obviar hasta que el proyecto empresarial ya está en marcha.

### **1.6.1.2 Costos Fijos y Variables**

#### **¿Qué es costo fijo?**

El costo fijo son todos los pagos mínimos invariables e imprescindibles que debe hacer una empresa o negocio periódicamente para continuar funcionando. Estos costos siempre se deben pagar, independientemente del nivel de producción que tenga el negocio.

Sin embargo, aunque imprescindibles, los costos fijos no guardan una relación directa con la producción de la empresa en sí mismos, es decir, estos no son susceptibles de fluctuar según se incremente o disminuya la productividad.

#### **¿Qué es costo variable?**

Como su nombre lo indica, los costos variables son aquellos pagos que dependen directamente del nivel de productividad de la empresa o negocio, son directamente proporcionales, es decir: a mayor productividad, mayor costo variable, y a menor productividad, menor costo variable.

Entre los costos variables de una empresa o negocio podemos encontrar:

**Materia prima:** si la producción aumenta, es necesario comprar más cantidad de materia prima, pero si baja, se reduce también la compra.

**Comisiones sobre ventas:** en este caso si hay más ventas aumentan las comisiones de los empleados, pero si hay menos, baja el número de comisiones que se deben pagar.

### 1.6.2 Métricas de Calidad

Para evaluar este proyecto se aplicará las métricas de calidad según el estándar ISO/IEC 25010 El modelo de calidad representa la piedra angular en torno a la cual se establece el sistema para la evaluación de la calidad del producto. En este modelo se determinan las características de calidad que se van a tener en cuenta a la hora de evaluar las propiedades de un producto software determinado.

El modelo de calidad del producto definido por la ISO/IEC 25010 se encuentra compuesto por las nueve características de calidad que se muestran en la siguiente figura: (Hidalgo, 2020)

Adecuación funcional

Eficiencia de desempeño

Compatibilidad

Capacidad de interacción

Fiabilidad

Seguridad

Mantenibilidad

Flexibilidad

Protección

### **1.6.3 Costos**

Para estimar el costo se utilizará la herramienta COCOMO II el cual permitirá realizar estimaciones en función del tamaño del software, y en un conjunto de factores de costos y de escala. COCOMO II posee tres modelos que 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: (Sommerville, 2011, p. 637).

- Modelo de composición de aplicación
- Modelo de fase de diseño previo
- Modelo de fase posterior a la arquitectura

### **1.6.4 Seguridad ISO/IEC 25001**

Para el presente trabajo de grado se utilizará la norma 25001, conocida como SQuaRE (System and Software Quality Requirements and Evaluation), es una familia de normas que tiene por objetivo la creación de un marco de trabajo común para evaluar la calidad del software.

La información es un activo muy valioso que puede impulsar a destruir una empresa. Si se gestiona de forma adecuada, le permite trabajar con confianza capacidad de protección de la información y los datos de manera que personas o sistemas no autorizados no puedan leerlos o modificarlos. (Miss & Baena, 2019, p. 37)

Por lo tanto, el sistema para la seguridad de la información se basará en dicha norma.

### **1.6.5 Pruebas de Software**

Existe varios métodos modernos para realizar pruebas a distintos softwares que se realizan, la prueba de caja blanca, pruebas de caja negra, pruebas de carga, entre otros. Se hará uso de los siguientes:

#### **1.6.5.1 Pruebas de Caja Blanca**

La prueba de caja blanca es una forma de prueba de aplicaciones que proporciona al evaluador un conocimiento completo de la aplicación que se está probando, incluido el acceso al código fuente y a los documentos de diseño. Esta visibilidad en profundidad hace posible que las pruebas de caja blanca identifiquen problemas que son invisibles para las pruebas de caja gris y negra. (Pressman, 2023, p.120)

Al usar los métodos de prueba de caja blanca, puede derivar casos de prueba que:

Garanticen que todas las rutas independientes dentro de un módulo se revisaron al menos una vez,

- ✓ Revisen todas las decisiones lógicas en sus lados verdadero y falso,
- ✓ Ejecuten todos los bucles en sus fronteras y dentro de sus fronteras operativas.
- ✓ Revisen estructuras de datos internas para garantizar su validez.

#### **1.6.5.2 Pruebas de Caja Negra**

La prueba de caja negra, test funcional o prueba comportamental, es un tipo de prueba de software directa, cuya finalidad es analizar la compatibilidad entre las interfaces de cada uno de los componentes del software. No tiene en consideración la lógica interna del sistema. (Pressman, 2002, p. 423).

Las pruebas de caja negra intentan encontrar errores en las categorías siguientes:

- Funciones incorrectas o faltantes.

- Errores de interfaz.
- Errores en las estructuras de datos o en el acceso a bases de datos externas.
- Errores de comportamiento o rendimiento.
- Errores de inicialización y terminación.

### **1.6.5.3 Pruebas de Estrés**

Las pruebas de estrés se diseñan para simular situaciones de carga del mundo real. Conforme aumenta el número de usuarios simultáneos de la aplicación o el número de transacciones en línea o la cantidad de datos descargados o subidos estas pruebas de rendimiento ayudarán a responder preguntas sobre el rendimiento de la aplicación. (Pressman, 2023, p. 210)

### **1.6.5.4 Pruebas de Unitarias**

Las pruebas unitarias son el proceso en el que se prueba la unidad funcional de código más pequeña. Las pruebas de software ayudan a garantizar la calidad del código y son una parte integral del desarrollo de software. Una práctica recomendada en el desarrollo de software es escribir el software como unidades pequeñas y funcionales, y luego escribir una prueba unitaria para cada unidad de código. Puede escribir primero pruebas unitarias como código. Luego, ejecute ese código de prueba de forma automática cada vez que realice cambios en el código del software. De esta forma, si una prueba falla, puede aislar con rapidez el área del código que tiene el error. Las pruebas unitarias imponen paradigmas de pensamiento modular y mejoran la cobertura y calidad de las pruebas. Las pruebas unitarias automatizadas permiten que usted o sus desarrolladores dispongan de más tiempo para centrarse en la programación.

## 1.7 HERRAMIENTAS

En lo que respecta a las herramientas para el desarrollo del software se usará el servidor apache, como editor de interfaces web HTML, el FRAMEWORK de CODEIGNITER L, editor de código VISUAL STUDIO CODE además del lenguaje de alto nivel PHP para operar las consultas del sistema, para la construcción de la base de datos y poder centralizar la información se usará el gestor de base de datos MySQL que nos permitirá:

- Crear y administrar la base de datos, los usuarios y los permisos.
- Diseñar y ejecutar las instrucciones SQL.
- Realizar las consultas requeridas por el usuario.

### 1.7.1 Apache (servidor Web HTTP)

El servidor web Apache es el principal producto de software mantenido por la fundación Apache. Es un software de servidor web de código abierto, multiplataforma y gratuito. Como todo servidor web es responsable de atender las solicitudes de los clientes cuando quieren consultar una URL. Apache está atento a las solicitudes, que llegan mediante el protocolo HTTP y se encarga de enviar las respuestas a los clientes.(Zúñiga, 2023, p. 54)

### 1.7.2 CodeIgniter (Framework)

**CodeIgniter** es un potente framework PHP de tamaño muy reducido, creado para desarrolladores que necesitan un conjunto de herramientas simple y elegante para crear aplicaciones web con todas las funciones.(Jacinto, 2023, p. 10).

### 1.7.3 Visual Studio Code (VS Code)

Visual Studio Code, al que conocemos también como VSCode, es un editor de código para programadores gratuito, de código abierto y multiplataforma. Está desarrollado por Microsoft, una compañía con una dilatada experiencia en la creación de IDEs (entornos de

desarrollo integrados), que ha conseguido plasmar su larga tradición en el sector para ofrecer una herramienta ligera y práctica que la comunidad ha adoptado en masa (Zúñiga, 2024, p. 2).

#### **1.7.4 Lenguaje de Programación**

PHP es un lenguaje de programación de uso general que se utiliza, sobre todo, en el entorno del desarrollo web. Este lenguaje se utiliza generalmente para desarrollar el backend de una web, el lado del servidor. Aun así, tiene numerosas utilidades en frontend. Es por esto que es uno de los principales lenguajes de programación en el mundo de la programación web.

Cualquiera puede beneficiarse del aprendizaje de PHP, pero es aún más esencial para aquellos interesados en programación web. PHP está disponible en los principales sistemas operativos como Linux, Microsoft Windows y macOS. La mayoría de los servidores web incluyendo a Apache e IIS, también soportan PHP. (Institute, 2022, p. 2).

#### **1.7.5 Gestor de Base de Datos**

Es un sistema de gestión de base de datos relacionales (RDBMS) gratuito y de código abierto. Fue creado por los desarrolladores originales de MySQL por la preocupación de que MySQL pasara a ser comercializado después de que Oracle lo adquiriera en 2009.

MaríaDB está escrito en C y C++ y es compatible con varios lenguajes de programación, incluidos C, C#, Java, Python, PHP y Perl. MariaDB también es compatible con todos los principales sistemas operativos, incluidos Windows, Linux y macOS.

### **1.8 ALCANCES Y LIMITES**

#### **1.8.1 Alcances**

Para cumplir las metas y requerimientos de la institución se realiza la implementación y el desarrollo del sistema de información web contable de costos, el mismo que contendrá los siguientes módulos:

- Módulo de Administración de sistemas
  - ✓ Gestión de usuarios
  - ✓ Cargos de áreas
- Módulo de Reportes.
  - ✓ Búsqueda mediante numero
  - ✓ Reporte de flujo de entrada
  - ✓ Reporte de libros diarios
- Módulo de Gestión de Archivos
  - ✓ Adición
  - ✓ Consultas
  - ✓ Tipos de documentos
- Módulo de Costos y determinación de utilidades
  - ✓ Costos directos
  - ✓ Costos indirectos
  - ✓ Mano de obra
- ✓ Costos directos + Costos indirectos + Mano de obra

### **1.8.2 Limites**

El Sistema De Información Web de costos de producción, no realizara las siguientes tareas debido a que se enfocara solo y necesariamente en el área de producción.

- Los límites del sistema web de información de costos de producción son las siguientes:
- El sistema web de información de costos está orientado al control exhaustivo de los reportes diarios para que el administrador tenga un mejor control.
- La contabilidad de costos tendrá solo tendrá las funciones de calcular los costos de mano de obra, costos de producción, costos indirectos de producción.
- El sistema de desarrollo tendrá medidas de seguridad y confidencialidad que será implementado para el uso del mismo.

## **1.9 APORTES**

El siguiente aporte del presente proyecto, será la automatización de un sistema de información web contable de costos, que desde antes hasta ahora se procedían de forma manual. Se dejará una estructura de procesos administrativos para el manejo de los programas de evaluación.

Como aporte para la empresa “Kubic Textiles” es poderles brindar el sistema de información web para que se pueda desarrollar mejor sus actividades debido a que con la sistematización de costos se tiene un mejor seguimiento facilitando a que los informes y reportes sean eficientes a simple descarga para la toma de decisiones.

# **CAPÍTULO II**

## **MARCO TEÓRICO**

**INGENIERÍA  
DE SISTEMAS**  
UNIVERSIDAD PÚBLICA DE EL ALTO



## 2 CAPÍTULO II MARCO TEÓRICO

### 2.1 INTRODUCCIÓN

En este capítulo, se abordará la teoría que respalda el proyecto de grado en cuestión. Se presentarán conceptos clave y antecedentes relevantes que sirven de base para el desarrollo del proyecto de grado.

El objetivo de este capítulo es proporcionar un marco teórico sólido que permita comprender la relevancia y originalidad del proyecto, así como su pertinencia y contribución al campo de estudio.

### 2.2 SISTEMA

Un sistema se puede definir como una aplicación en la cual un usuario por medio de un navegador realiza peticiones a una aplicación remota accesible a través de Internet (o a través de una intranet) y que recibe una respuesta que se muestra en el propio navegador. Los sistemas web se pueden definir como aquellas aplicaciones o herramientas informáticas accesibles desde un servidor web a través de Internet o de una intranet mediante un navegador. Las aplicaciones web son muy utilizadas debido a la facilidad del navegador web y a la independencia del sistema operativo. (Angulo, 2021, p. 36)

### 2.3 SISTEMA DE INFORMACIÓN

Para (Angulo, 2021, p. 37), un sistema de información es un conjunto de componentes interrelacionados que recolectan, procesan, almacenan y distribuyen información para apoyar los procesos de toma de decisiones y de control en una organización.

- Un Sistema de Información realiza cuatro actividades básicas:
- Entrada de información: el sistema toma los datos que requiere.
- Almacenamiento de información: se realiza por computadora o archivos físicos para guardar la información.
- Procesamiento de información: transforma la información de entrada en

información útil para la toma de decisiones.

- Salida de información: propiedad de sacar la información procesada al exterior

## 2.4 WEB

Web es un documento que incluye un archivo **HTML** con texto, imágenes, videos, animaciones Flash, etc. Al conjunto de páginas web que suelen formar parte del mismo dominio o subdominio de Internet se lo conoce como sitio web. Dentro del sitio web, todas las páginas guardan alguna relación entre sí y están vinculadas mediante **vínculos** (también conocidos como enlaces, hipervínculos, hiperenlaces o links).

Muchas son las recomendaciones existentes acerca de las características imprescindibles que debe tener una página web para ser considerada buena, útil y óptima. No obstante, entre todas ellas destacaríamos las siguientes:

- Debe poseer una jerarquía perfectamente clara. Es decir, todos sus elementos deben estar claros y organizados en una estructura comprensible.
- Es imprescindible que respete y cumpla con los estándares web establecidos.
- Debe contar con el formato más útil para abordar y tratar los asuntos a los que se refiere.
- No menos importante es que ofrezca al usuario algo más. Es decir, se trata de que este encuentre en ella información, curiosidades y soluciones.
- Debe tener claro el público al que se dirige para así abordar las cuestiones que más le interesen y preocupen a aquel.
- Es fundamental que permita una navegación sencilla y rápida, sin grandes tiempos de espera para que se carguen sus elementos.
- Debe incluir material audiovisual que la haga más atractiva y dinámica.
- Tiene que permitir la interacción de todos sus internautas.

### 2.4.1 Web 2.0

**Web 2.0**, por otra parte, es un concepto que permite nombrar a las aplicaciones de Internet que facilitan la interacción de los usuarios. Los sitios que forman parte de la Web 2.0 posibilitan que los internautas compartan información y desarrollen tareas de forma mancomunada. Entre los ejemplos de Web 2.0 pueden nombrarse a las redes sociales. (Véliz Marisol, 2022, p. 36)

### 2.4.2 Diferencias.

La diferencia entre web 1.0 y 2.0 se puede definir al contemplar los cambios en la forma de consumir contenidos y de relacionarnos, la proliferación de actividad en las redes sociales, y principalmente el paso a una nueva forma de acercarnos a la generación de contenidos.

**Web 1.0** Este modelo no permitía la interacción entre usuarios, comentarios, citas, ni aportar ningún tipo de contenido; esta función era destinada por completo al administrador del sitio.

**Web 2.0** Con el paso del tiempo los avances en tecnología permitieron que aumentara el tamaño del ancho de banda, la consolidación de los dispositivos y funcionalidades inalámbricos, y la capacidad de conexión a la red. Así, se ha podido llegar a un nuevo punto en el modelo de creación y distribución del contenido, borrando las brechas técnicas del pasado y ofreciendo herramientas al alcance de un público más amplio. En síntesis, todos estamos en posibilidad de consumir, pero también de generar y compartir contenidos. Esta es una transformación de la dinámica unidireccional de la web 1.0 a una versión con rasgos muy participativos para el usuario.

## 2.5 COSTOS DE PRODUCCIÓN

El término Costos se viene usando en toda actividad de la vida cotidiana, ya sea con fines o sin fines de lucro; en sentido monetario el desembolso para la obtención de un bien o

servicio, puede significar la suma de esfuerzos y recursos que se han invertido para producir algo útil. Son los que se generan en el proceso de transformar la materia prima en productos terminados. (Véliz Marisol, 2022, p. 36)

Se subdividen en los siguientes elementos:

- Costos de materia prima. - El costo de materiales integrados al producto.
- Costos de mano de obra. - Es el costo que interviene directamente en la transformación del producto.
- Costos indirectos de producción. - Son los costos que intervienen en la transformación de los productos, con excepción de la materia prima y la mano de obra directa.

### 2.5.1 Costo

El costo, también llamado coste, es el desembolso económico que se realiza para la producción de algún bien o la oferta de algún servicio. El costo incluye la compra de insumos, el pago de la mano de obra, los gastos en la producción y los gastos administrativos, entre otras actividades.

### 2.5.2 Tipos de Costos

Los costos pueden clasificarse según diversos criterios:

Según su comportamiento:

**Costos fijos.** Son aquellos costos que posee una empresa más allá de la producción obtenida, es decir, que su valor no varía según lo producido. Los costos fijos solo pueden ser estipulados a corto plazo, ya que con el correr del tiempo eventualmente varían. Algunos ejemplos son: el pago de alquileres y los impuestos.

**Costos variables.** Son aquellos costos que varían en relación a lo producido, es decir, que si se aumenta la producción estos costos serán mayores y viceversa. Algunos ejemplos son: la cantidad de materia prima utilizada, el embalaje y empaquetado de los productos.

**Costo semi – variable.** Son aquellos costos que pueden variar según lo producido, pero estos cambios son más bien progresivos.

Según la relación entre los factores de producción y los productos:

**Costos indirectos.** Son aquellos costos que tienen consecuencias sobre la producción en forma total, es decir, que no pueden ser asignados a un determinado producto. Los costos deben ser repartidos equitativamente entre todos los bienes producidos. Por ejemplo: el aumento en insumos de limpieza de la fábrica o el aumento de los salarios de los trabajadores.

**Costos directos.** Son aquellos costos que pueden ser asignados a cada bien o servicio en particular. Por ejemplo: el aumento de harina (que tiene una repercusión directa sobre la producción de pan) o el aumento de tinta (que impacta en la producción de libros).

Según su naturaleza:

**Costos de mano de obra.** Son aquellos costos que derivan del pago de salarios de todos los miembros que participan en el proceso productivo y de todos los que forman parte de la estructura empresarial.

**Costos de materia prima.** Son aquellos costos en los que se incurre al adquirir la materia prima para realizar un determinado producto.

**Costos financieros.** Son aquellos costos que se necesitan para el financiamiento del negocio.

**Costos de distribución.** Son aquellos costos que se producen en el sistema de distribución del producto.

**Costos tributarios.** Son aquellos costos relacionados al pago de impuestos.

### 2.5.3 Costo y Gasto

Costo y gasto son dos conceptos que se suele tomar como sinónimos, pero que en contabilidad difieren sustancialmente.

El costo se define como todas aquellas inversiones necesarias para la producción del bien o servicio, como son: la mano de obra, las materias primas. Los costos son considerados como inversiones, ya que se espera que vuelvan como forma de ganancias para la empresa.

Los gastos son todos aquellos pagos que la empresa debe realizar por aspectos que no tienen que ver directamente con la producción, como el pago de sueldos del personal administrativo. Estos gastos son los que permiten el correcto funcionamiento de la empresa, pero que no se encuentran directamente relacionados con las ganancias.

### 2.5.4 Utilidades

El margen ganado depende del ingreso total ganado menos todos los costos en los que se tuvo que incurrir para llegar a dichos ingresos. (Véliz, 2022, p. 62). Dicho de esta manera, si se requiere conocer el margen ganado en términos monetarios solo es cuestión de tomar el valor total de los ingresos o unitarios y restarlos con el valor total de los costos o unitarios, es decir que si se quiere expresar el margen en términos monetarios sería ingresos menos costos (que corresponde a la parte de arriba de la fórmula). En caso de requerir el margen en términos porcentuales, se debe dividir este resultado entre los ingresos:

$$\text{Margen ganado en \%} = \frac{\text{Ingresos} - \text{Costo total}}{\text{Ingresos}} \times 100$$

## 2.6 METODOLOGÍA WEB BASADA EN UML(UWE)

La metodología que se emplea para este proyecto es UWE (Ingeniería Web Basada en UML) por las características que presenta. UWE es una metodología basada en el proceso

unificado y UML (Lenguaje Unificado de Modelado) para el desarrollo de aplicaciones web, cubre todo el ciclo de vida de las aplicaciones web (Zaens, 2015, p. 2).

### **2.6.1 Metodología UWE (UML-Based Web Engineering)**

Es una metodología detallada para el proceso de autoría de aplicaciones con una definición exhaustiva del proceso de diseño que debe ser utilizado. Este proceso, iterativo e incremental, incluye flujos de trabajo y puntos de control, y sus fases coinciden con las propuestas en el Proceso Unificado de Modelado.

UWE está especializada en la especificación de aplicaciones adaptativas, y por tanto hace especial hincapié en características de personalización, como es la definición de un modelo de usuario o una etapa de definición de características adaptativas de la navegación en función de las preferencias, conocimiento o tareas de usuario (Sanz, 2022, p. 9).

### **2.6.2 Características de UWE**

La metodología UWE define vistas especiales representadas gráficamente por diagramas en UML, tales como el modelo de navegación y el modelo de presentación. Los diagramas se pueden adaptar como mecanismos de extensión basados en estereotipos que proporciona UML.

Los diagramas se pueden adaptar como mecanismos de extensión basados en estereotipos que proporciona UML. Estos mecanismos de extensión son los que UWE utiliza para definir estereotipos que son los que finalmente se utilizan en las vistas especiales para el modelado de aplicaciones web. De esta manera se obtiene una notación UML adecuada para un dominio específico a la que se conoce como “perfil UML”

Un perfil UML consiste en una jerarquía de estereotipos y un conjunto de restricciones. Los estereotipos son utilizados para preparar instancias de las clases. Las ventajas de utilizar los perfiles de UML es que casi todas las herramientas CASE de UML los reconocen. Los

modelos deben ser fácilmente adaptables al cambio en cualquier etapa del desarrollo. (Pachas, 2019, p. 15)

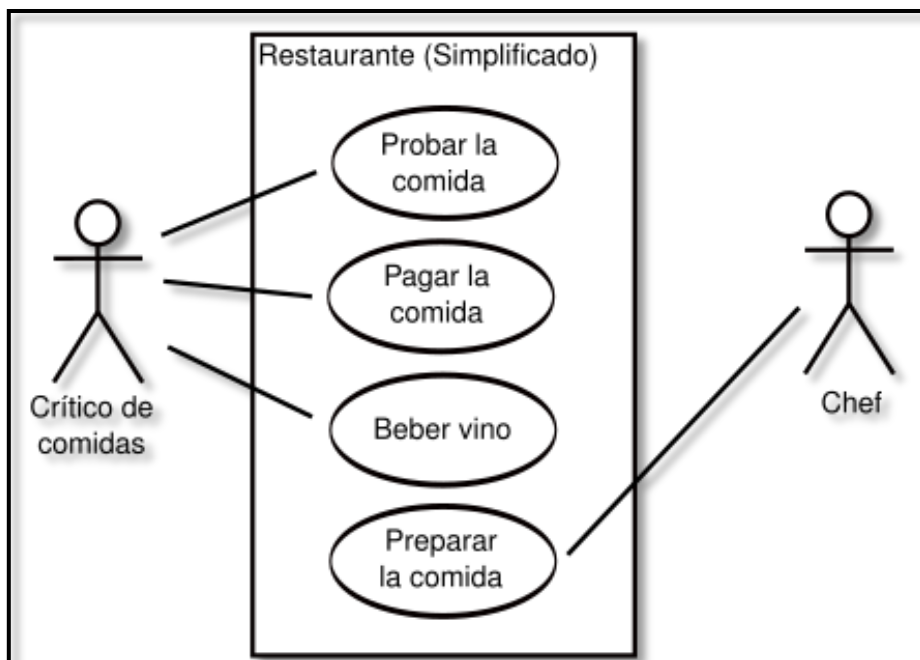
### **2.6.3 Modelos de la Metodología UWE**

El modelo que propone UWE está compuesto por sub modelos:

- Modelo de Requisitos “caso de uso”
- Modelo de contenido “conceptual”
- Modelo de navegación
- Modelo de presentación
- Modelo de proceso

#### **2.6.3.1 Modelo de Requisitos “Casos de Uso”**

El modelo de caso de uso identifica los requisitos del sistema en términos de la funcionalidad que debe existir para conseguir los objetivos establecidos por el usuario o para resolver un problema identificado por el usuario. Los casos de uso describen los comportamientos principales identificados en los requisitos y describen el valor que los resultados dan a los usuarios; no describen cómo funciona internamente el sistema. Los actores son los usuarios del sistema y representan los diferentes roles que la gente y otros sistemas representan cuando interactúan con el sistema (Model, 2021, p. 22).

**Figura 2***Modelo de Casos de Uso*

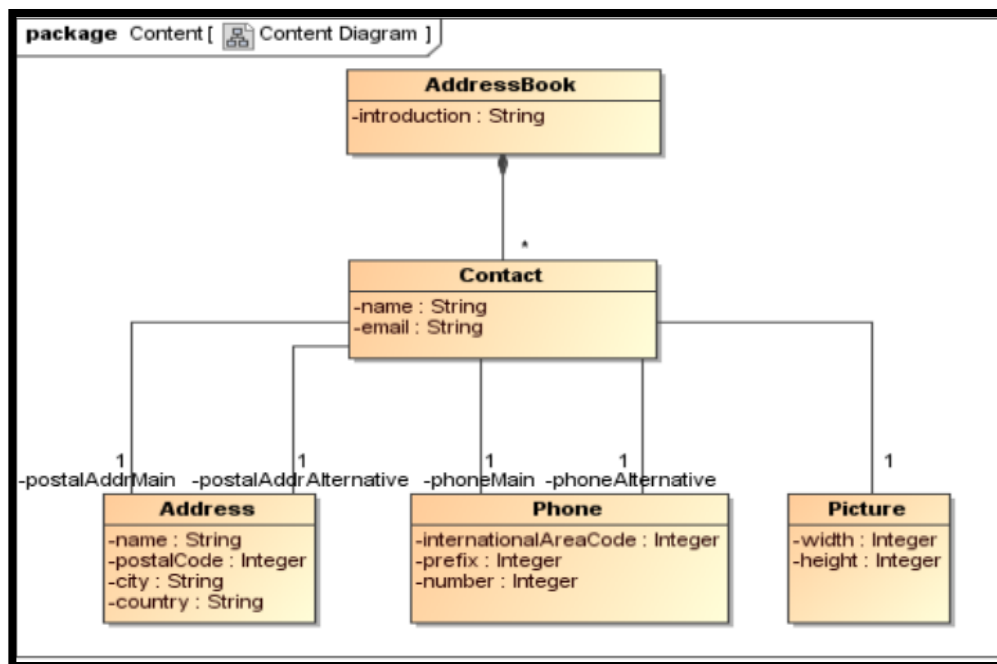
*Nota: Caso de uso de manera general, Pressma, 2019, Diagrama de casos de uso.*

### **2.6.3.2 Modelo de Contenido "Conceptual"**

UWE apunta a construir un modelo conceptual de una aplicación Web, procura no hacer caso en la medida de lo posible de cuestiones relacionadas con la navegación, y de los aspectos de interacción de la aplicación Web. La construcción de este modelo lógico-conceptual se debe llevar a cabo de acuerdo con los casos de uso que se definen en la especificación de requerimientos. El modelo conceptual incluye los objetos implicados en las actividades típicas que los usuarios realizarán en la aplicación Web.

Figura 3

Modelo de Contenido



*Nota. Modelo de contenido en Ingeniería de la Web, Guerrero, 2014, según UWE-UML.*

### 2.6.3.3 Modelo de Navegación

Consta de la construcción de dos modelos de navegación, el modelo del espacio de navegación y el modelo de la estructura de navegación. El primero especifica que objetos serán visitados por el navegador a través de la aplicación. El segundo define como se relacionarán. (Pachas, 2019, p. 15)

### 2.6.3.4 Modelo de Presentación

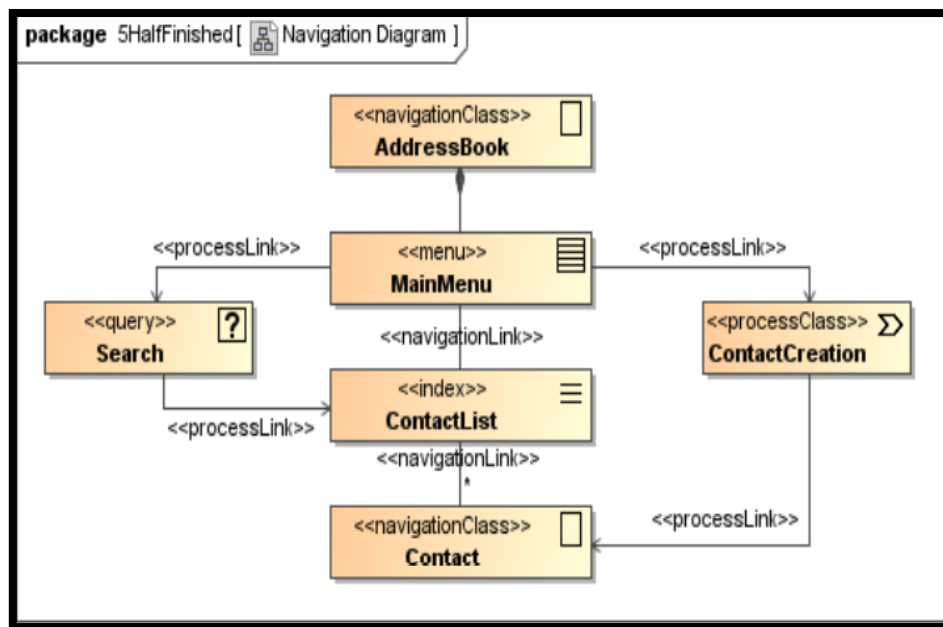
Describe dónde y cómo los objetos de navegación y accesos primitivos serán presentados al usuario, es decir, una representación esquemática de los objetos visibles al usuario.

Los elementos de modelado son:

- **Clases de Navegación**, que se denotan con (0), representan los nodos navegables de la estructura de hipertexto.
- **Links de Navegación**, que muestran el vínculo directo entre las clases de navegación.
- **Caminos de Navegación Alternativos**, los cuales son visualizados con el estereotipo <<menu>> ().
- **Primitivas de Acceso**, las cuales se utilizan ya sea para llegar a múltiples instancias de una clase de navegación (<<index>> o <<guided tour>>) o para seleccionar ítems (<<query>>).

**Figura 4**

Modelo de Navegación



*Nota. Modelo de navegación en Ingeniería de la Web, Guerrero, 2014, según UWE-UML.*

### 2.6.3.5 Modelo de Proceso

Representa los aspectos dinámicos de la aplicación Web y especifica funcionalidad cómo transacciones y workflows de actividades. Se modela mediante un diagrama de actividades de UML, y es resultado de refinar el diagrama de actividades modelado durante la especificación de requerimientos. Muestra el flujo de la ejecución representado por nodos de actividad conectados, los nodos de control que proveen constructores de flujo de control como decisiones y sincronización y nodos de objetos que representan el flujo de datos. (Rotta, 2016, p. 10)

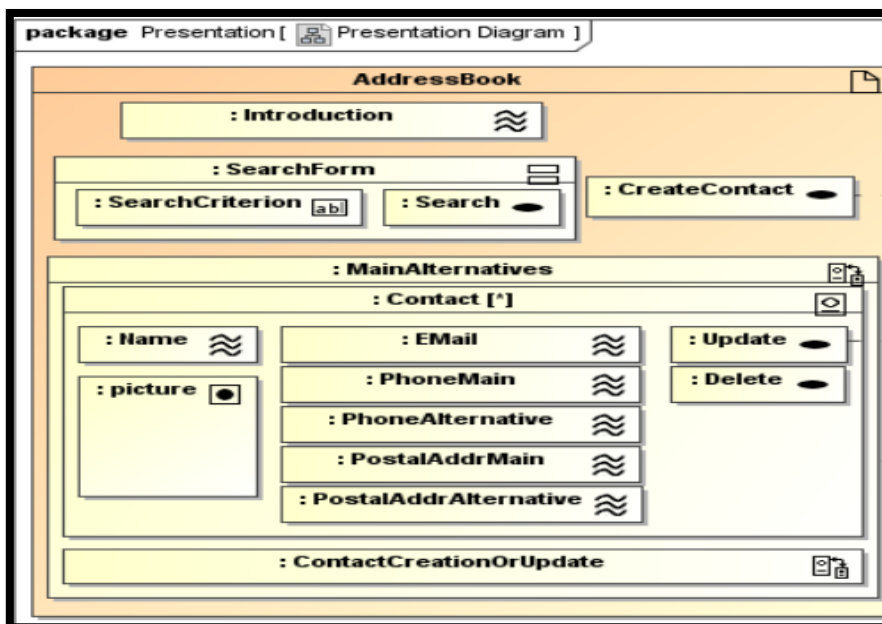
**Clases de Presentación**, las cuales se basan directamente en los nodos del modelo de navegación, cada clase está compuesta por elementos de UI tales como, texto (<<text>>), vinculo (<<anchor>>), botón (<<button>>), imagen (<<image>>), formulario (<<form>>), y colección de vínculos (<<anchored collection>>)

**Páginas Web (<<page>>)**, que se utilizan para modelar la información proveniente de varios nodos de navegación y que se presentan en una misma página web.

**Grupo de Presentación (<<presentation group>>)**, el cual es un contenedor de clases de presentación, y a su vez de otros grupos de presentación.

Figura 5

Modelo de Presentación



*Nota. Modelo de presentación en Ingeniería de la Web, Guerrero, 2014, según UWE-UML.*

#### 2.6.4 Fases de la Metodología UWE

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

##### 2.6.4.1 Captura, Análisis y Especificación de Requisitos

En simple palabras y básicamente, durante esta fase, se adquieren, reúnen y especifican las características funcionales y no funcionales que deberá cumplir la aplicación web.

Trata de diferente forma las necesidades de 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.

#### **2.6.4.2 *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.

#### **2.6.4.3 *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.

#### **2.6.4.4 *Pruebas***

Las pruebas se utilizan para asegurar el correcto funcionamiento de secciones de código.

#### **2.6.4.5 *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.

#### **2.6.4.6 *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.

## 2.7 ARQUITECTURA DE SOFTWARE (MVC)

MVC es una propuesta de arquitectura del software utilizada para separar el código por sus distintas responsabilidades, manteniendo distintas capas que se encargan de hacer una tarea muy concreta, lo que ofrece beneficios diversos.

MVC se usa inicialmente en sistemas donde se requiere el uso de interfaces de usuario, aunque en la práctica el mismo patrón de arquitectura se puede utilizar para distintos tipos de aplicaciones. Surge de la necesidad de crear software más robusto con un ciclo de vida más adecuado, donde se potencie la facilidad de mantenimiento, reutilización del código y la separación de conceptos.

MVC es un "invento" que ya tiene varias décadas y fue presentado incluso antes de la aparición de la Web. No obstante, en los últimos años ha ganado mucha fuerza y seguidores gracias a la aparición de numerosos framework de desarrollo web que utilizan el patrón MVC como modelo para la arquitectura de las aplicaciones web (Gonzales, 2021, p. 10).

### 2.7.1 Modelo

Es la capa donde se trabaja con los datos, por tanto, contendrá mecanismos para acceder a la información y también para actualizar su estado. Los datos los tendremos habitualmente en una base de datos, por lo que en los modelos tendremos todas las funciones que accederán a las tablas y harán los correspondientes selects, updates, inserts, etc.

El modelo es el responsable de:

Acceder a la capa de almacenamiento de datos. Lo ideal es que el modelo sea independiente del sistema de almacenamiento.

Define las reglas de negocio (la funcionalidad del sistema). Un ejemplo de regla puede ser: "Si la mercancía pedida no está en el almacén, consultar el tiempo de entrega estándar del proveedor".

Lleva un registro de las vistas y controladores del sistema.

Si estamos ante un modelo activo, notificará a las vistas los cambios que en los datos pueda producir un agente externo (por ejemplo, un fichero por lotes que actualiza los datos, un temporizador que desencadena una inserción, etc.).

### **2.7.2 Vista**

Las vistas, como su nombre nos hacen entender, contienen el código de nuestra aplicación que va a producir la visualización de las interfaces de usuario, o sea, el código que nos permitirá renderizar los estados de nuestra aplicación en HTML. En las vistas nada más tenemos los códigos HTML y PHP que nos permite mostrar la salida.

En la vista generalmente trabajamos con los datos, sin embargo, no se realiza un acceso directo a éstos. Las vistas requerirán los datos a los modelos y ellas se generará la salida, tal como nuestra aplicación requiera. Este presenta el modelo en un formato adecuado para interactuar, usualmente la interfaz de usuario.

Las vistas son responsables de:

Recibir datos del modelo y los muestra al usuario.

Tienen un registro de su controlador asociado (normalmente porque además lo instancia).

Pueden dar el servicio de "Actualización ()", para que sea invocado por el controlador o por el modelo (cuando es un modelo activo que informa de los cambios en los datos producidos por otros agentes).

### 2.7.3 Controlador

Contiene el código necesario para responder a las acciones que se solicitan en la aplicación, como visualizar un elemento, realizar una compra, una búsqueda de información, etc.

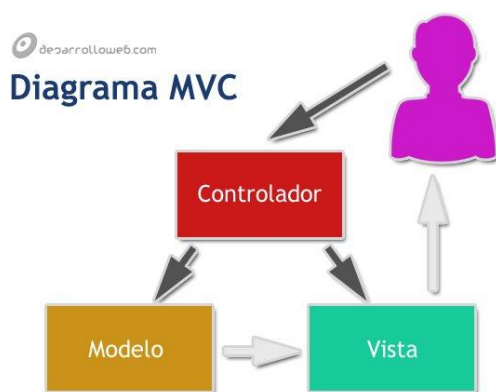
En realidad, es una capa que sirve de enlace entre las vistas y los modelos, respondiendo a los mecanismos que puedan requerirse para implementar las necesidades de nuestra aplicación. Sin embargo, su responsabilidad no es manipular directamente datos, ni mostrar ningún tipo de salida, sino servir de enlace entre los modelos y las vistas para implementar las diversas necesidades del desarrollo.

### Arquitectura de aplicaciones MVC

A continuación, encontrarás un diagrama que te servirá para entender un poco mejor cómo colaboran las distintas capas que componen la arquitectura de desarrollo de software en el patrón MVC.

**Figura 6**

*Modelo Vista Controlador*



**Nota. Característica en un Web, Guerrero, 2014, según MVC.**

Veamos paso a paso cómo sería el flujo de trabajo característico en un esquema MVC.

1. El usuario **realiza una solicitud** a nuestro sitio web. Generalmente estará desencadenada por acceder a una página de nuestro sitio. Esa solicitud le llega al controlador.
2. El **controlador comunica tanto con modelos como con vistas**. A los modelos les solicita datos o les manda realizar actualizaciones de los datos. A las vistas les solicita la salida correspondiente, una vez se hayan realizado las operaciones pertinentes según la lógica del negocio.
3. Para producir la salida, en ocasiones las **vistas pueden solicitar más información a los modelos**. En ocasiones, el controlador será el responsable de solicitar todos los datos a los modelos y de enviarlos a las vistas, haciendo de puente entre unos y otros. Sería corriente tanto una cosa como la otra, todo depende de nuestra implementación; por eso esa flecha la hemos coloreado de otro color.
4. **Las vistas envían al usuario la salida**. Aunque en ocasiones esa salida puede ir de vuelta al controlador y sería éste el que hace el envío al cliente, por eso he puesto la flecha en otro color.

## 2.8 HERRAMIENTAS

### 2.8.1 PHP (Hypertext Pre-processor)

PHP, acrónimo de Hypertext Preprocessor, es un lenguaje de programación orientado a objetos, de código abierto muy popular para el desarrollo web que puede ajustarse en HTML. Es considerado como uno de los lenguajes más potentes y populares para el desarrollo web y de aplicaciones. (González, 2020, p.80).

Las páginas de PHP contienen HTML con código incrustado que cumple una función y está encerrado entre las etiquetas especiales de comienzo y final.

Lo que distingue a PHP es su extrema simplicidad para programadores, desarrolladores o ingenieros en un nivel principiante, pero al mismo tiempo ofrece una gran variedad de características avanzadas para los profesionales en niveles más desarrollados.

### **Funciones del lenguaje PHP**

- Programación del lado del servidor: ideal para páginas webs de prueba antes de la publicación en la web.
- Programación a través de la línea de comandos: los scripts PHP pueden crearse sin necesidad de un navegador o servidor, sólo se requiere de un analizador PHP. Los scripts son ideales para las tareas regulares de una web.
- Aplicaciones de escritorio: a pesar de ser una función que se utiliza con menor frecuencia, el lenguaje PHP también es adecuado para el desarrollo de aplicaciones de escritorio gracias a su nivel de complejidad.

### **2.8.2 Apache**

Apache es un servidor web que se encarga de almacenar, procesar y servir las páginas web a los usuarios de las mismas. Se distribuye bajo una licencia de código abierto, lo que quiere decir que es gratuito y fácilmente adaptable, personalizable y reutilizable.

Creado por el desarrollador de software estadounidense Robert McCool en 1995, es considerado una de las tecnologías que impulsó el crecimiento inicial de Internet.

Desde entonces, es ofrecido por la Apache Software Foundation, una organización sin fines de lucro que se encarga de supervisar la comunidad de usuarios que lo desarrollan y mantienen.

### 2.8.3 Gestor de Base de Datos

Es un sistema de gestión de bases de datos relacionales (RDBMS) gratuito y de código abierto. Fue creado por los desarrolladores originales de MySQL por la preocupación de que MySQL pasara a ser comercializado después de que Oracle lo adquiriera en 2009.

MySQL está escrito en C y C++ y es compatible con varios lenguajes de programación, incluidos C, C#, Java, Python, PHP y Perl. MariaDB también es compatible con todos los principales sistemas operativos, incluidos Windows, Linux y macOS (Bartholomew, 2014, p. 4).

### 2.8.4 Bootstrap

Bootstrap es un *framework* que permite a los desarrolladores web construir páginas web *responsives* de una forma más rápida y sencilla. En este sentido, proporciona un conjunto de componentes y plantillas CSS, HTML y JavaScript que cualquiera puede utilizar o modificar de manera gratuita.

Su origen lo encontramos en las oficinas de Twitter, donde Mark Otto y Jacob Thornton lanzaron la primera versión en agosto de 2011 bajo el nombre Twitter Blueprint. Inicialmente, esta fue una herramienta interna que permitía que el diseño de esta popular red social fuera coherente y consistente. Sin embargo, meses más tarde, Twitter liberó el código y el *framework* en GitHub y lo renombró como Bootstrap.

Actualmente, Bootstrap se encuentra en su quinta versión y se ha convertido en uno de los *frameworks* de desarrollo más populares. De hecho, según estadísticas de [W3Techs](#), el 19% de las páginas web lo utilizan.

### 2.8.5 Visual Studio Code

Visual Studio Code es un editor de código fuente desarrollado por Microsoft para Windows, Linux y macOS. Es un editor de código fuente súper rápido y liviano que se puede

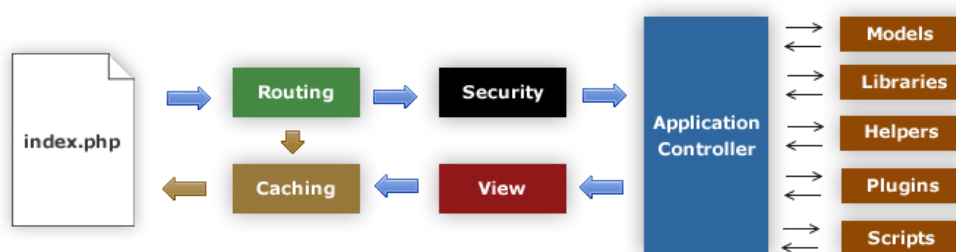
usar para ver, editar, ejecutar y depurar código fuente para aplicaciones. Es utilizado principalmente por desarrolladores front-end. Visual Studio Code es un producto completamente diferente en comparación con Visual Studio.

### 2.8.6 Framework CodeIgniter

CodeIgniter es un marco de desarrollo de aplicaciones (un conjunto de herramientas) para personas que crean sitios web con PHP. Su objetivo es permitirle desarrollar proyectos mucho más rápido de lo que podría hacerlo si escribiera el código desde cero, al proporcionar un amplio conjunto de bibliotecas para tareas que se necesitan con frecuencia, así como una interfaz simple y una estructura lógica para acceder a estas bibliotecas. CodeIgniter le permite centrarse creativamente en su proyecto al minimizar la cantidad de código necesario para una tarea determinada.

**Figura 7**

*Arquitectura de la Aplicación*



**Nota. Estructura lógica para acceder a estas bibliotecas, Guerrero, 2014, según CodeIgniter.**

Algunas ventajas de usar el framework CodeIgniter:

- Las páginas se procesan más rápido, el núcleo de CodeIgniter es bastante ligero.
- Es sencillo de instalar, basta con subir los archivos al ftp y tocar un archivo de configuración para definir el acceso a la base de datos.

- Existe abundante documentación en la red.
- Facilidad de edición del código ya creado.
- Facilidad para crear nuevos módulos, páginas o funcionalidades.
- Estandarización del código
- Separación de la lógica y arquitectura de la web, el MVC.
- Cualquier servidor que soporte PHP+MySQL sirve para CodeIgniter.
- CodeIgniter se encuentra bajo una licencia open source, es código libre.

## **2.9 ESTIMACIÓN DE COSTOS**

El objetivo de la estimación de costo de software consiste en estimar el tamaño, el esfuerzo, la complejidad y el costo del proyecto de software para poder encontrar la mejor decisión de desarrollo y asegurar que el gasto se encuentre de acuerdo con lo presupuestado. Por todo lo anterior, la estimación de costos de software es importante para la planificación, programación, presupuesto y establecimiento del precio indicado al desarrollo del software (Alarcón, 2023, p. 14).

### **2.9.1 Cocomo II**

Como se conoce, una de las tareas de mayor importancia en la planificación de proyectos de software es la estimación, la cual consiste en determinar, con cierto grado de certeza, los recursos de hardware y software, costo, tiempo y esfuerzo necesarios para el desarrollo de los mismos.

Este trabajo describe un modelo de estimación, propuesto por Barry Boehm, llamado COCOMO II. Este modelo permite realizar estimaciones en función del tamaño del software, y de un conjunto de factores de costo y de escala. Los factores de costo describen aspectos relacionados con la naturaleza del producto, hardware utilizado, personal involucrado, y

características propias del proyecto. El conjunto de factores de escala explica las economías y deseconomías de escala producidas a medida que un proyecto de software incrementa su tamaño.(Gómez, 2024, p. 1)

Cocomo II posee tres modelos y se adaptan a las necesidades de los diferentes sectores, como al tipo de 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.(COCOMO II, 2016b, p. 5)

### **2.9.1.1 Ecuaciones del Modelo COCOMO**

El esfuerzo necesario para concretar un proyecto de desarrollo de software, cualquiera sea el modelo empleado, se expresa en meses/persona (PM) y representa los meses de trabajo de una persona fulltime, requeridos para desarrollar el proyecto (Bustamante, 2021, p. 5).

Las ecuaciones que se utilizan en los tres modelos son:

#### **( 1 ) Ecuación de Esfuerzo**

$$E = a * KLDC^b * FAE \quad (1)$$

**Donde**

**E** = Es el esfuerzo requerido por el proyecto en persona-mes.

**KLDC** = Es la cantidad de líneas de código en miles

**FAE** = Factores de atributos de esfuerzo.

**a** = Coeficiente Cocomo II

**b:** = Coeficiente Cocomo II

**( 2 ) Ecuación Tiempo de Esfuerzo**

$$Tdev = c(E)^d, \text{ en meses} \quad (2)$$

$E$  = Es el esfuerzo requerido por el proyecto, en persona-mes.

$Tdev$  = Es el esfuerzo requerido por el proyecto, en meses.

$c$  y  $d$  = Constantes con valores definidos en una tabla

**( 3 ) Ecuación Número de Personas**

$$P = \frac{E}{Tdev} \quad (3)$$

**Dónde:**

$E$  = Es el esfuerzo requerido por el proyecto, en persona-mes.

$Tdev$  = Es el esfuerzo requerido por el proyecto, en meses.

$P$  = Número de personas

A la vez, cada Submodelo también se divide en modos que representan el tipo de proyecto, y puede ser:

**Modo Orgánico:** Un pequeño grupo de programadores experimentados desarrollan software en un entorno familiar. El tamaño del software varía desde unas pocas líneas.

**Modo Semilibre o Semiencajado:** Corresponde a un esquema intermedio entre el orgánico y el rígido; el grupo de desarrollo puede incluir una mezcla de personas experimentadas y no experimentadas.

**Modo Rígido o Empotrado:** El proyecto tiene fuertes restricciones, que pueden estar relacionadas con la funcionalidad y/o pueden ser técnicas. El problema a resolver es único y es difícil basarse en la experiencia.

### 2.9.1.2 Modelos de COCOMO II

- **Modelo Básico**

Se utiliza para obtener una primera aproximación rápida del esfuerzo, y hace uso de la siguiente tabla de constantes para calcular distintos aspectos de costes.

**Tabla 1**

*Tabla de Valores de Modelo Básico*

Modo	a	b	c	d
Orgánico	2.40	1.05	2.5	0.38
Semi acoplado	3.00	1.12	2.5	0.35
Empotrado	3.60	1.20	2.5	0.32

Estos valores son para las fórmulas:

**( 4 ) Ecuación Personas Por Mes Según Modelo**

$$E = a * KLDC^b * FAE \quad (4)$$

Donde:

$a$  = Coeficiente Cocomo II

$b$  = Coeficiente Cocomo II

$c$  = Coeficiente Cocomo II

$d$  = Coeficiente Cocomo II

$KLDC$  = líneas de código

**( 5 ) Ecuación Tiempo de Proyecto Según Modelo**

$$Tdev = c(E)^d \quad (5)$$

Donde:

$c$  = Coeficiente Cocomo II

$d$  = Coeficiente Cocomo II

$E$  = Esfuerzo

$Tdev$  = Tiempo del proyecto

**( 6 ) Ecuación Costo Personal**

$$P = \frac{E}{Tdev} \quad (6)$$

Donde:

$E$  = Esfuerzo

$Tdev$  = Tiempo requerido

$P$  = Costo personal

**( 7 ) Ecuación de Costo Total del Proyecto**

$$Csof = Smes * P * Tdev \quad (7)$$

Donde:

$Smes$  = Sueldo mes

$P$  = Número de personas

$Tdev$  = Tiempo requerido

$Csof$  = Costo total del proyecto

Se puede observar que a medida que aumenta la complejidad del proyecto (modo), las constantes aumentan de 2.4 a 3.6, que corresponde a un incremento del esfuerzo del personal. Hay que utilizar con mucho cuidado el modelo básico puesto que se obvian muchas características del entorno

### **2.9.1.3 Ventajas de COCOMO II**

Este modelo se usa en las etapas tempranas de un proyecto de software, cuando se conoce muy poco del tamaño del producto a ser desarrollado, de la naturaleza de la plataforma, del personal a ser incorporado al proyecto o detalles específicos del proceso a utilizar. Este modelo podría emplearse tanto en productos desarrollados en sectores de Generadores de Aplicación, Sistemas Integrados o Infraestructura. (Galarza, 2017, p. 10)

Las ventajas del modelo de estimación de costos COCOMO II son:

- Es fácil de realizar y de interpretar
- Tiene pocas variables
- Se acerca a la realidad en la mayoría de los casos

Desventajas del Modelo Cocomo II

- No saca resultados fiables en proyectos demasiado pequeños
- La elección de las variables es muy subjetiva depende de la persona que realiza el estudio.

## **2.10 MÉTRICAS DE CALIDAD**

Son todas las métricas de software que definen de una u otra forma la calidad del software. Tales como exactitud, estructuración o modularidad, pruebas, mantenimiento, reusabilidad, cohesión del módulo, acoplamiento del módulo, etc. Estas son los puntos críticos en el diseño, codificación, pruebas y mantenimiento. (García, 2010, p. 8)

Para el presente proyecto tomaremos en cuenta la métrica de calidad ISO/IEC 25010 hace parte de la familia ISO 25000. Es una norma que está centrada hacia la usabilidad, en el cual se determinan las características de calidad que se deben tener en cuenta en el momento de evaluar las propiedades de un producto.

### 2.10.1 ISO/IEC 25010

La calidad del producto software se puede interpretar como el grado en que dicho producto satisface los requisitos de sus usuarios aportando de esta manera un valor. Son precisamente estos requisitos (funcionalidad, rendimiento, seguridad, mantenibilidad, etc.) los que se encuentran representados en el modelo de calidad, el cual categoriza la calidad del producto en características y subcaracterísticas.

El modelo de calidad del producto definido por la ISO/IEC 25010 se encuentra compuesto por las nueve características de calidad que se muestran en la siguiente figura

**Figura 8**

*ISO/IEC 25010*

CALIDAD DEL PRODUCTO SOFTWARE								
ADECUACIÓN FUNCIONAL	EFICIENCIA DE DESEMPEÑO	COMPATIBILIDAD	CAPACIDAD DE INTERACCIÓN	FIABILIDAD	SEGURIDAD	MANTENIBILIDAD	FLEXIBILIDAD	PROTECCIÓN
COMPLETITUD FUNCIONAL	COMPORTAMIENTO TEMPORAL	COEXISTENCIA	RECONOCIBILIDAD DE ADECUACIÓN	AUSENCIA DE FALLOS	CONFIDENCIALIDAD	MODULARIDAD	ADAPTABILIDAD	RESTRICCIÓN OPERATIVA
CORRECCIÓN FUNCIONAL	UTILIZACIÓN DE RECURSOS	INTEROPERABILIDAD	APRENDIZABILIDAD	DISPONIBILIDAD	INTEGRIDAD	REUSABILIDAD	ESCALABILIDAD	IDENTIFICACIÓN DE RIESGOS
PERTINENCIA FUNCIONAL	CAPACIDAD		OPERABILIDAD	TOLERANCIA A FALLOS	NO-REPUDIO	ANALIZABILIDAD	INSTALABILIDAD	PROTECCIÓN ANTE FALLOS
			PROTECCIÓN FRENTE A ERRORES DE USUARIO	RECUPERABILIDAD	RESPONSABILIDAD	CAPACIDAD DE SER MODIFICADO	REEMPLAZABILIDAD	ADVERTENCIA DE PELIGRO
			INVOLUCRACIÓN DEL USUARIO		AUTENTICIDAD	CAPACIDAD DE SER PROBADO		INTEGRACIÓN SEGURA
			INCLUSIVIDAD		RESISTENCIA			
			ASISTENCIA AL USUARIO					
			AUTO-DESCRIPTIVIDAD					
<a href="http://iso25000.com">iso25000.com</a>								

**Nota. Calidad del producto software, 2011, según ISO/IEC 25010**

#### 2.10.1.1 Funcionalidad

Representa la capacidad del producto software para proporcionar funciones que satisfacen las necesidades declaradas e implícitas de los usuarios cuando el producto se usa en las condiciones especificadas. Esta característica se subdivide a su vez en las siguientes subcaracterísticas:

- Completitud funcional. Grado en el que el conjunto de funcionalidades del producto cubre todas las tareas y los objetivos de usuario especificados.

- Corrección funcional. Capacidad del producto o sistema para proveer resultados exactos cuando es usado por los usuarios especificados.
- Pertinencia funcional. Capacidad del producto software para proporcionar un conjunto de funciones que facilitan la consecución de tareas y objetivos de usuario especificados.

Se llevará a cabo una exhaustiva evaluación para determinar si el software cumple con los requisitos funcionales establecidos. Este proceso garantiza que el programa posea las características esenciales para satisfacer las demandas y expectativas del usuario, asegurando su plena funcionalidad.

A continuación, en la tabla se muestra la suma total de los puntos de función sin ajustar, para obtener los valores ajustados se aplican factores de complejidad específicos que se detallan a continuación en la siguiente tabla. Estos factores consideran diversas características del sistema.

**Tabla 2**

*Indices de Usabilidad*

IMPORTANCIA	0%	20%	40%	60%	80%	100%
ESCALA	NO INFLUENCIA	INCIDENTAL	MODERADO	MEDIO	SIGNIFICATIVO	ESENCIAL
FACTOR	0	1	2	3	4	5
¿Requiere el sistema copias de seguridad y de recuperación fiable?						X
¿Se requiere 0 comunicación de datos?					X	
¿Existe funciones de procesos distribuidos?					X	

¿Es crítico el rendimiento?	X	
¿El sistema web será ejecutado el SO Actual?		X
¿Se requiere una entrada interactiva para el sistema?		X
¿Se requiere que el sistema tenga entradas a datos con múltiples ventanas?	X	
¿Se actualiza los archivos de forma interactiva?		X
¿Son complejas las entradas, salidas, los archivos o las peticiones?		X
¿Es complejo el procesamiento interno del sistema?		X
¿Se ha diseñado el código para ser reutilizado?		X
¿Se ha diseñado el sistema para facilitar al usuario el trabajo y ayudarlos a encontrar la información?	X	

Fórmula para ajustar puntos de función, que considera factores como la complejidad técnica, el rendimiento y la usabilidad.

**( 8 ) Fórmula de Ajuste**

$$PF = CT \left[ 0.65 + 0.01 * \sum fi \right] \quad (8)$$

Donde:

- $\sum (fi)$  : Sumatoria de los valores de los factores de ajuste.  
 $PF$  : Punto de funcion  
 $CT$  : Conteo total

Con el límite superior establecido en la suma de los factores de ajuste de acuerdo a los parámetros propuestos, se realiza el cálculo del valor siguiente en la secuencia, este proceso es fundamental para mantener la coherencia en la evaluación de la complejidad del sistema.

**( 9 ) Punto Función Máxima**

$$PF_{max} = CT \left[ 0.65 + (0.01 * \sum fi) \right] \quad (9)$$

Donde:

$\sum (fi)$  : Sumatoria de los valores de los factores de ajuste.

PF<sub>max</sub>. : Punto función máxima.

Con estos factores podemos encontrar el valor de la funcionalidad.

**( 10 ) Formula de la Funcionalidad**

$$F = \frac{PF}{PF_{max}} * 100 \quad (10)$$

Donde:

F : Funcionalidad

PF : Punto función.

PF<sub>max</sub> : Punto función máxima.

**2.10.1.2 Confiabilidad**

La confiabilidad de un sistema se refiere a su capacidad para mantener un nivel de funcionamiento adecuado durante un período de prueba establecido que implica considerar aspectos como la madurez del sistema. Aplicaremos las fórmulas correspondientes, teniendo en cuenta las probabilidades de fallas y éxitos debido a varios factores.

- Comportamiento temporal. Grado en que un producto o realiza sus funciones de forma que el tiempo de respuesta y la ratio de rendimiento cumple los requisitos especificados.
- Utilización de recursos. Grado en que la cantidad y tipos de recursos utilizados por el producto al llevar a cabo su función bajo condiciones determinadas no exceden lo especificado.
- Capacidad. Grado en que el producto cumple los requisitos relativos a límites máximos para un parámetro (ítems almacenados, usuarios concurrentes, ancho de banda de comunicaciones).

Considerando que se tiene en cuenta que:

**( 11 ) Probabilidad de Fallas**

$$P(T \leq t) \quad \Rightarrow \quad PF \quad (11)$$

Probabilidad de trabajo sin fallas

**( 12 ) Probabilidad de Trabajo Sin Fallas**

$$P(T > t) = 1 - F(t) \Rightarrow \text{Probabilidad de trabajo sin fallas} \quad (12)$$

Se considera la siguiente función al calcular la confiabilidad del sistema.

**( 13 ) Función de la Confiabilidad del Sistema MT**

$$F(t) = f * e^{-\mu * t} \quad (13)$$

Donde:

$f$	=	Funcionalidad del sistema
$\mu$	=	Probabilidad de error del sistema
$t$	=	Tiempo de prueba del sistema
$F(t)$	=	Confiabilidad

### 2.10.1.3 Usabilidad

Al evaluar la usabilidad de un sistema, es importante considerar el factor humano, asegurando que el software cumpla con los requisitos y expectativas del usuario, por lo tanto, se lleva a cabo una evaluación mediante encuestas dirigidas a los usuarios del sistema.

Estas encuestas permiten recopilar información valiosa sobre la experiencia del usuario, su satisfacción y la facilidad de uso percibida que se utiliza luego para calcular la usabilidad del sistema, aplicando una fórmula diseñada específicamente para medir la eficacia y la experiencia del usuario en la interacción con el software.

#### ( 14 ) Función para Determinar la Usabilidad MT

$$FU = \left( \frac{\sum \text{valor}}{n} * 100 \right) / 5 \quad (14)$$

Donde:

$FU$	=	Función usabilidad
$\sum \text{valor}$	=	Sumatoria de los valores de la usabilidad del sistema
$n$	=	Cantidad de valores de la usabilidad del sistema

### 2.10.1.4 Eficiencia

Capacidad del producto software para que el usuario interactúe mediante su interfaz intercambiando información para completar determinadas tareas.

Para determinar la eficiencia, se emplea la fórmula siguiente.

#### ( 15 ) Fórmula de la Eficiencia MT

$$E = \frac{\sum x_i}{n} * \frac{100}{n} \quad (15)$$

Donde:

$$\begin{aligned} \sum x_i &= \text{Sumatoria de los valores de eficiencia} \\ n &= \text{Número de preguntas} \\ E &= \text{Eficiencia} \end{aligned}$$

### **2.10.1.5 Mantenibilidad**

Esta métrica evalúa la cantidad de trabajo requerido para realizar cambios en el sistema, ya sea para corregir errores o agregar nuevas funcionalidades. El estándar IEEE94 recomienda utilizar el índice de madurez del sistema como indicador de su estabilidad. Por lo tanto, la ecuación para este índice es la siguiente:

*( 16 ) Cálculo del Nivel de Madurez del Sistema*

$$IMS = \frac{[Mt - (Fa + Fc + Fd)]}{Mt} \quad (16)$$

Donde:

$$\begin{aligned} Mt &= \text{Número de módulos total de la versión actual} \\ Fa &= \text{Número de módulos de la versión actual que se añadieron.} \\ Fc &= \text{Número de módulos de la versión actual que se cambiaron.} \\ Fd &= \text{Número de módulos de la versión anterior que se eliminaron} \\ &\quad \text{en la versión actual} \\ IMS &= \text{Nivel de madurez del sistema} \end{aligned}$$

### **2.10.1.6 Portabilidad**

La capacidad de un software para ser transferido de un entorno a otro incluye varios aspectos importantes:

### **Adaptabilidad**

Habilidad del software para ajustarse a diferentes entornos sin requerir modificaciones adicionales.

### **Facilidad de Instalación**

Esfuerzo requerido para instalar el software en un entorno específico.

### **Conformidad**

Verificación de que el software cumple con estándares o convenciones de portabilidad.

### **Capacidad de sustitución**

Facilidad y esfuerzo necesarios para sustituir el software con otro producto que tenga funciones similares.

El sistema, desarrollado con Laravel, puede implementarse en cualquier servidor con Apache y las herramientas necesarias.

Como la tecnología web es ejecuta fácilmente en cualquier dispositivo con conexión a Internet y un navegador.

#### **( 17 ) Fórmula de la Portabilidad MT**

$$Portabilidad = 1 - \left( \frac{NAS}{NIS} \right) \quad (17)$$

## **2.11 SEGURIDAD**

La información es un activo valioso que puede hacer crecer o destruir a su organización. Cuando se gestiona apropiadamente, le permite operar con confianza. La gestión de seguridad de la información le da la libertad de crecer, innovar y ampliar su base de clientes con el conocimiento que toda su información confidencial permanecerá de esa manera.

Seguridad es la capacidad de dar respuesta efectiva a riesgos, amenazas o vulnerabilidades y estar preparados para prevenirlos, contenerlos y enfrentarlos. El término, identifica a una situación de ausencia o disminución de riesgos para un determinado entorno social y natural. (entiéndase riesgo como el daño o pérdida que pudiera causar un particular evento, encontrándose en función de la intensidad de la amenaza y del grado de vulnerabilidad que experimente la sociedad en el momento determinado; esta vulnerabilidad podrá disminuir si el Estado o la comunidad poseen determinadas capacidades para enfrentarla).

### **2.11.1 ISO 27001**

La norma ISO 27001 es un estándar internacional que establece un marco para la gestión para la seguridad de la información en las organizaciones. Según la Organización Internacional de Normalización (ISO), esta norma "define requisitos para establecer, implementar, mantener y mejorar continuamente un sistema de gestión de la seguridad de la información Dentro del contexto de la organización" (ISO, 2013, p. 1).

Su objetivo principal es salvaguardar la confidencialidad, integridad y disponibilidad de la información a través de la aplicación de controles y procesos de gestión de riesgos.

Según Calder & Watkins (2015), una de las características fundamental de la ISO 27001 es su enfoque en el ciclo de mejora continua PDCA (Planificar, Hacer, Verificar, Actuar). Este ciclo permite a las organizaciones evaluar y mejorar continuamente su sistema de gestión de seguridad de la información (SGSI) para adaptarse a los cambios en los requisitos y a nuevas amenazas. El objetivo esencial de la norma ISO 27001 es proporcionar un marco de gestión para establecer, implementar, operar, monitorear, revisar, mantener y mejorar un SGSI dentro de una organización, centrado en preservar tres aspectos clave de la seguridad de la información: confidencialidad, integridad y disponibilidad.

- **Confidencialidad:** La ISO 27001 establece controles y medidas para asegurar que la información no sea accedida, divulgada o revelada a personas, entidades o procesos no autorizados. "La confidencialidad se refiere a la protección de la información confidencial contra su divulgación no autorizada" (Calder, 2015, p. 18).
- **Integridad:** La norma busca garantizar que la información no sea modificada, alterada o destruida de manera no autorizada, ya sea de forma accidental o deliberada. "La integridad se refiere a la protección de la exactitud y completitud de la información y los métodos de procesamiento" (Diesterer, 2013, p. 96).
- **Disponibilidad:** La ISO 27001 resalta la importancia de garantizar que la información y los sistemas asociados estén accesibles para las personas, procesos y entidades autorizadas cuando sea necesario. "La disponibilidad se refiere a la protección contra la negación de acceso a la información o recursos por parte de usuarios autorizados" (Calder & Watkins, 2015, p. 19).

Al preservar estos tres pilares esenciales de la seguridad de la información, la ISO 27001 permite a las organizaciones proteger sus activos de información críticos, mantener la confianza de los clientes y socios comerciales, cumplir con las normativas legales, y reducir los riesgos asociados con posibles brechas de seguridad.

### **2.11.2 Seguridad Física**

La "seguridad física" se refiere a la identificación y análisis de las amenazas y riesgos que enfrentan o pueden llegar a enfrentar instalaciones, bienes y procesos a fin de implementar planes y sistemas tendientes a prevenir, dificultar o limitar los resultados de las posibles acciones dañinas contra la seguridad de ellas y ellos.

Seguridad física podemos entender todas aquellas mecanismos - generalmente de prevención y detección, destinados a proteger físicamente cualquier recurso del sistema; estos

recursos son desde un simple teclado hasta una cinta de backup con toda la información que hay en el sistema, pasando por la propia CPU de la máquina.

### 2.11.3 Seguridad Lógica

La seguridad lógica se define como un conjunto de procesos destinados a garantizar la seguridad en el uso de los sistemas y los programas destinados a gestionar los datos y procesos de cualquier empresa. Del mismo modo, también hace referencia al acceso autorizado y ordenado de los usuarios a la información almacenada por la compañía.

Por lo tanto, la seguridad lógica, en términos de ciberseguridad, engloba todas las medidas de administración necesarias con el objetivo de reducir al máximo los riesgos asociados a las actividades cotidianas realizadas usando tecnología de la información.

### 2.11.4 Seguridad de la Base de Datos

La seguridad de las bases de datos son los procesos, herramientas y controles que aseguran y protegen las bases de datos contra amenazas accidentales e intencionadas. El objetivo de la seguridad de las bases de datos es proteger los datos confidenciales y mantener la confidencialidad, disponibilidad e integridad de la base de datos.

- **Confidencialidad** significa que los datos, objetos y recursos están protegidos contra la visualización y otros accesos no autorizados.
- **Integridad** significa que los datos están protegidos de cambios no autorizados para garantizar que son fiables y correctos.
- **Disponibilidad** significa que los usuarios autorizados tienen acceso a los sistemas y recursos que necesitan.

### 2.11.5 Seguridad del Framework CodeIgniter

CodeIgniter es un framework de PHP que ofrece herramientas para hacer que las aplicaciones sean seguras, pero es responsabilidad del desarrollador usarlas correctamente.

Para mejorar la seguridad de las aplicaciones web PHP con CodeIgniter, se pueden implementar otras medidas, como:

- Usar HTTPS y SSL
- Realizar copias de seguridad y actualizaciones con regularidad
- Implementar la protección CSRF
- Ubicar las carpetas del sistema y de la aplicación por encima de la raíz web

#### Restricciones en las URL

CodeIgniter limita los caracteres que se pueden usar en las cadenas de URL para evitar que se pasen datos maliciosos a la aplicación. Las URL solo pueden contener texto alfanumérico, tilde, signo de porcentaje, punto, dos puntos, guion bajo y guion.

#### Función `xss_clean()`

Esta función filtra posibles ataques XSS, pero solo elimina los atributos de las etiquetas HTML que se cerraron correctamente.

## 2.12 PRUEBAS DE SOFTWARE

Las pruebas de software son el proceso de evaluar y verificar que un producto o aplicación de software hace lo que se supone que debe hacer. Entre los beneficios de unas buenas pruebas se incluyen la prevención de errores y la mejora del rendimiento.

### **2.12.1 Pruebas de Caja Blanca**

El objetivo de las pruebas de caja blanca es realizar pruebas que cubran la estructura interna de un sistema, por estructura interna nos referimos a código, arquitectura y flujos de trabajo. En las pruebas de caja blanca, el código es visible para los testers, por lo que también se denominan pruebas de caja transparente o pruebas de caja abierta.

La profundidad de las pruebas de caja blanca se puede medir a través de la cobertura estructural. La cobertura estructural es la medida que se utiliza para saber si todas las rutas dentro de un módulo se han ejecutado al menos una vez (Osmosis, 2024, p. 10).

### **2.12.2 Pruebas de Caja Negra**

Las pruebas de caja negra, una forma de prueba que se realiza sin conocimiento de los componentes internos de un sistema, se pueden realizar para evaluar la funcionalidad, la seguridad, el rendimiento y otros aspectos de una aplicación. Análisis dinámico de código es un ejemplo de pruebas automatizadas de seguridad de caja negra. Los evaluadores de caja negra definen casos de prueba e interactúan con el software como lo haría un usuario para validar que hace lo que debería, como debería (Zaptest, 2024, p. 7).

### **2.12.3 Pruebas de Estrés**

Una prueba de estrés (stress) consiste en probar los límites que un sistema puede soportar. En este tipo de pruebas se suele enviar más peticiones de las que el software podría atender normalmente para saber el comportamiento de la aplicación.

Por otro lado, las pruebas de estrés (stress testing) son realizadas sobrecargando un sistema más allá de sus especificaciones, para verificar cómo y cuándo fallará. Dentro de informática podemos colocar una gran carga en la base de datos, entradas (peticiones) continuas al sistema o almacenar información más allá de la capacidad de memoria del sistema. (Focus, 2022, p. 7).

## 2.12.4 Pruebas de Accesibilidad

- **Axe (Axe DevTools)**

Esta herramienta es de utilidad para inspeccionar la estructura de la página web y verificar si la misma cumple con todas las pautas de la WCAG y otros estándares existentes ya mencionados. Para utilizarla en Google Chrome, simplemente se debe instalar la extensión desde el Chrome Web Store ([link](#)). Para ejecutarla se debe abrir la consola de desarrollador y seleccionar la pestaña “Axe DevTools”. La herramienta brinda distintas opciones de validación, es posible escanear y validar toda la página web o un elemento específico. Por ejemplo, si se selecciona la opción de validar toda una página, rápidamente se genera un reporte completo de todos los hallazgos de accesibilidad que contiene el sitio, categorizados en distintos niveles de criticidad.

- **Colour Contraser Analyser (CCA)**

Las pruebas que se realizan utilizando la herramienta CCA son fundamentales para simular situaciones de personas con baja sensibilidad al contraste de colores. Esta condición puede presentarse en personas en situación de discapacidad visual como, por ejemplo, algunas de las variantes del daltonismo. A su vez, este tipo de situación se vuelve más común en personas mayores de edad y puede verse empeorada con el pasar de los años.

- **WAVE (Website Accessibility Versatile Evaluator )**

WAVE es una suite de aplicaciones que ayuda a los diseñadores, creadores y autores de sitios web a garantizar la accesibilidad de sus contenidos. Esta suite puede ser utilizada por empresas de todos los tamaños, y también dispone de una herramienta de seguimiento e informes a nivel empresarial llamada herramienta de accesibilidad Pop Tech. WAVE tiene previsto ofrecer una API pública completa que permitirá a los programadores acceder a todas las funciones clave.

Los usuarios pueden emplear extensiones de Firefox o Chrome para utilizar el programa y comprobar fácilmente páginas individuales. WAVE incorpora comentarios directamente en el contenido del sitio web con detalles sobre posibles errores.

### 2.12.5 Pruebas de Unitarias

Las pruebas unitarias son un recurso viable para estos efectos, al aplicarse con el propósito de identificar errores, vulnerabilidades o funciones defectuosas. A continuación, explicamos a detalle qué son, sus características y las ventajas de su implementación.

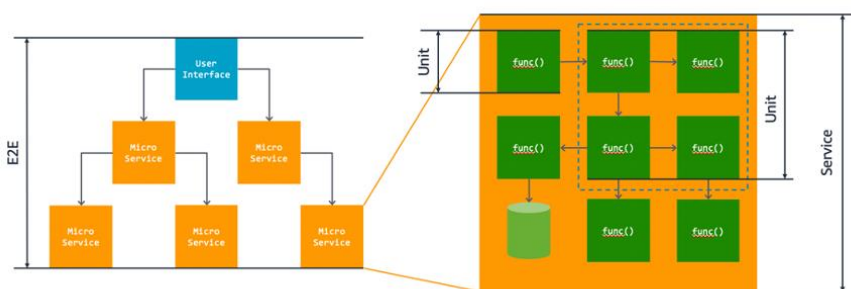
Las pruebas de software son un recurso fundamental para los desarrolladores, quienes recurren a ellas para asegurar la calidad, funcionamiento y rendimiento de un producto informático.

De acuerdo con un informe del Comité Internacional de Certificaciones de Pruebas de Software (ISTQB, por sus siglas en inglés), la detección de bugs, el correcto desempeño de un sistema y el aumento de la confianza con los clientes son los principales beneficios que ofrecen las actividades de testing para una organización.

El 28% de los encuestados aseguró realizar una inspección de código fuente antes de ejecutar una prueba de software, lo cual implica la revisión del lenguaje de programación mediante prácticas específicas.

**Figura 9**

*Funciones de la Pruebas Unitarias*



# **CAPÍTULO III**

## **MARCO**

### **APLICATIVO**

**INGENIERÍA**  
**DE SISTEMAS**  
UNIVERSIDAD PÚBLICA DE EL ALTO



### 3 CAPÍTULO III MARCO APLICATIVO

#### 3.1 INTRODUCCIÓN

En este capítulo se hará un análisis de los requerimientos funcionales y no funcionales, se aplicará la metodología UWE, con todas las fases y diagramas correspondientes, con el fin de identificar los requerimientos que se precisan en el sistema, también se identificarán los principales actores.

#### 3.2 DESARROLLO DE LA METODOLOGÍA

En la aplicación de la metodología obtendremos los requerimientos funcionales y no funcionales, se debe obtener la información de forma clara y precisa, a través de un análisis profundo, requisitos que necesita el sistema, esto con el fin de tener el correcto desarrollo de este capítulo, aplicando las herramientas descritas.

**Tabla 3**

*Herramientas para la Obtención de Requerimientos*

HERRAMIENTA	CARACTERÍSTICAS
<b>Observación</b>	Se hizo la observación, en las oficinas de la Unidad correspondiente, viendo los pasos a seguir durante el registro.
<b>Documentación</b>	Se revisaron los documentos que se manejan dentro del proceso, el tipo de presentación, las normas que se debe seguir y los formatos para la realización de los reportes.
<b>Entrevistas</b>	Se realizó entrevistas con el personal involucrado dentro del proceso, quienes son: Personal administrativo Personal técnico

### 3.2.1 Requerimientos Funcionales

Se describen los requisitos mínimos que necesita el sistema para los diferentes procesos, a continuación, en la siguiente tabla se da los detalles de los requerimientos funcionales.

**Tabla 4**  
*Requerimientos Funcionales*

ROL	DESCRIPCIÓN	FUNCIÓN
F-1	Acceder al sistema	Accede al sistema 3 tipos de usuarios (Administrador, Personal Operativo, secretaria), mediante su usuario y contraseña
F-2	Gestionar Usuarios	Permite registrar, editar, listar, cambio de contraseña, dar de baja a usuarios
F-3	Administración del software	Permite al usuario con el perfil de Administrador o director de la Unidad, tener acceso total a los módulos y operaciones de la gestión de la información: creación, listado, actualización, inactivación de los recursos del sistema.
F-4	Gestionar los pedidos	Permite registrar, editar, listar todos los clientes y sus pedidos en la unidad correspondiente.
F-5	Gestionar registros de costos de producción	Permite registrar, editar, listar los costos incurridos que se realizará en la unidad.
F-6	Gestionar registro de pedidos	Permite registrar los pedidos acordes a las especificaciones del cliente.
F-7	Administrar roles y permisos	Administra los roles y permisos que se otorga a los usuarios.
F-8	Generar reportes de usuarios.	Permite generar un reporte pdf de los usuarios de acuerdo al rol especificado.
F-9	Generar Clasificación de Costos	Permite clasificar los costos de acuerdo al rol especificado.

<b>F-10</b>	Generar cálculo de Utilidades	Permite calcular las utilidades de acuerdo al rol especificado.
<b>F-11</b>	Generar libro diario	Permite generar los libros diarios de acuerdo al rol especificado.
<b>F-12</b>	Generar datos estadísticos	Permite mostrar los datos estadísticos de acuerdo al rol especificado.

### 3.2.2 Requerimientos No Funcionales

Los Requerimientos no funcionales (RFN), se describen cualidades y características del sistema que el usuario puede visualizar. A continuación, se muestra la siguiente tabla:

**Tabla 5**  
*Requerimientos No Funcionales*

<b>ROL</b>	<b>FUNCIÓN</b>
<b>RNF-1</b>	El Software debe ser compatible en todos los navegadores, debe funcionar correctamente en: Google Chrome, Mozilla Firefox, IE/Edge y otros.
<b>RNF-2</b>	La interfaz del sistema debe ser intuitiva y fácil de usar, asegurando una experiencia óptima para el usuario y facilitando su interacción de manera eficiente y sin complicaciones.
<b>RNF-3</b>	El sistema debe incluir un menú principal con opciones desplegadas o submenús para facilitar la navegación del usuario.
<b>RNF-4</b>	El mantenimiento y soporte periódico del sistema garantizan un rendimiento óptimo, prolongando su vida útil y evitando posibles fallos, lo que mejora su eficiencia y funcionalidad general.
<b>RNF-5</b>	El sistema deberá responde a todas las solicitudes del usuario, con toda disponibilidad.
<b>RNF-6</b>	El sistema debe proteger las contraseñas de los usuarios mediante un proceso de encriptación, asegurando que los datos confidenciales permanezcan seguros.
<b>RNF-7</b>	La sesión del usuario estará activa durante un tiempo específico,

	después del cual se cerrará automáticamente para garantizar mayor seguridad y protección de la información.
<b>RNF-8</b>	El sistema debe ser escalable a agregar nuevos módulos y funciones a futuro.
<b>RNF-9</b>	Los cambios realizados en la base de datos deben estar actualizados para todos los usuarios según los permisos definidos
<b>RNF-10</b>	Conexión a Internet
<b>RNF-11</b>	El sistema debe incluir manuales de usuario bien estructurados.
<b>RNF-12</b>	La sesión de usuario tendrá un tiempo de actividad definido.
<b>RNF-13</b>	Solo el administrador de acceso a datos podrá modificar los permisos de acceso al sistema.
<b>RNF-14</b>	El sistema necesita un menú lateral para facilitar navegación.
<b>RNF-15</b>	Conexión a Internet

### 3.2.3 Descripción de Actores

De acuerdo a los principales actores que interactuarán con el sistema, se debe identificar a estos, de acuerdo a los roles y permisos que se registren en este proyecto de Sistema de Información Web, para la empresa “Kubic textiles”. área de producción.

**Tabla 6**

*Descripción de Actores*

DESCRIPCIÓN	RESPONSABILIDADES
<b>Administrador, responsable del Área contable</b>	Administra a los usuarios en el sistema web, asignando roles y permisos adecuados. Además, gestiona las configuraciones del sistema para garantizar su correcto funcionamiento y optimización.
<b>Personal Operativo de la unidad (producción)</b>	Es responsable de registrar las peticiones y especificaciones, además de supervisar y gestionar las solicitudes del cliente, asegurando un seguimiento adecuado en el proceso de producción del pedido.

---

**Técnico**

Responsable de registrar y mantener un seguimiento actualizado de los pedidos garantizando el cumplimiento de la producción.

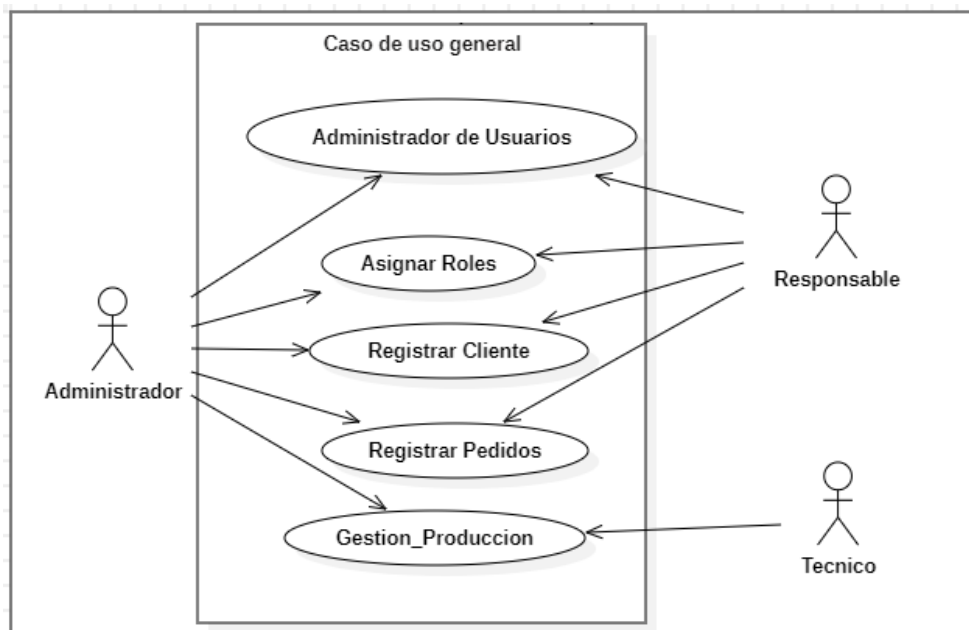
---

### 3.2.4 Diagramas de Caso de Uso General

El diagrama de caso de uso general del sistema es una parte esencial del análisis y diseño que proporciona una visión global de las interacciones entre usuarios y sistema, simplificando la comprensión de las funciones, esta representación visual estructurada de las funciones disponibles no solo facilitara la comunicación entre los desarrolladores y las partes interesadas, sino que también guía la planificación y el diseño detallado.

**Figura 10**

*Caso de Uso General del Sistema*



### 3.2.5 Diagrama de Caso de Uso Específico

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

Figura 11

Gestión de Usuarios

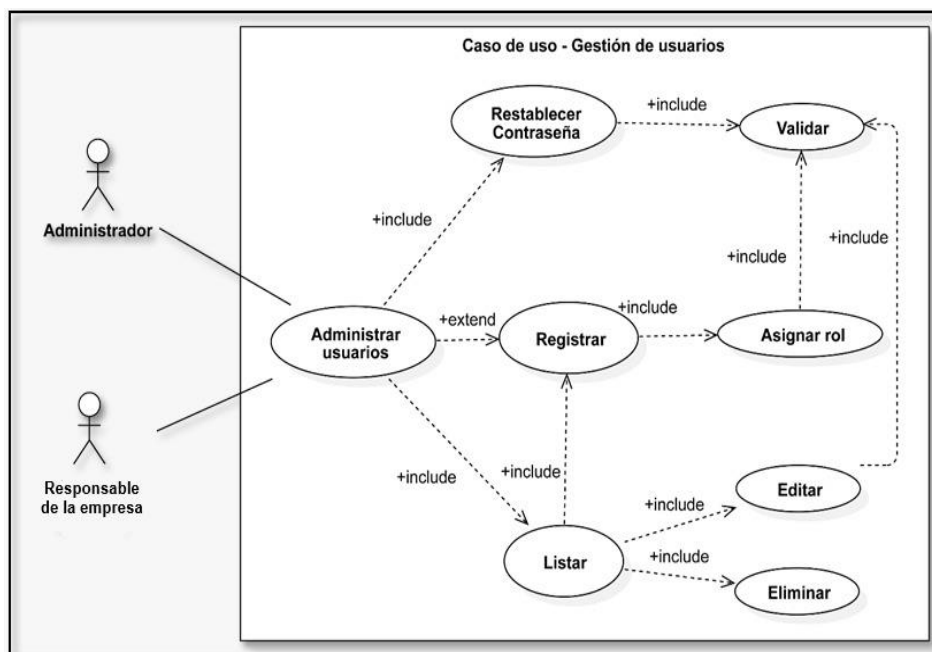


Tabla 7

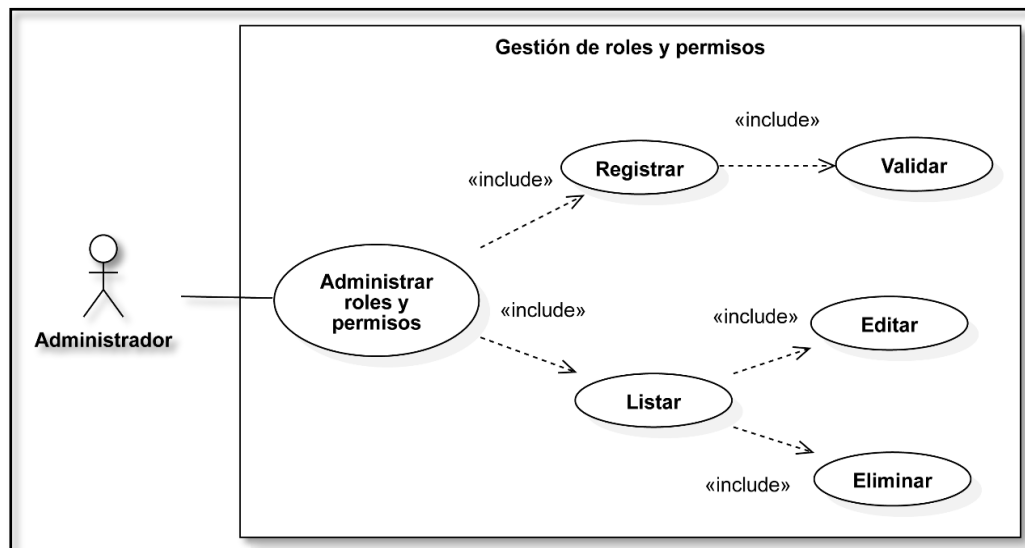
Detalle del Caso de Uso – Gestión de Usuarios.

CASO DE USO	GESTIÓN DE USUARIOS
IDENTIFICADOR	CU – 1
TIPO	Primario
ACTORES	Administrador, responsable de la empresa “kubic textiles”
REQUISITOS PREVIOS	El usuario debe registrarse, autenticarse e ingresar al panel de administración.
DESCRIPCIÓN	El proceso comienza cuando el usuario selecciona la opción de usuarios, visualiza la lista de usuarios y realiza acciones como crear, modificar y/o activar/desactivar a los usuarios, seguidamente el sistema debe verificar la información ingresada.

## Diagrama de Caso de Uso: Gestión de Roles y Permisos

**Figura 12**

*Gestión de Roles y Permisos*



**Tabla 8**

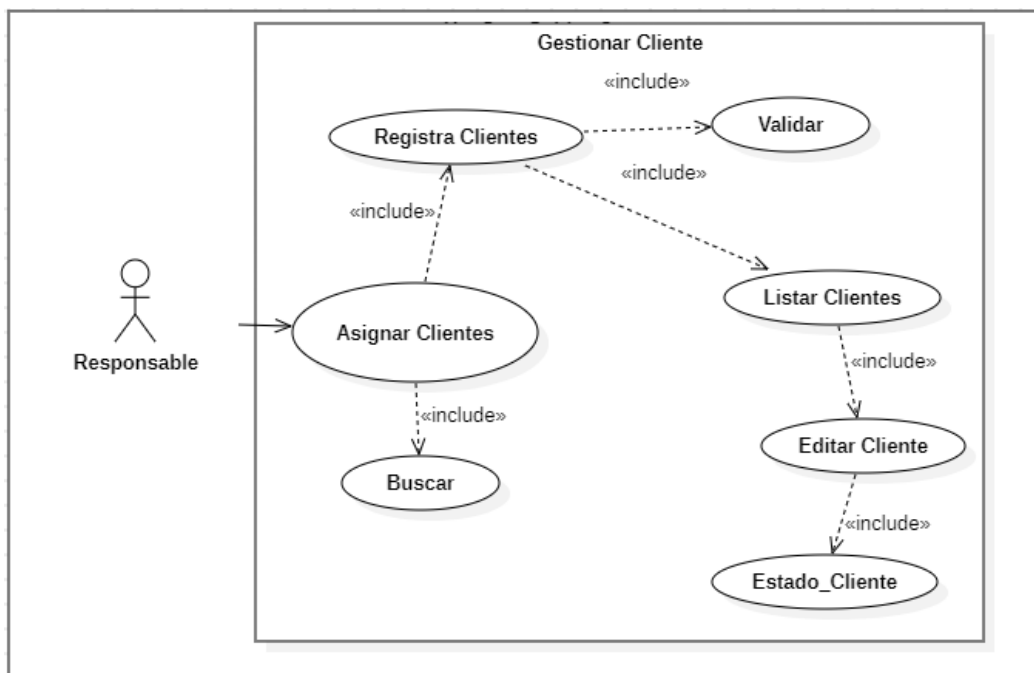
*Detalle del Caso de Uso – Gestión de Roles y Permisos*

CASO DE USO	GESTIÓN DE ROLES Y PERMISOS
IDENTIFICADOR	CU – 2
TIPO	Primario
ACTORES	Administrador, responsable de la empresa “kubic textiles”
REQUISITOS PREVIOS	El usuario debe registrarse, autenticarse e ingresar al panel de administración.
DESCRIPCIÓN	El proceso inicia cuando el usuario selecciona la opción de roles y permisos, visualiza la lista de roles y ejecuta acciones como crear nuevos roles, asignando los permisos correspondientes, modificar los roles existentes, seguidamente el sistema verifica la información ingresada.

## Diagrama de Caso de Uso: Gestión de clientes

**Figura 13**

*Gestión de Clientes*



**Tabla 9**

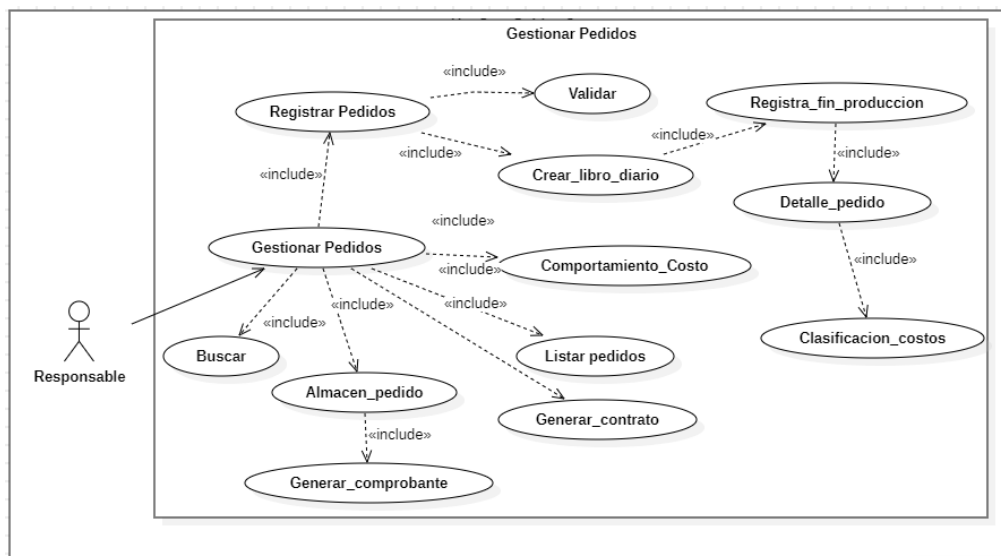
*Detalle del Caso de Uso – Gestión de Clientes*

CASO DE USO	GESTIÓN DE CLIENTE
IDENTIFICADOR	CU – 3
TIPO	Primario
ACTORES	Administrador, responsable de la empresa “kubic textiles”
REQUISITOS PREVIOS	El usuario debe registrarse, autenticarse e ingresar al panel de administración.
DESCRIPCIÓN	El proceso inicia cuando el usuario selecciona los clientes, visualiza un formulario para la respectiva gestión, este se enlista la cual este proceso y ejecuta acciones como crear, modificar y eliminar.

## Diagrama de Caso de Uso: Gestión de Pedidos

**Figura 14**

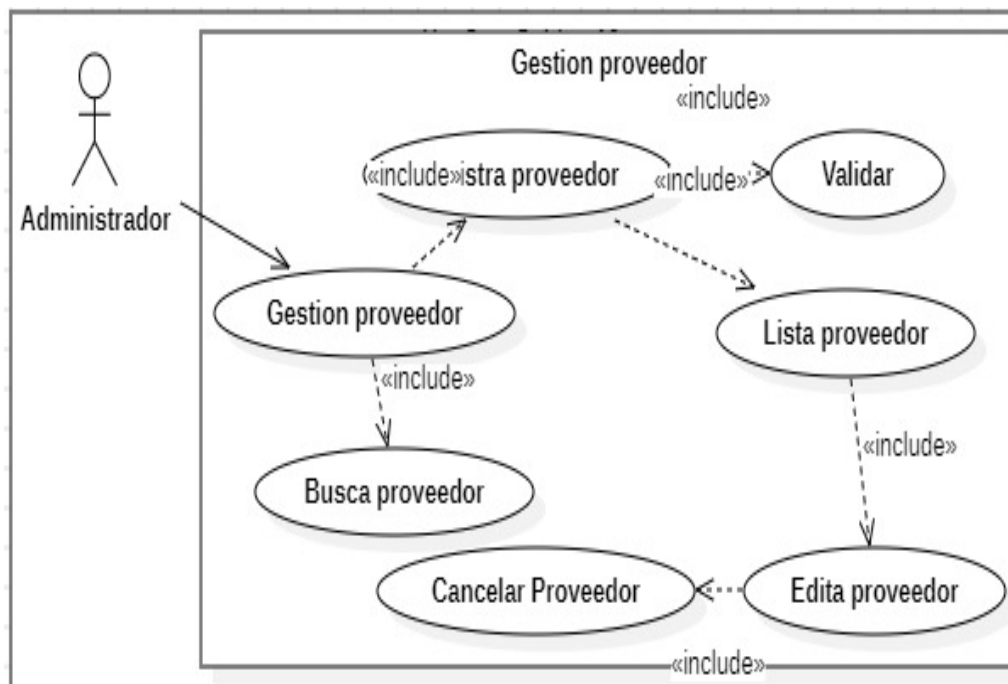
*Gestión de Pedidos*



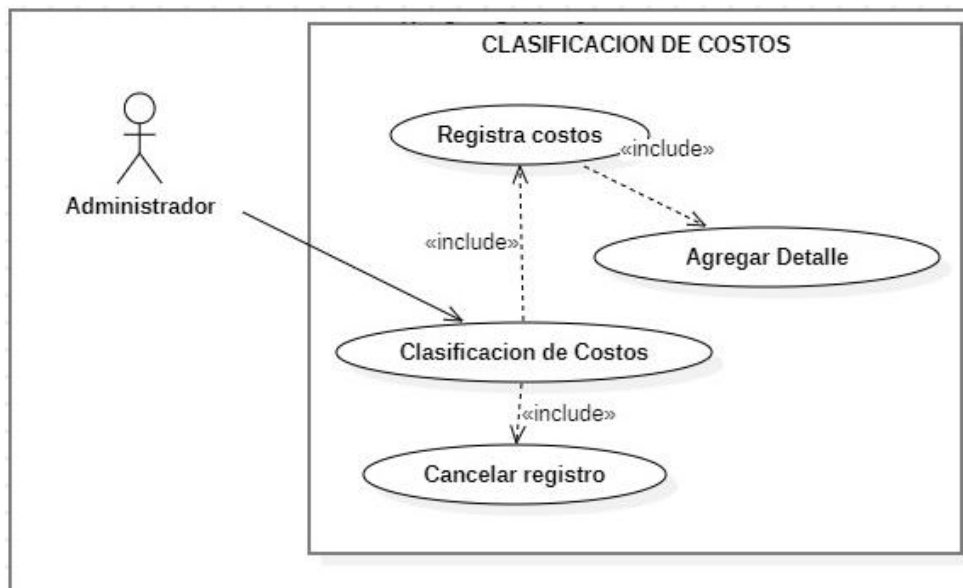
**Tabla 10**

*Detalle del Caso de Uso – Gestión de Pedidos*

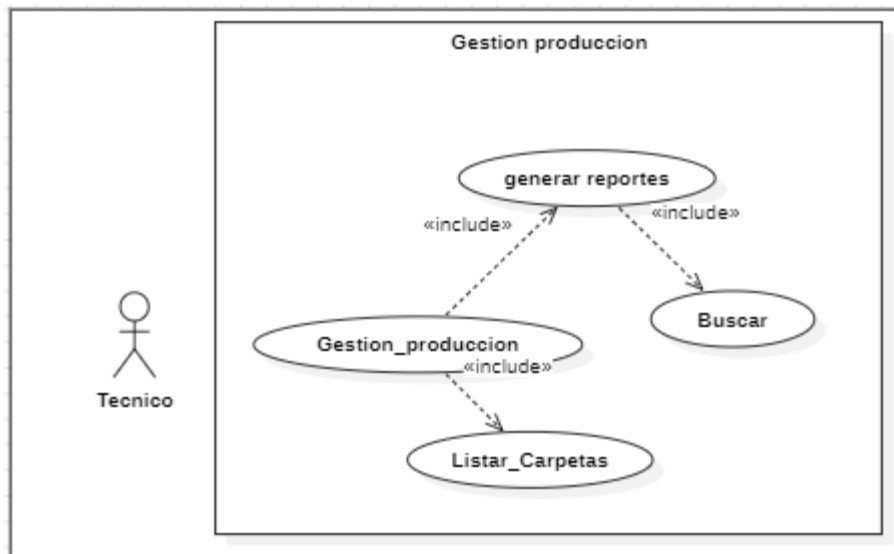
CASO DE USO	GESTIÓN DE PEDIDOS
IDENTIFICADOR	CU – 4
TIPO	Primario
ACTORES	Responsable de la empresa “kubic textiles”
REQUISITOS PREVIOS	El usuario debe registrarse, autenticarse e ingresar al panel de administración.
DESCRIPCIÓN	El proceso inicia cuando el usuario selecciona la opción pedidos, visualiza un formulario para la respectiva gestión de pedidos, se enlista este proceso y ejecuta acciones como crear pedidos, modificar y eliminar.

**Figura 15***Gestión Proveedor***Tabla 11***Detalle de Caso de Uso - Gestión Proveedor*

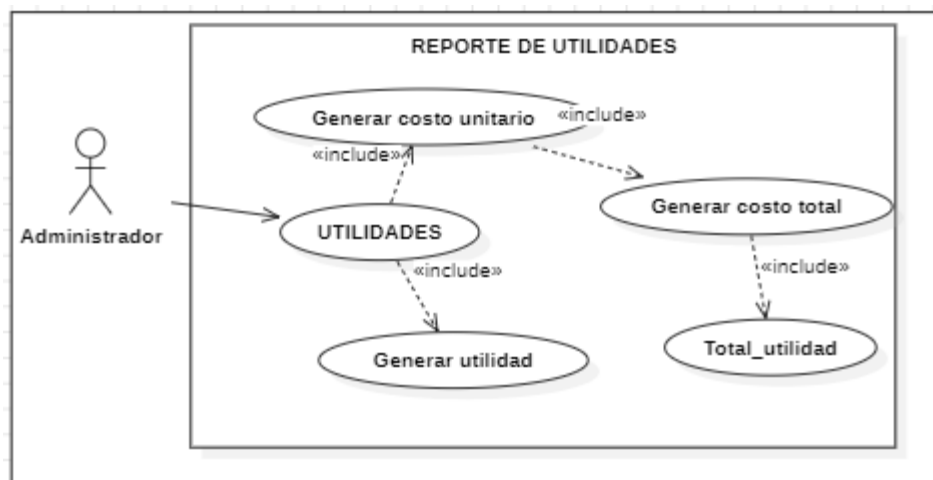
CASO DE USO	GESTIÓN DE PROVEEDOR
IDENTIFICADOR	CU – 5
TIPO	Primario
ACTORES	Responsable de la empresa “kubic textiles”
REQUISITOS PREVIOS	El usuario debe registrarse, autenticarse e ingresar al panel de administración.
DESCRIPCIÓN	El proceso inicia cuando el usuario selecciona la opción, visualiza un formulario del proveedor, se enlista este proceso y ejecuta acciones como crear, modificar.

**Figura 16***Gestión Clasificación Costos***Tabla 12***Detalle de Caso de Uso – Clasificación de Costos*

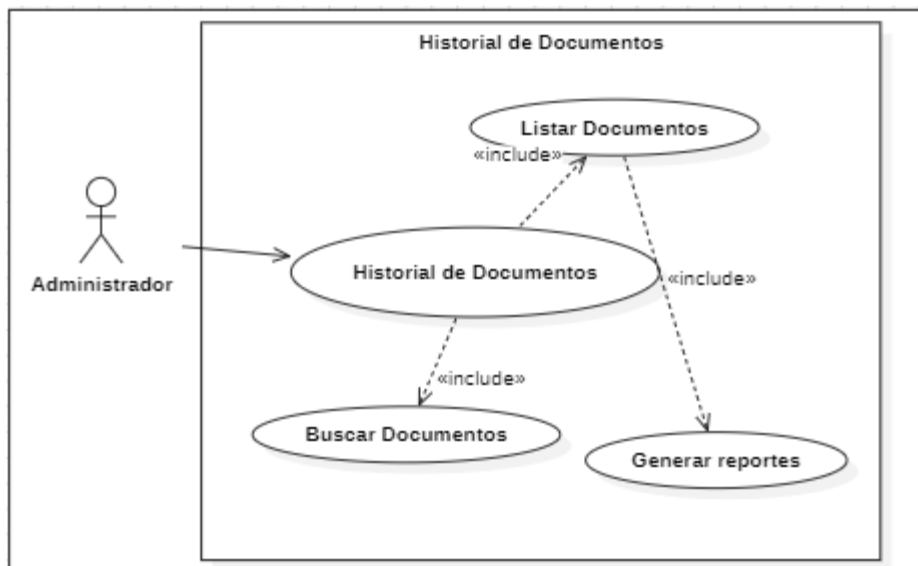
CASO DE USO	GESTIÓN DE CLASIFICACIÓN DE COSTOS
IDENTIFICADOR	CU –6
TIPO	Primario
ACTORES	Responsable de la empresa “kubic textiles”
REQUISITOS PREVIOS	El usuario debe registrarse, autenticarse e ingresar al panel de administración.
DESCRIPCIÓN	El proceso inicia cuando el usuario selecciona la opción, visualiza un formulario de costos, se enlista este proceso y ejecuta acciones como crear, modificar.

**Figura 17***Gestión de Producción***Tabla 13***Detalle de Caso de Uso – Gestión Producción*

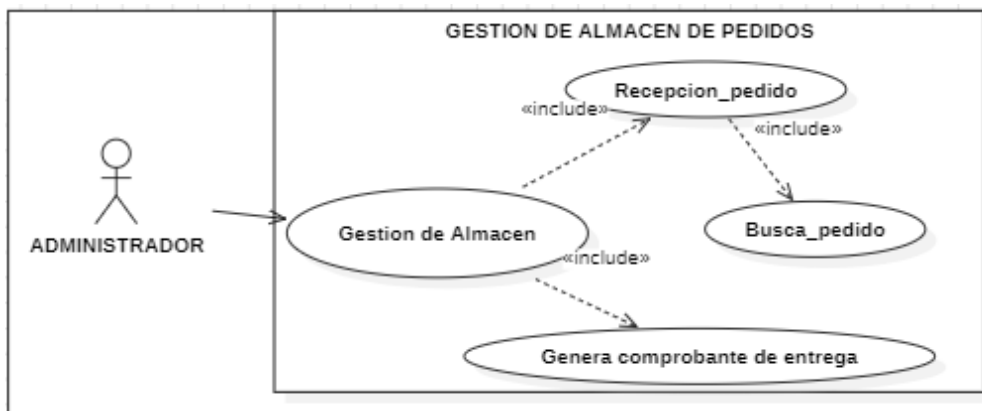
CASO DE USO	GESTIÓN DE PRODUCCIÓN
IDENTIFICADOR	CU – 7
TIPO	Primario
ACTORES	Responsable de la empresa “kubic textiles”
REQUISITOS PREVIOS	El usuario debe registrarse, autenticarse e ingresar al panel de administración.
DESCRIPCIÓN	El proceso inicia cuando el usuario selecciona la opción, visualiza un formulario de costos, se enlista este proceso y ejecuta acciones.

**Figura 18***Gestión de Utilidades***Tabla 14***Detalle de Caso de Uso - Gestión de Utilidades*

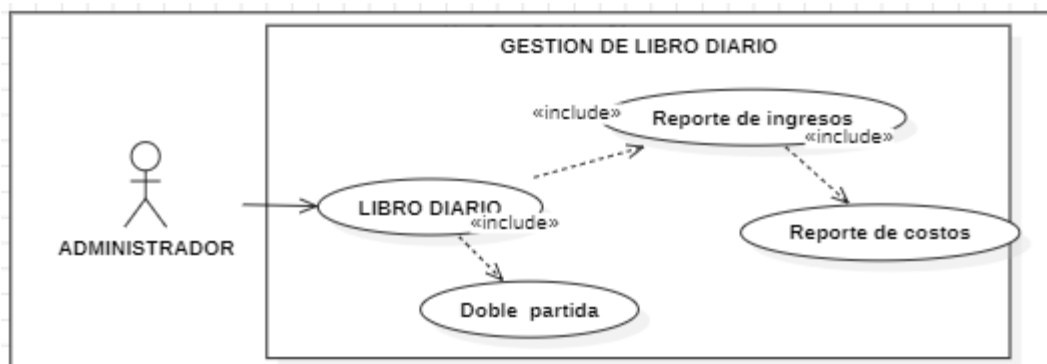
CASO DE USO	GESTIÓN DE UTILIDADES
IDENTIFICADOR	CU – 8
TIPO	Primario
ACTORES	Responsable de la empresa “kubic textiles”
REQUISITOS PREVIOS	El usuario debe registrarse, autenticarse e ingresar al panel de administración.
DESCRIPCIÓN	El proceso inicia cuando el usuario selecciona la opción, visualiza los documentos, se enlista este proceso y ejecuta acciones.

**Figura 19***Gestión Historial de Documentos***Tabla 15***Detalle de Caso de Uso – Gestión Historial de Documentos*

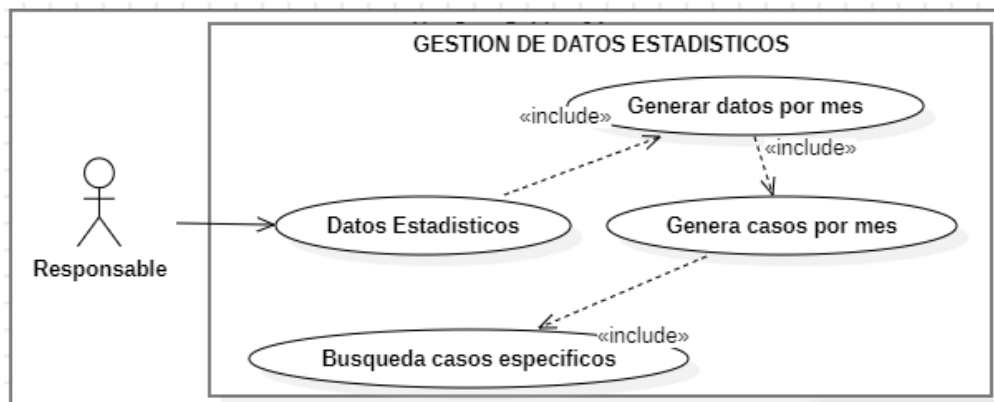
CASO DE USO	GESTIÓN DE DOCUMENTOS
IDENTIFICADOR	CU – 9
TIPO	Primario
ACTORES	Responsable de la empresa “kubic textiles”
REQUISITOS PREVIOS	El usuario debe registrarse, autenticarse e ingresar al panel de administración.
DESCRIPCIÓN	El proceso inicia cuando el usuario selecciona la opción, visualiza los documentos, se enlista este proceso y ejecuta acciones.

**Figura 20***Gestión de Almacén de Pedidos***Tabla 16***Detalle de Caso de Uso - Gestión de Almacén de Pedidos*

CASO DE USO	GESTIÓN DE ALMACÉN DE PEDIDOS
IDENTIFICADOR	CU – 10
TIPO	Primario
ACTORES	Responsable de la empresa “kubic textiles”
REQUISITOS PREVIOS	El usuario debe registrarse, autenticarse e ingresar al panel de administración.
DESCRIPCIÓN	El proceso inicia cuando el usuario selecciona la opción, visualiza las utilidades, se enlista este proceso y ejecuta acciones.

**Figura 21***Gestión de Libro Diario***Tabla 17***Detalle de Caso de Uso - Gestión Libro Diario*

CASO DE USO		GESTIÓN DE LIBRO DIARIO
IDENTIFICADOR	CU – 11	
TIPO	Primario	
ACTORES	Responsable de la empresa “kubic textiles”	
REQUISITOS PREVIOS	El usuario debe registrarse, autenticarse e ingresar al panel de administración.	
DESCRIPCIÓN	El proceso inicia cuando el usuario selecciona la opción, libro diario, se enlista este proceso y ejecuta acciones.	

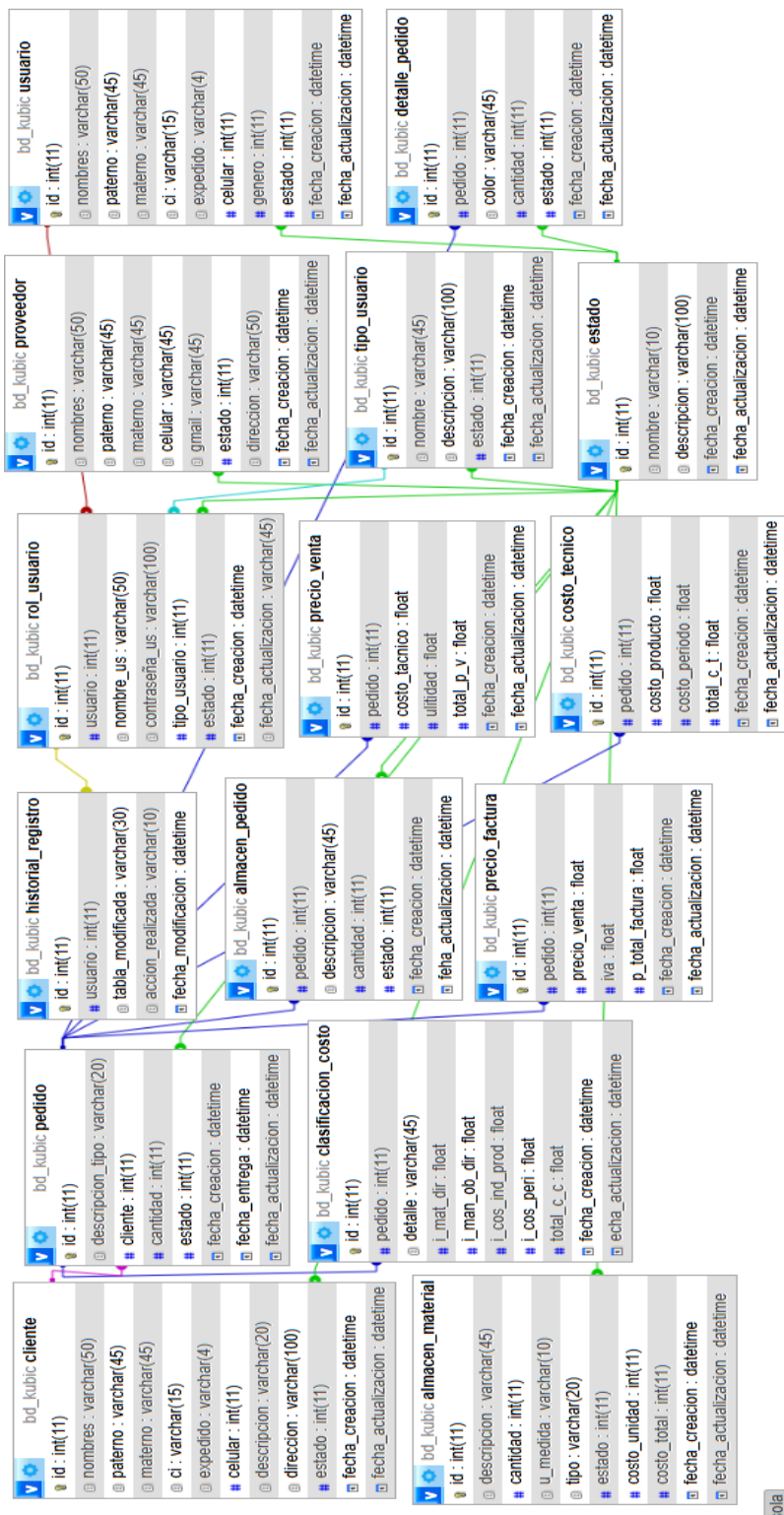
**Figura 22***Gestión de Datos Estadísticos***Tabla 18***Detalle de Caso de Uso - Gestión de Datos Estadísticos*

CASO DE USO		GESTIÓN DE DATOS ESTADÍSTICOS
IDENTIFICADOR	CU – 12	
TIPO	Primario	
ACTORES	Responsable de la empresa “kubic textiles”	
REQUISITOS PREVIOS	El usuario debe registrarse, autenticarse e ingresar al panel de administración.	
DESCRIPCIÓN	El proceso inicia cuando el usuario selecciona la opción, datos estadísticos, se enlista este proceso y ejecuta acciones.	

### 3.2.6 Modelo Conceptual

Figura 23

Diseño de la Base de Datos Relacional



### 3.2.7 Modelo de Presentación

Figura 24

Modelo de Presentación - Login

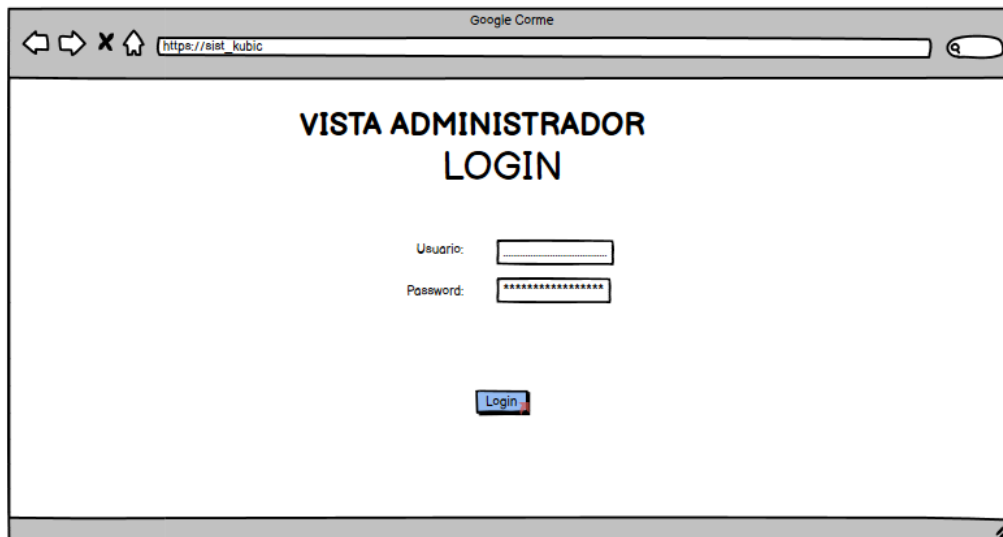
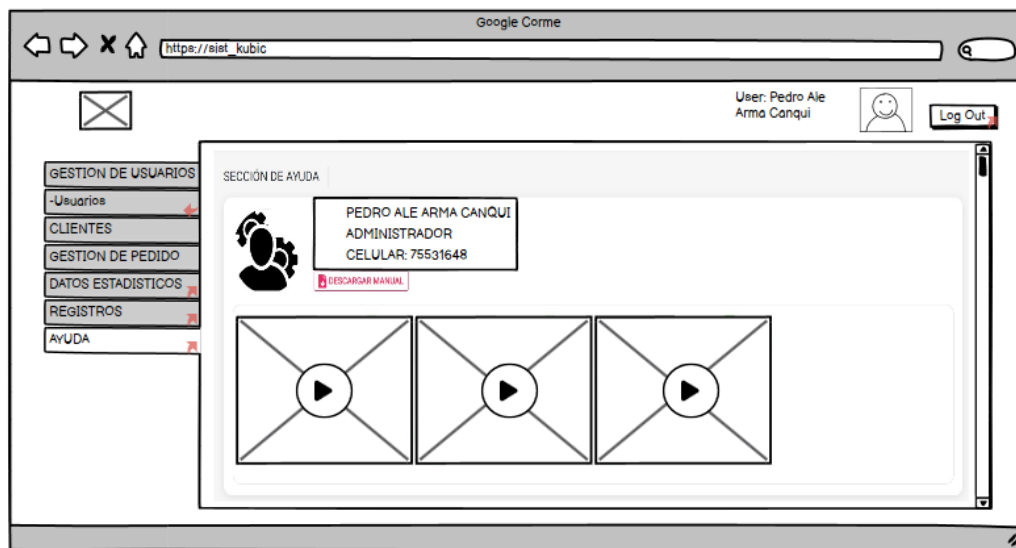
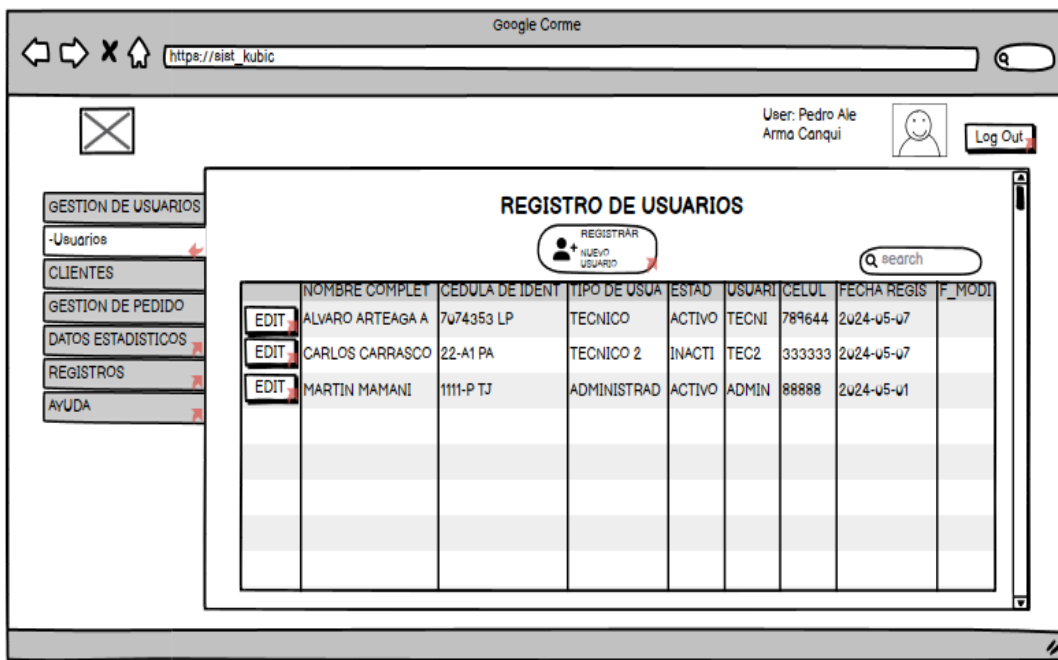


Figura 25

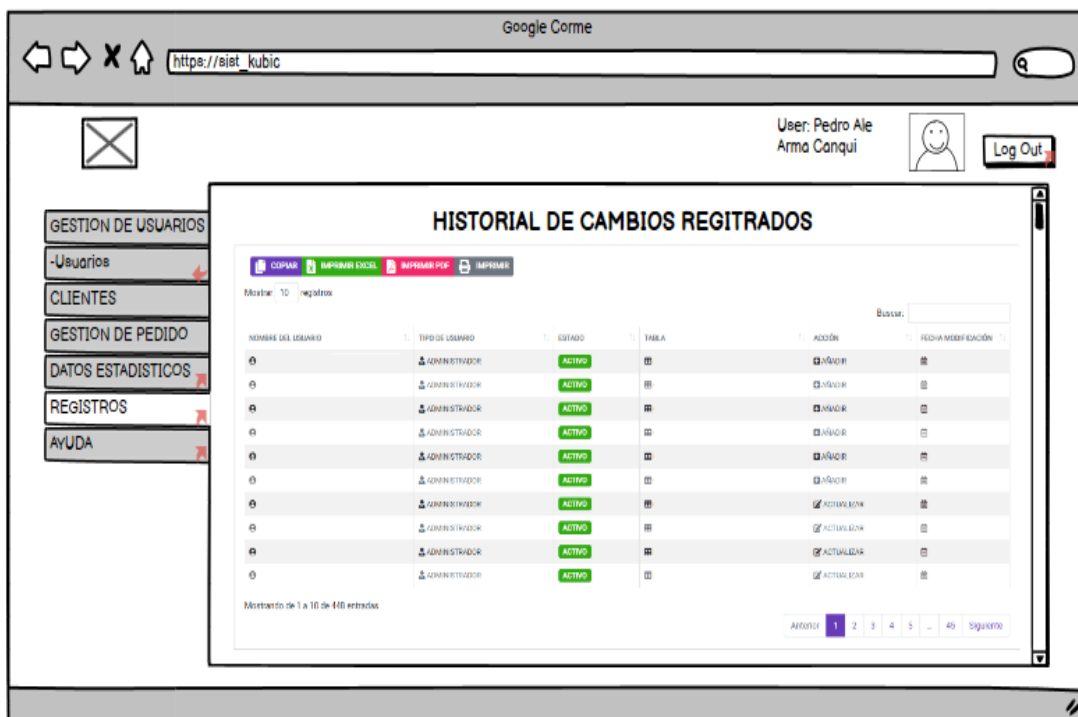
Modelo de Presentación - Administrador



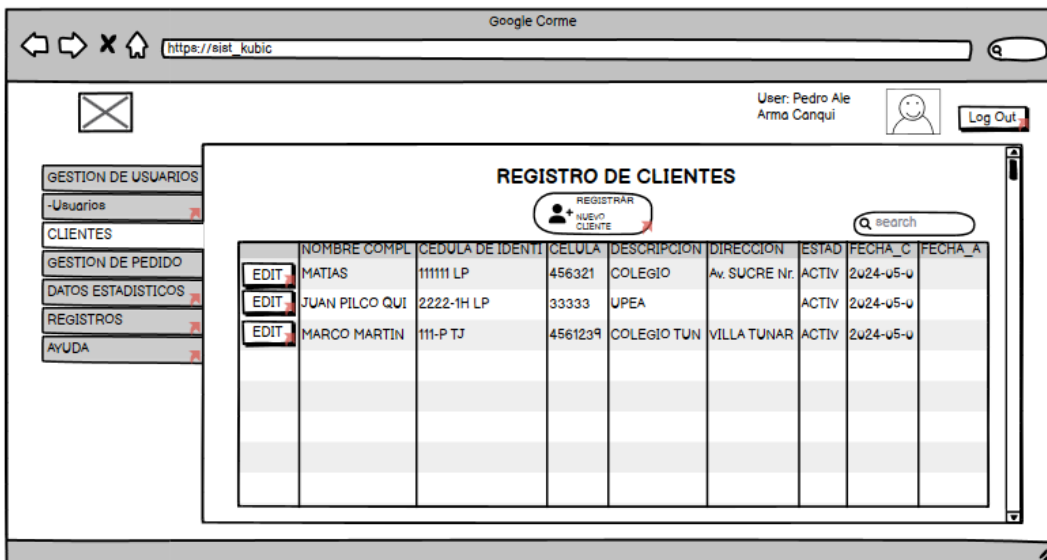
**Figura 26**  
*Modelo de Presentación - Usuarios*



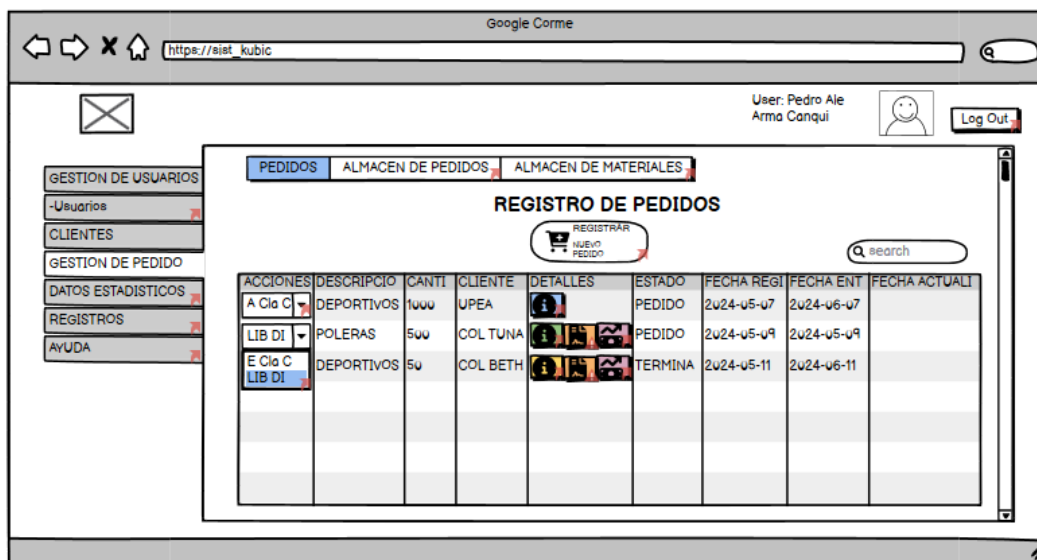
**Figura 27**  
*Modelo de Presentación - Roles y Permisos*



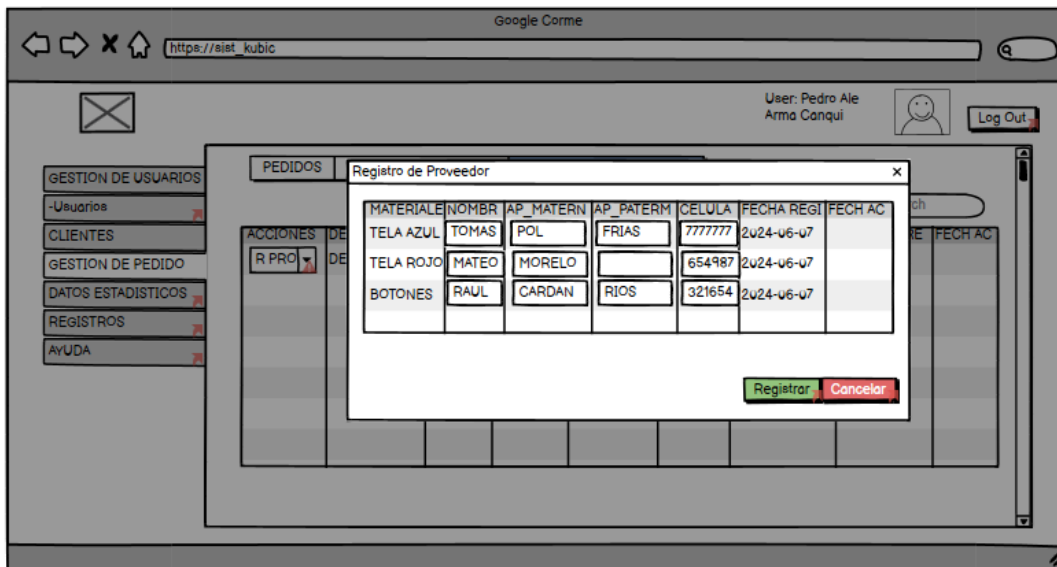
**Figura 28**  
Modelo de Presentación - Clientes



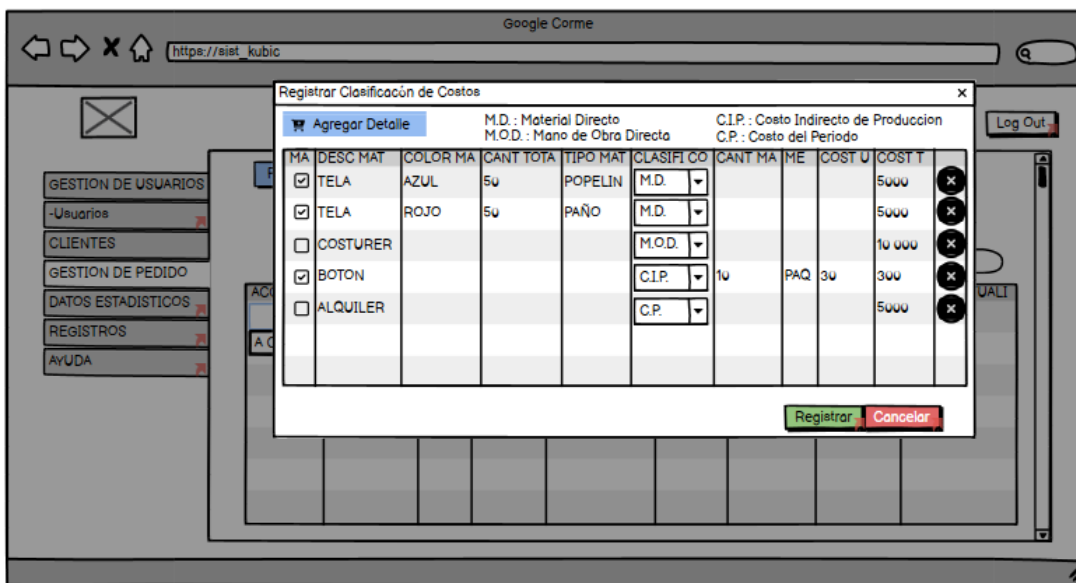
**Figura 29**  
Modelo de Presentación - Pedidos



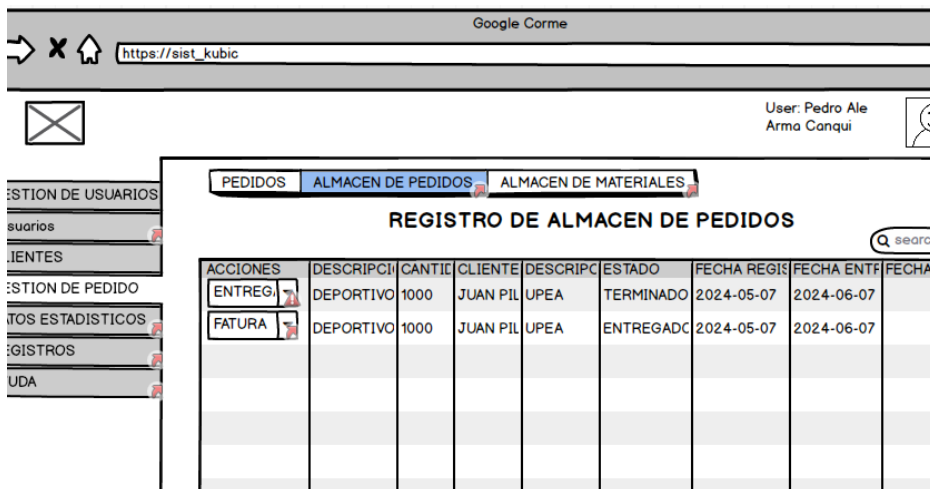
**Figura 30**  
*Modelo de Presentación - Registro del Proveedor*



**Figura 31**  
*Modelo de Presentación – Clasificación de Costos*



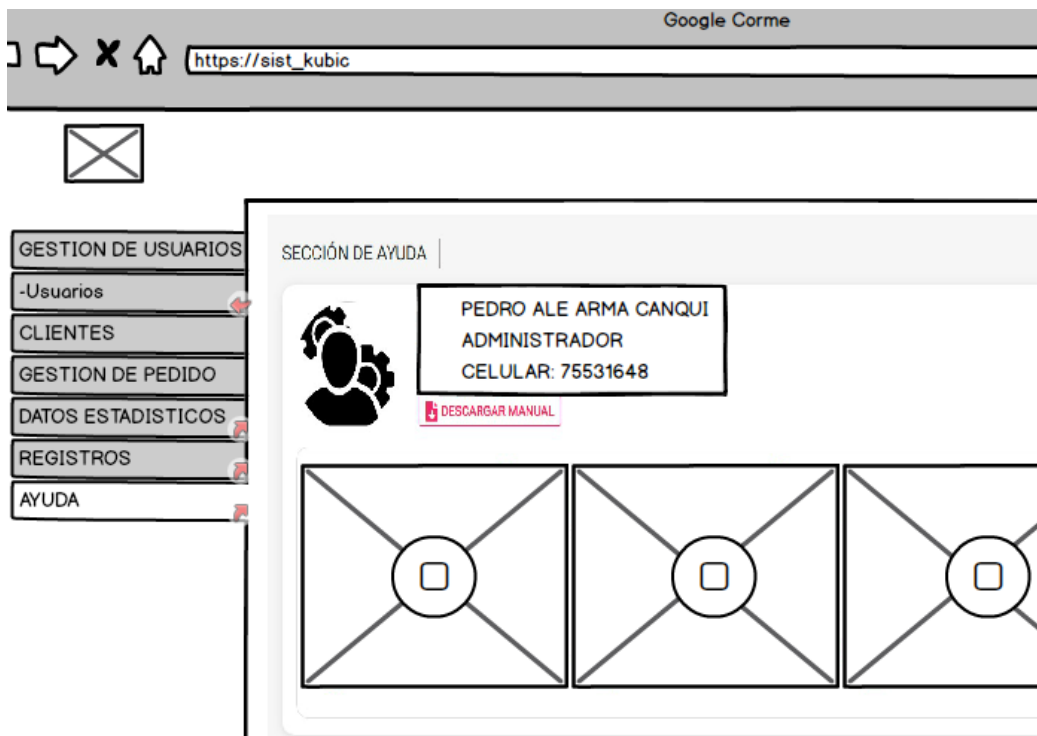
**Figura 32**  
Gestión de Producción



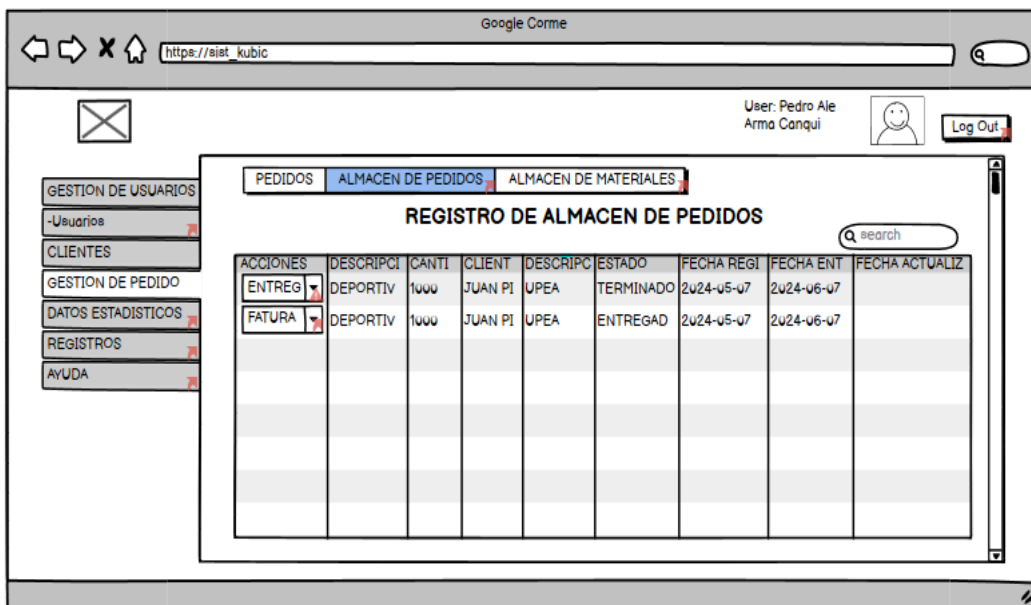
**Figura 33**  
Modelo de Presentación - Formulario de Determinación de Utilidades



**Figura 34**  
Gestión de Documentos



**Figura 35**  
Modelo de Presentación – Registro de Almacén de Pedidos



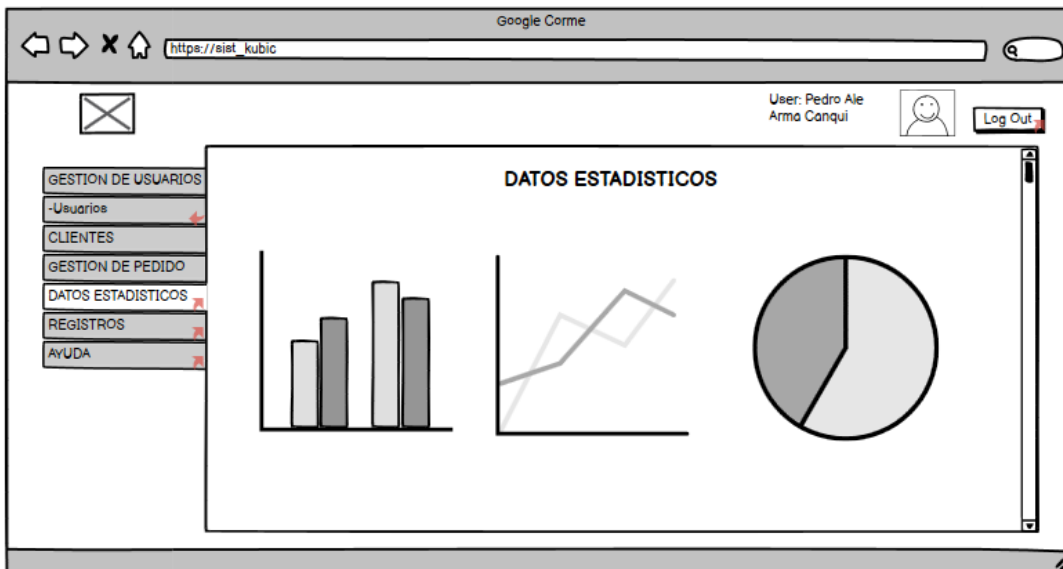
**Figura 36**

*Modelo de Presentación – Gestión Libro Diario*

FECHA	DETALLE	DEBE	HABER
-- X--			
DIA/MES/AÑO	Inv. Prod. Terminados	36000 bs	
	Pord. en proceso		36000 bs
	Mat. Dir		
	Man Ob Dir		
	Cost de Pord		
	total	36000 bs	36000 bs
-- X--			
DIA/MES/AÑO	Cliente por Cobrar	75908 bs	
	IT	2277 bs	
	Venta De Prod Ter		66040 bs
	Devito Fiscal - IVA		8585 bs
	IMP.T x P		2277 bs
	total	75908 bs	75908 bs

**Figura 37**

*Modelo de Presentación – Datos Estadísticos*



### 3.2.8 Modelo de Proceso

Figura 38

Página Principal

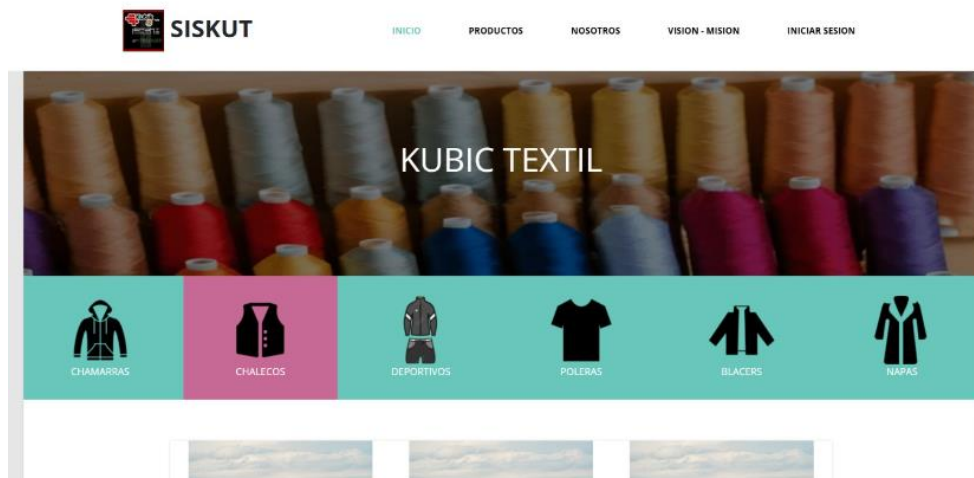


Figura 39

Catálogos Productos

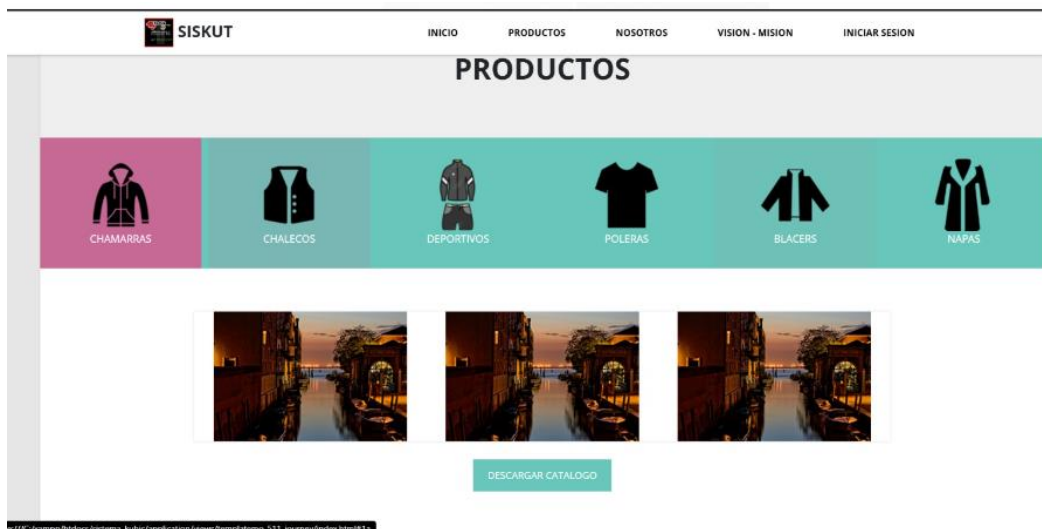
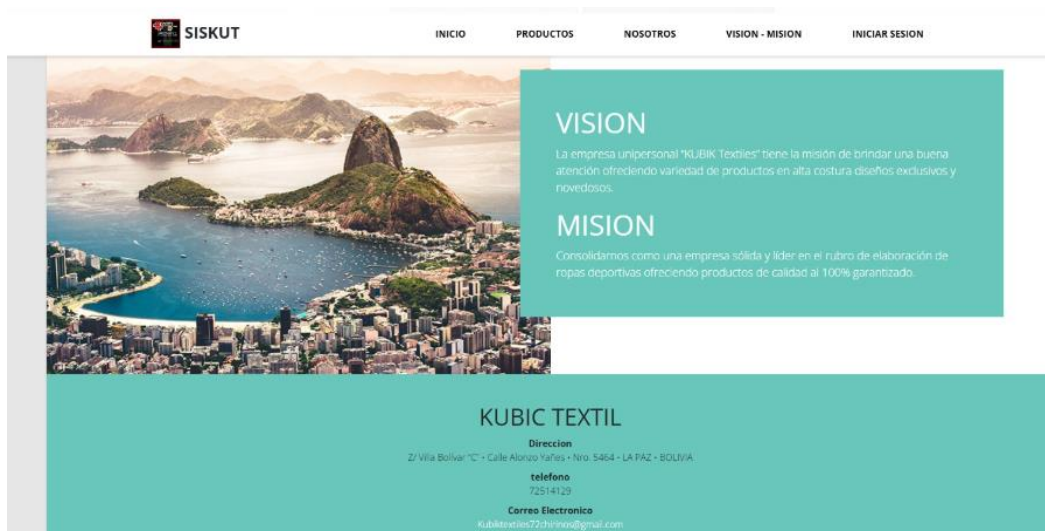


Figura 40

Datos de la Empresa



**SISKUT** INICIO PRODUCTOS NOSOTROS VISION - MISION INICIAR SESION

## VISION

La empresa unipersonal "KUBIK Textiles" tiene la misión de brindar una buena atención ofreciendo variedad de productos en alta costura diseños exclusivos y novedosos.

## MISION

Consolidarnos como una empresa sólida y líder en el rubro de elaboración de ropas deportivas ofreciendo productos de calidad al 100% garantizado.

### KUBIK TEXTIL

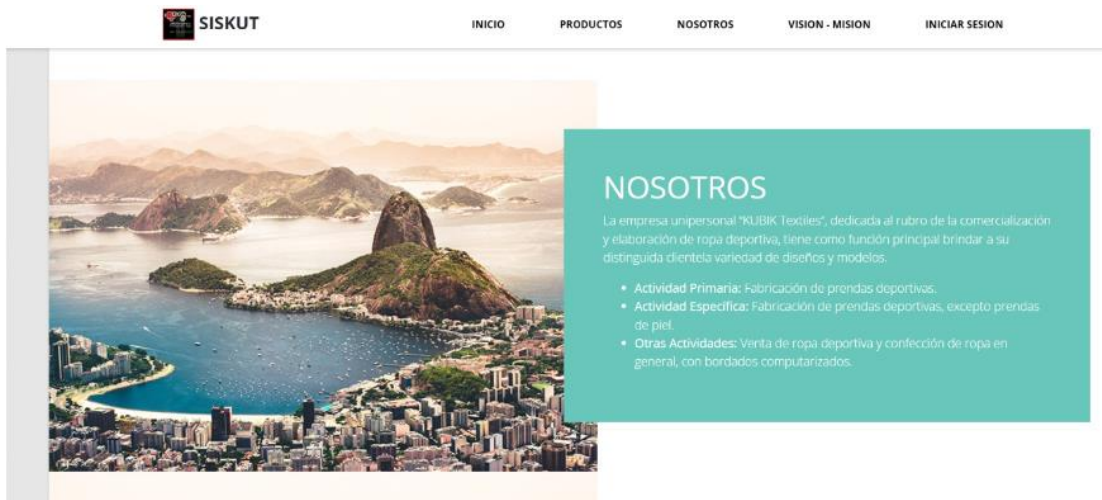
**Direccion**  
Z/ Vila Bolivar "C" - Calle Alonso Yañes - Nro. 5464 - LA PAZ - BOLIVIA.

**telefono**  
7251-4129

**Correo Electronico**  
kubiktextiles72311904@gmail.com

Figura 41

Empresa KUBIK TEXTILES



**SISKUT** INICIO PRODUCTOS NOSOTROS VISION - MISION INICIAR SESION

## NOSOTROS

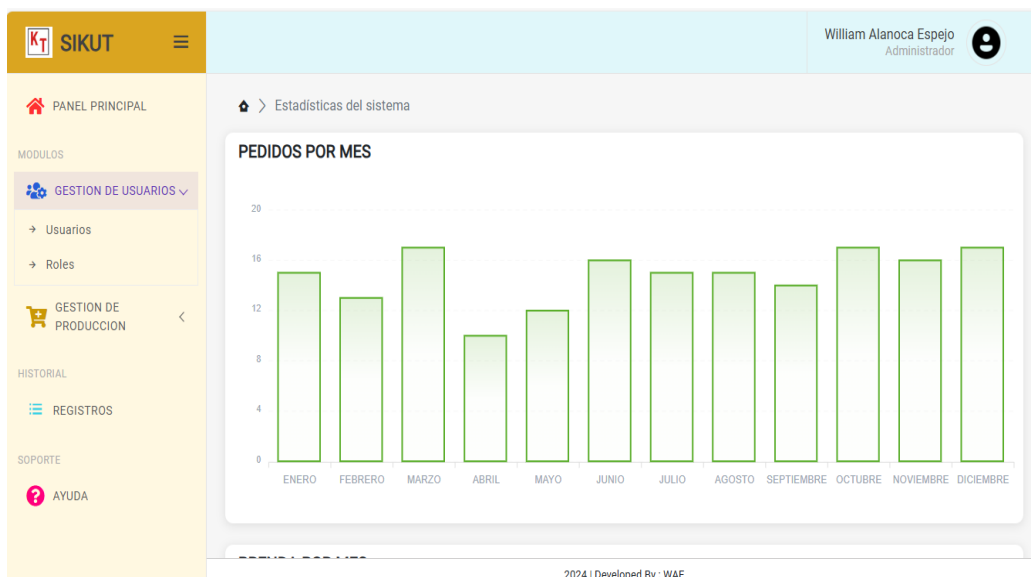
La empresa unipersonal "KUBIK Textiles", dedicada al rubro de la comercialización y elaboración de ropa deportiva, tiene como función principal brindar a su distinguida clientela variedad de diseños y modelos.

- **Actividad Primaria:** Fabricación de prendas deportivas.
- **Actividad Especifica:** Fabricación de prendas deportivas, excepto prendas de piel.
- **Otras Actividades:** Venta de ropa deportiva y confección de ropa en general, con bordados computerizados.

**Figura 42**  
*Inicio de Sesión del Sistema*



**Figura 43**  
*Ingreso al Sistema - Administrador*



**Figura 44**  
*Listado de Usuarios*

**REGISTRO DE USUARIOS**

Nuevo Usuario

COPIAR | DESCARGAR EXCEL | DESCARGAR PDF | IMPRIMIR

Mostrar 10 registros

Buscar:

	NOMBRE COMPLETO	CÉDULA IDENTIDAD	TIPO DE USUARIO	ESTADO	USUARIO	CELULAR
<a href="#">+</a> <a href="#">EDITAR</a>	WILLIAMA ALANOCASDASD ESPEJOASD	91994 LP	ADMINISTRADOR	ACTIVO	KADMIN2	68318098
<a href="#">+</a> <a href="#">EDITAR</a>	WILLIAM ALANOCA ESPEJO	9199444 LP	ADMINISTRADOR	ACTIVO	KADMIN	68318099

Mostrando de 1 a 2 de 2 entradas

Anterior 1 Siguiente

2024 | Developed By : WAE

**Figura 45**  
*Listado de Roles y Permisos*

**ROLES DE USUARIO**

COPIAR | DESCARGAR EXCEL | DESCARGAR PDF | IMPRIMIR

Mostrar 10 registros

Buscar:

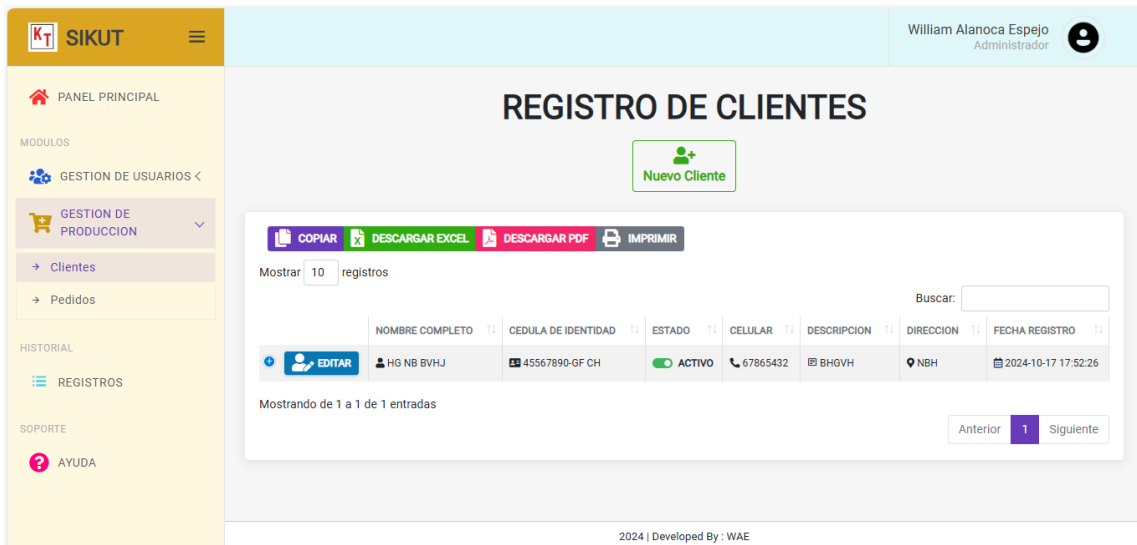
TIPO DE USUARIO	DESCRIPCION	ESTADO	FECHA REGISTRO	FECHA MODIFICACION
ADMINISTRADOR	ADMINISTRADOR	ACTIVO	2023-09-08 19:11:00	

Mostrando de 1 a 1 de 1 entradas

Anterior 1 Siguiente

2024 | Developed By : WAE

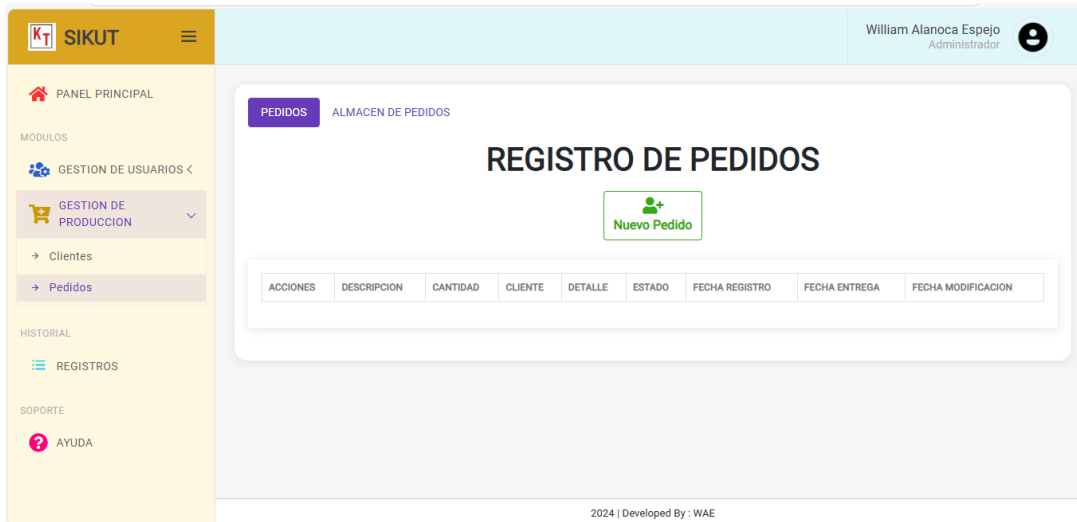
**Figura 46**  
Administración de Clientes



**Figura 47**  
Registro de Clientes



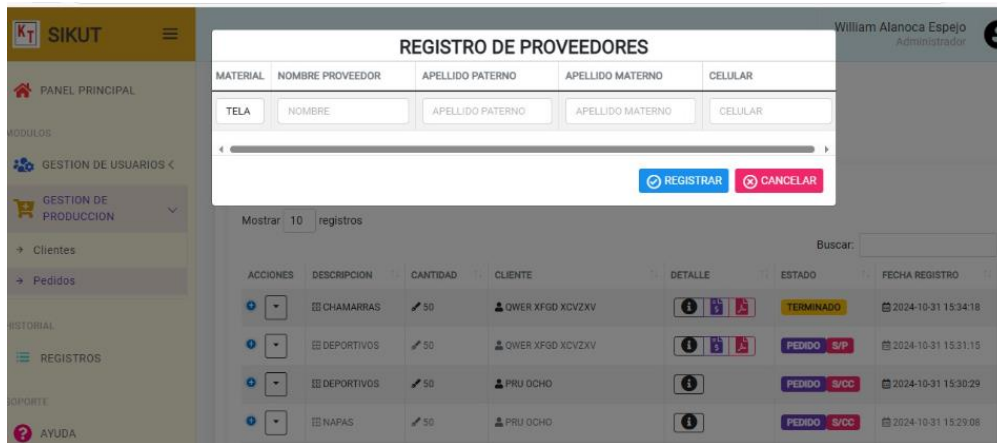
**Figura 48**  
*Administración de Pedidos*



**Figura 49**  
*Formulario Registro de Pedidos*



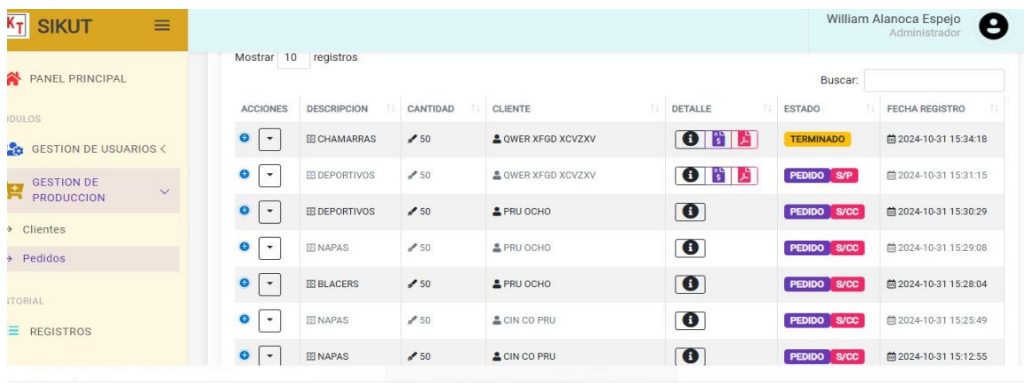
**Figura 50**  
Gestión de Proveedor



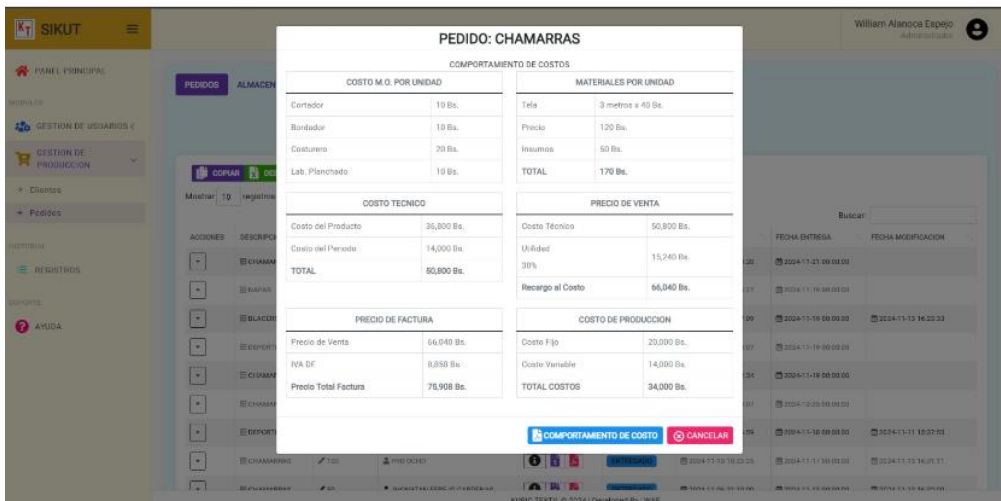
**Figura 51**  
Formulario de Clasificación de Costos



**Figura 52**  
Gestión de Producción



**Figura 53**  
Gestión de Utilidades



**Figura 54**  
Comportamiento de Costos

**UTILIDADES**  
**KUBIK TEXTILES**

Comportamiento de Costos y Proyección de utilidades bruta a nivel total y unitario.

Producción Q	Costo total de Prod.			Costo Unitario de Prod.			Precio Venta	Ingreso Venta	utilidad Bruta
	CFTP	CVTP	CTP	cfup	cvup	cup			

**CFTP:** Costo Fijo Total Producción.  
**CVTP:** Costo Variable Total Producción.  
**CTP:** Costo Total Producción.

**Figura 55***Gestión de Documentos***Contrato de Pedido****Empresa:** Kubik Textiles**Dirección:** Z/ Villa Bolívar "C" Calle Alonzo Yañes Nro. 5464**Representante:** Eustaquio Charles Chirinos Aduviri (El Fabricante)**Cliente:** Marco Martín (El Cliente)**Objeto**

Fabricación de 20 chamarras:

- 10 de color azul
- 10 de color rojo

**Precio Total**

Descripción	Cantidad	Precio Unitario	Total
Chamarra Azul	10	50 bs.	500 bs.
Chamarra Roja	10	50 bs.	500 bs.
<b>Total</b>			1,000 bs.

**Plazo de Entrega**

Dentro de 30 días hábiles desde la firma del contrato.

**Pago**

Anticipo: 50% (500 bs.)

Resto al recibir la mercancía.

**Figura 56***Gestión de Almacén de Pedidos*

PEDIDOS
ALMACEN DE PEDIDOS

## REGISTRO DE ALMACEN DE PEDIDOS

COPIAR
DESCARGAR EXCEL
DESCARGAR PDF
IMPRIMIR

Mostrar  registros
Buscar:

ACCIONES	DESCRIPCION	CANTIDAD	CLIENTE	DESCRIPCION	ESTADO	INFORMACION	FECHA REGISTRO
▼	CHAMARRAS	50	JHONATAN ESPEJO CARDENAS	COLEGIO	ENTREGADO	<span style="font-size: 1.2em;">i</span> <span style="font-size: 1.2em; margin-left: 5px;">P</span> <span style="font-size: 1.2em; margin-left: 5px;">S</span>	📅 2024-10-23 13:37:26
▼	CHAMARRAS	50	JHONATAN ESPEJO CARDENAS	COLEGIO	ENTREGADO	<span style="font-size: 1.2em;">i</span> <span style="font-size: 1.2em; margin-left: 5px;">P</span> <span style="font-size: 1.2em; margin-left: 5px;">S</span>	📅 2024-10-23 12:41:19

**Figura 57**

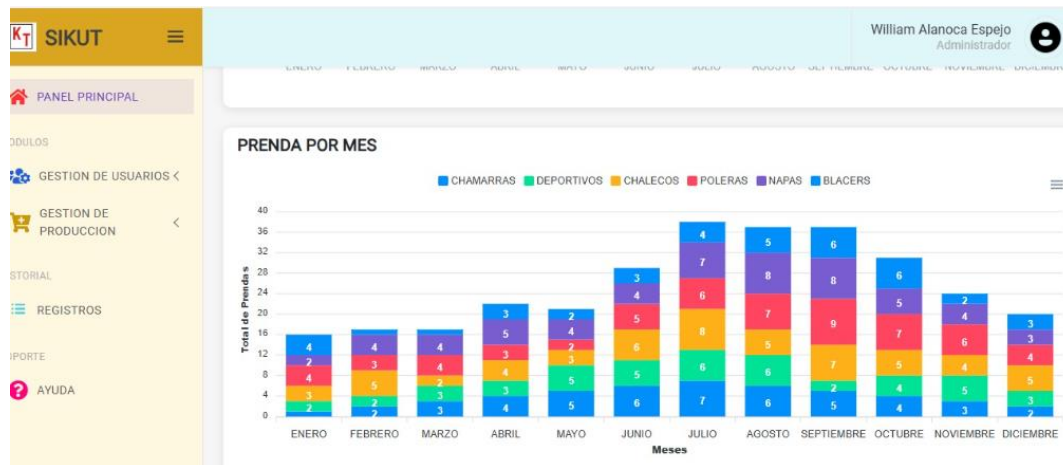
Modelo de Presentación – Libro Diario

**LIBRO DIARIO**

FECHA	DETALLE	DEBE	HABER
09/11/2024	Inventario de Productos Terminados	234.00 Bs.	-
09/11/2024	Producto en proceso M.D.	-	234.00 Bs.
<b>TOTAL</b>		<b>234.00 Bs.</b>	<b>234.00 Bs.</b>

**Figura 58**

Gestión de Datos Estadísticos



# **CAPÍTULO IV**

## **CALIDAD, SEGURIDAD Y COSTOS**

**INGENIERÍA  
DE SISTEMAS**  
UNIVERSIDAD PUBLICA DE EL ALTO



## **4 CAPÍTULO IV CALIDAD, SEGURIDAD Y COSTOS**

### **4.1 INTRODUCCIÓN**

En este capítulo se presenta detalladamente la calidad, el costo y la seguridad del presente proyecto a través de diversas métricas, estas métricas son cruciales para asegurar que el sistema cumpla con los requisitos técnicos y las expectativas del usuario.

Las métricas específicas que se utilizan para evaluar cada uno de estos aspectos, proporcionando una base sólida para medir y controlar el desempeño del sistema que ayudará a determinar que el software opera de manera efectiva y cumple con los estándares establecidos, las métricas de costo permitirán analizar la eficiencia en la gestión de recursos y el presupuesto, mientras que las métricas de seguridad evaluarán la capacidad del sistema para proteger en contra de las amenazas y vulnerabilidades.

Al detallar estas métricas, el capítulo ofrecerá una comprensión integral de cómo evaluar y asegurar la excelencia del sistema en todos estos ámbitos clave.

### **4.2 MÉTRICAS DE CALIDAD - ESTÁNDAR ISO/IEC 25000**

Para el presente Proyecto de Grado, se han implementado métricas de calidad que permiten evaluar diversos parámetros del sistema. Estas métricas son fundamentales para especificar de manera ordenada las características y atributos del software. Entre los diversos modelos y criterios de calidad para productos de software, uno de los más destacados es el ISO 25000. Este estándar ofrece un marco integral para evaluar la calidad del software y se utiliza para medir varios aspectos, incluyendo la funcionalidad, fiabilidad, usabilidad, eficiencia, mantenibilidad y portabilidad del sistema, de acuerdo al siguiente detalle:

#### **4.2.1 Funcionalidad**

Se llevará a cabo una exhaustiva evaluación para determinar si el software cumple con los requisitos funcionales establecidos. Este proceso garantiza que el programa posea las

características esenciales para satisfacer las demandas y expectativas del usuario, asegurando su plena funcionalidad.

#### **4.2.1.1 Punto Función**

El proceso para obtener la estimación de costos del software se apoya en cinco atributos clave, los cuales son fundamentales para asegurar tanto la precisión como la eficacia al momento de sacar los costos. Por lo tanto, uno de estos atributos es la cantidad de entradas de usuario, estas entradas representan la información de acuerdo al software desarrollado.

La importancia de esta métrica radica en su capacidad para influir en la complejidad y el esfuerzo requerido tanto para el desarrollo inicial como para el mantenimiento continuo del sistema, cada entrada de usuario tiene el potencial de impactar la arquitectura del software, lo que a su vez repercute directamente en su diseño y en su capacidad de mantenimiento a largo, a continuación, se mostrará la cantidad de entradas del usuario.

**Tabla 19**

*Número de Usuarios del Sistema*

<b>ENTRADAS DE USUARIO</b>	
Administración de usuarios	1
Administración de técnicos	1
Administración de usuarios	15
<b>TOTAL</b>	<b>17</b>

Salidas de usuarios. estos datos obtenidos, es la cantidad de resultados que el sistema entrega, incluye diversos tipos de outputs, como mensajes, notificaciones, reportes y alertas. Estas salidas son muy importantes para la interacción del usuario con el sistema y su capacidad de responder a las necesidades informativas. A continuación, se muestra la siguiente tabla de salidas del usuario.

**Tabla 20***Número de Salidas de los Usuarios*

<b>SALIDAS DE USUARIO</b>	
Administración de usuarios	3
Administración de técnicos	3
Administración de usuarios	15
<b>TOTAL</b>	<b>21</b>

Peticiones de usuario, es la cantidad de veces que una entrada del usuario desencadena una respuesta interactiva del software, lo que refleja la capacidad del sistema para reaccionar y procesar información en tiempo real.

**Tabla 21***Número de Peticiones*

<b>PETICIONES DE USUARIO</b>	
Administración de usuarios	3
Administración de técnicos	3
Administración de usuarios	20
<b>TOTAL</b>	<b>14</b>

Cantidad de archivos, este aspecto se refiere al total de archivos que maneja el sistema. Puede incluir tanto grupos lógicos de datos, que agrupan información relacionada de manera coherente, como archivos individuales que almacenan datos específicos.

Cantidad de interfaces externas, este parámetro indica el total de interfaces que el sistema puede leer o interpretar, es decir, todas las conexiones externas con otros sistemas, aplicaciones o dispositivos que intercambian información con él.

Las interfaces externas pueden incluir APIs, servicios web, archivos de intercambio de datos y cualquier otra forma de comunicación legible por máquina. La cantidad de interfaces

externas es importante porque influye en la integración y la interoperabilidad del sistema, así como en la complejidad de su desarrollo y mantenimiento.

**Tabla 22**

*Parámetros de Medición*

	Parámetro de Medición	Cuenta	Factor medio	Total
1	Numero de entrada del usuario	17	3	51
2	Número de Salidas del Usuario	21	4	84
3	Número de Petición del Usuario	14	5	70
4	Número de archivos	8	5	40
5	Número de interfaces externas	1	5	5
CT	Cuenta Total			250

Después de contar los parámetros de función, se procede al cálculo de los puntos de función sin ajustar en la tabla siguiente, este proceso implica sumar las diversas métricas recolectadas, proporcionando una medida cuantitativa que refleja la funcionalidad total del sistema antes de considerar cualquier ajuste de complejidad.

En la tabla se muestra la suma total de los puntos de función sin ajustar, para obtener los valores ajustados se aplican factores de complejidad específicos que se detallan a continuación en la tabla 20, estos factores consideran diversas características del sistema.

Ajustar los puntos de función iniciales con estos factores permite una evaluación más precisa y realista de la carga de trabajo y los recursos necesarios para el desarrollo del software, este método asegura que todas las particularidades y desafíos del proyecto se reflejen adecuadamente en la estimación final.

Tabla 23

Indicé de Usabilidad

IMPORTANCIA	0%	20%	40%	60%	80%	100%
ESCALA	NO INFLUENCIA	INCIDENTAL	MODERADO	MEDIO	SIGNIFICATIVO	ESENCIAL
FACTOR	0	1	2	3	4	5
¿Requiere el sistema copias de seguridad y de recuperación fiable?						5
¿Se requiere 0 comunicación de datos?					X	4
¿Existe funciones de procesos distribuidos?						5
¿Es crítico el rendimiento?					X	4
¿El sistema web será ejecutado el SO Actual?						5
¿Se requiere una entrada interactiva para el sistema?					X	4
¿Se requiere que el sistema tenga entradas a datos con múltiples ventanas?					X	4
¿Se actualiza los archivos de forma interactiva?					X	4
¿Son complejas las entradas, salidas, los archivos o las peticiones?						5
¿Es complejo el procesamiento interno del sistema?					X	4
¿Se ha diseñado el código para ser reutilizado?					X	4
¿Se ha diseñado el sistema para facilitar al usuario el trabajo y ayudarlos a encontrar la información?					X	4
¿Se ha diseñado la aplicación para ser fácilmente utilizada por el usuario?						5
TOTAL						61

Fórmula para ajustar puntos de función, que considera factores como la complejidad técnica, el rendimiento y la usabilidad.

**( 18 ) Ecuación de Ajuste**

$$PF = \text{conteo total} * [0.65 + 0.01 * \sum fi] \quad (18)$$

Donde:

$\sum (fi)$  : Sumatoria de los valores de los factores de ajuste.

*Conteo total* : Número de parámetros del punto función

*PF* : Ajuste

Se obtiene la sumatoria de ponderación de los índices de usabilidad.

$$\sum(fi) = 61$$

$$PF = 250 * [0.65 + 0.01 * 61]$$

$$PF = 315$$

Con el límite superior establecido en la suma de los factores de ajuste máximo  $\sum(fi \text{ max}) = 61$ , se realiza el cálculo del valor siguiente en la secuencia, este proceso es fundamental para mantener la coherencia en la evaluación de la complejidad del sistema, hallamos el punto función máximo.

$$PFmax = \text{Cuenta Total} [0.65 + (0.01 * \sum fi \text{ max})]$$

$$PFmax = 250 * [0.65 + (0.01 * 70)]$$

$$PF \text{ max} = 337.5$$

La medición de la funcionalidad se basa en comparar el valor máximo posible con el valor obtenido del punto de función ajustado, esta relación proporciona información crucial sobre la eficiencia y la complejidad del sistema evaluado.

$$Funcionalidad = \frac{PF}{PFmax} * 100$$

$$Funcionalidad = \frac{315}{337.5} * 100$$

$$Funcionalidad = 0.93 * 100$$

<b>Funcionalidad = 93%</b>
----------------------------

Entonces, la eficacia del sistema alcanza el 93 %, indicando su alta capacidad operativa con un riesgo mínimo de fallos (7 %),

#### 4.2.2 Confiabilidad

La confiabilidad de un sistema se refiere a su capacidad para mantener un nivel de funcionamiento adecuado durante un período de prueba establecido que implica considerar aspectos como la madurez del sistema. Aplicaremos las fórmulas correspondientes, teniendo en cuenta las probabilidades de fallas y éxitos debido a varios factores.

Considerando que se tiene en cuenta que:

Probabilidad de fallas

$$F(t) = P(T \leq t)$$

Probabilidad de trabajo sin fallas

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

Se considera la siguiente función al calcular la confiabilidad del sistema.

#### (19) Función de la Confiabilidad del Sistema

$$R(t) = f * e^{-\mu * t} \tag{19}$$

Donde:

$f$	=	Funcionalidad del sistema
$\mu$	=	Probabilidad de error del sistema
$t$	=	Tiempo de prueba del sistema
$(t)$	=	Confiabilidad

Durante un periodo de prueba de 20 días, se registra una falla por cada 12 ejecuciones del sistema.

$$F(t) = 0.93 * e^{-\frac{1}{10} * 18}$$

$$F(t) = 0.93 * 0.135$$

$$F(t) = 0.125$$

$$F(t) = 0.110 * 100$$

$$\mathbf{F(t) = 11 \%}$$

Este valor representa la probabilidad de fallas del sistema en un periodo de tiempo, por lo que la siguiente formula representa la probabilidad de que el sistema sea confiable.

Sustituyendo:

$$\text{Confiabilidad} = P(T \leq t) = 1 - F(t)$$

$$\mathbf{\text{Confiabilidad} = P(T \leq t) = 0.110}$$

$$\text{Confiabilidad} = P(T \leq t) = 1 - 0.110$$

$$\text{Confiabilidad} = P(T \leq t) = 0.875 * 100$$

$$\mathbf{\text{Confiabilidad} = P(T \leq t) = 89 \%}$$

Por lo tanto, para este software desarrollado, la confiabilidad se sitúa en un sólido 87.5%, verificada a lo largo de un periodo de prueba de 20 días, este resultado confirma la consistencia y fiabilidad del sistema en condiciones operativas durante este período de evaluación establecido, cabe mencionar que en todo proceso se tiene un margen de error, que en lo recomendable sea mínimo.

### 4.2.3 Usabilidad

Al evaluar la usabilidad de un sistema, es importante considerar el factor humano, asegurando que el software cumpla con los requisitos y expectativas del usuario, por lo tanto, se lleva a cabo una evaluación mediante encuestas dirigidas a los usuarios del sistema.

Estas encuestas permiten recopilar información valiosa sobre la experiencia del usuario, su satisfacción y la facilidad de uso percibida que se utiliza luego para calcular la usabilidad del sistema, aplicando una fórmula diseñada específicamente para medir la eficacia y la experiencia del usuario en la interacción con el software.

#### **( 20 ) Función para Determinar la Usabilidad**

$$FU = \left( \frac{\sum \text{valor}}{n} * 100 \right) / 5 \quad (20)$$

Donde:

$\sum$  valor = Sumatoria de los valores de la usabilidad del sistema

$n$  = Cantidad de valores de la usabilidad del sistema

$FU$  = Usabilidad

**Tabla 24**

*Indicadores de Facilidades de Uso*

ESCALA	VALOR
--------	-------

Muy bueno	5
Bueno	4
Regular	3
Malo	2
Pésimo	1

A continuación, en la siguiente tabla se muestra:

**Tabla 25**

*Usabilidad del Sistema*

PREGUNTA	VALOR
¿Puede Utilizar con facilidad el sistema?	5
¿Puede Controlar operaciones que el sistema solicita?	5
¿Las Respuestas del sistema son complicadas?	4
¿El Sistema permitió la retroalimentación de información?	5
¿El sistema cuenta con interface agradable a la vista?	4
¿La respuesta del sistema es satisfactoria?	4
¿Le parece complicada las funciones del sistema?	5
¿Se hace difícil o dificultoso aprender a manejar el sistema?	5
¿Los resultados que proporciona el sistema facilitan el trabajo?	4
¿Durante el uso del sistema se produjo errores?	4
<b>TOTAL</b>	<b>45</b>

A partir de los datos recabados a través del cuestionario, llevamos a cabo un análisis exhaustivo para evaluar la usabilidad del sistema. Este proceso implica calcular diversos

indicadores y métricas que reflejen la facilidad con la que los usuarios interactúan con el sistema.

$$FU = \left( \frac{45}{10} * 100 \right) / 5$$

$$FU = (0.46 * 100) / 5$$

<b>Usabilidad = FU = 94%</b>
------------------------------

Con esta información, podemos afirmar que la facilidad de uso del sistema alcanza un porcentaje del 94%, este dato proporciona una medida concreta de la experiencia del usuario y su capacidad para interactuar efectivamente con el sistema.

#### 4.2.4 Eficiencia

La eficacia evalúa si el sistema utiliza eficientemente sus recursos. Para determinar su nivel, se considera una escala específica como referencia.

**Tabla 26**

*Escala de Valores de Eficiencia*

ESCALA	VALOR
Excelente	5
Bueno	4
Aceptable	3
Deficiente	2
Pésimo	1

**Tabla 27***Evaluación de Eficiencia*

PREGUNTA	PORCENTAJE
¿La distribución y estilo de la interfaz permite que un usuario introduzca con eficiencia las operaciones y la información?	4
¿Una secuencia de operaciones (o entrada de datos) puede realizarse con facilidad de movimientos?	5
¿Los datos de salida están presentados de modo que se entienden de inmediato?	5
¿Las operaciones jerárquicas están organizadas de manera que minimizan la navegación del usuario para hacer que alguna se ejecute?	5
¿Procesa y responde adecuadamente cuando realiza alguna consulta o búsqueda?	5
<b>TOTAL</b>	<b>24</b>

Para determinar la eficacia, se emplea la fórmula siguiente.

**( 21 ) Fórmula Eficiencia**

$$E = \frac{\sum x_i}{n} * \frac{100}{n} \quad ( 21 )$$

Donde:

$\sum x_i$  = Sumatoria de los valores de eficiencia

$n$  = Número de preguntas

$E$  = Eficiencia

Sustituyendo se tiene lo siguiente:

$$E = \frac{24}{5} * \frac{100}{5}$$

<b><i>Eficiencia = E = 96 %</i></b>
-------------------------------------

Con estos datos, determinamos que la eficiencia del sistema se sitúa en el 96%, lo que indica que los recursos del sistema se están utilizando de manera efectiva y óptima.

#### 4.2.5 Mantenibilidad

Esta métrica evalúa la cantidad de trabajo requerido para realizar cambios en el sistema, ya sea para corregir errores o agregar nuevas funcionalidades. El estándar IEEE94 recomienda utilizar el índice de madurez del sistema como indicador de su estabilidad. Por lo tanto, la ecuación para este índice es la siguiente:

**( 22 ) Ecuación para el Índice de Madurez del Sistema**

$$IMS = \frac{[Mt - (Fa + Fc + Fd)]}{Mt} \quad ( 22 )$$

Donde:

<i>Mt</i>	=	Número de módulos total de la versión actual
<i>Fa</i>	=	Número de módulos de la versión actual que se añadieron.
<i>Fc</i>	=	Número de módulos de la versión actual que se cambiaron.
<i>Fd</i>	=	Número de módulos que se eliminaron en la versión anterior a la actual.
IMS	=	Madurez del sistema

**Tabla 28**

*Indicadores de Mantenibilidad*

DESCRIPCIÓN	VALOR
Mt	6
Fa	0
Fc	2
Fd	0

Sustituyendo en la ecuación se tiene:

$$IMS = \frac{[8 - (0 + 1 + 0)]}{8}$$

$$IMS = 0.875 * 100\%$$

<b>IMS = 87.5%</b>
--------------------

En conclusión, el índice de mantenibilidad del sistema desarrollado alcanza el 87.5% indicando que el sistema tiene una alta capacidad para ser modificado, corregido y mejorado, facilitando su adaptación a nuevas necesidades y requisitos a lo largo del tiempo.

#### 4.2.6 Portabilidad

La capacidad de un software para ser transferido de un entorno a otro incluye varios aspectos importantes:

Adaptabilidad

Habilidad del software para ajustarse a diferentes entornos sin requerir modificaciones adicionales.

Facilidad de Instalación

Esfuerzo requerido para instalar el software en un entorno específico.

Conformidad

Verificación de que el software cumple con estándares o convenciones de portabilidad.

### Capacidad de sustitución

Facilidad y esfuerzo necesarios para sustituir el software con otro producto que tenga funciones similares.

El sistema, desarrollado con Laravel, puede implementarse en cualquier servidor con Apache y las herramientas necesarias.

Como la tecnología web es ejecuta fácilmente en cualquier dispositivo con conexión a Internet y un navegador.

#### **( 23 ) Fórmula Portabilidad**

$$P = 1 - \left( \frac{NAS}{NIS} \right) \quad (23)$$

Sustituyendo la fórmula

$$Portabilidad = 1 - \left( \frac{1}{12} \right)$$

$$Portabilidad = 0.917 * 100\%$$

$Portabilidad = 91,7\%$
-------------------------

Después de un análisis exhaustivo de cada parámetro según los criterios establecidos por la norma ISO 25000 se ha obtenido resultados que reflejan el rendimiento global del sistema, el cálculo detallado de cada aspecto relevante para asegurar que el sistema cumple con los estándares de calidad y funcionalidad esperados, proporcionando así una visión clara y detallada de su desempeño general.

#### **Tabla 29**

*Resultados Obtenidos - Norma ISO 25000*

<b>CRITERIOS EVALUADOS</b>	<b>RESULTADOS</b>
Funcionalidad	93 %
Confiabilidad	89 %
Usabilidad	94 %
Eficiencia	92%
Mantenibilidad	87,5 %
Portabilidad	91,7 %
<b>TOTAL</b>	<b>91.16%</b>

**4.3 SEGURIDAD DEL SISTEMA – ISO 27001**

Uno de los elementos fundamentales a considerar en este proyecto es la aplicación de medidas de seguridad. La norma ISO 27001 se encarga de evaluar y corregir el cumplimiento de dichas medidas, además de promover la mejora constante mediante un conjunto de controles destinados a mitigar el riesgo de incidentes de seguridad en el funcionamiento del sistema dentro de la institución. Para lograrlo, se consideran diversos tipos de seguridad, entre los cuales se incluyen.

**4.3.1 Identificación y Autenticación.**

Los usuarios cuentan con usuario y contraseña estrictamente controlada para acceder al sistema.

**Figura 59**  
Identificación y Autenticación

The screenshot shows the SIKUT web application interface. The main content area is titled 'REGISTRO DE USUARIOS' and features a 'Nuevo Usuario' button. Below this, there are action buttons for 'COPIAR', 'DESCARGAR EXCEL', 'DESCARGAR PDF', and 'IMPRIMIR'. A search bar is present with the text 'Mostrar 10 registros'. The table below lists four users:

	NOMBRE COMPLETO	CÉDULA IDENTIDAD	TIPO DE USUARIO	ESTADO	USUARIO	CELULAR
<a href="#">EDITAR</a>	PEDRO PINEDO PEREZ	6753421	TECNICO	ACTIVO	KTECNICO	98576243
<a href="#">EDITAR</a>	WILLIAMA ALANOCASDASD ESPEJOASD	91994	ADMINISTRADOR	INACTIVO	KDMIN2	68318098
<a href="#">EDITAR</a>	WILLIAMA ALANOCASDASD ESPEJOASD	91994 LP	ADMINISTRADOR	INACTIVO	KDMIN2	68318098
<a href="#">EDITAR</a>	WILLIAM ALANOCA ESPEJO	9199444 LP	ADMINISTRADOR	ACTIVO	KADMIN	68318099

### 4.3.2 Encriptación

Para la seguridad del password estará encriptado con el algoritmo que es una encriptación de alta seguridad para el ingreso al sistema.

**Figura 60**  
Seguridad de Password

The screenshot shows the phpMyAdmin interface displaying a SQL query result for the 'rol\_usuario' table. The query is 'SELECT \* FROM `rol\_usuario`'. The results show four rows of user data:

	id	usuario	nombre us	contraseña us	tipo_usuario	estado	fecha_crear
<a href="#">Editar</a> <a href="#">Copiar</a> <a href="#">Borrar</a>	1	1	KADMIN	\$2y\$10\$3gltz7v0iAuz8GxAGGiv0Z1gVhnp5RpXon6coM...	1	1	2023-09-09 1
<a href="#">Editar</a> <a href="#">Copiar</a> <a href="#">Borrar</a>	2	2	KDMIN2	\$2y\$10\$FBs3kz2ZMlQm5UJzm0e4OuyjAouuA27gpxky5aR5KN...	1	2	2024-09-13 1
<a href="#">Editar</a> <a href="#">Copiar</a> <a href="#">Borrar</a>	3	3	KDMIN2	\$2y\$10\$2Kp93LPeORMy7JA75Q409S1XnDYF0xzCZ9KWFlptz...	1	2	2024-10-09 1
<a href="#">Editar</a> <a href="#">Copiar</a> <a href="#">Borrar</a>	4	4	KTECNICO	\$2y\$10\$RrLUqTgCuzKj1ODGkz6.zzrNvH0eBQCY14teut9z...	2	1	2024-11-04 0

### **4.3.3 Seguridad Física**

Los Back-up de la base de datos deberán ser protegidas en áreas seguras, que solo permita el acceso a personal autorizado, deben ser almacenados por fechas.

### **4.3.4 Seguridad Organizativa**

Manejar el Back-up de acuerdo a la fecha que se realizó para mayor organización.

## **4.4 PRUEBAS DE SOFTWARE**

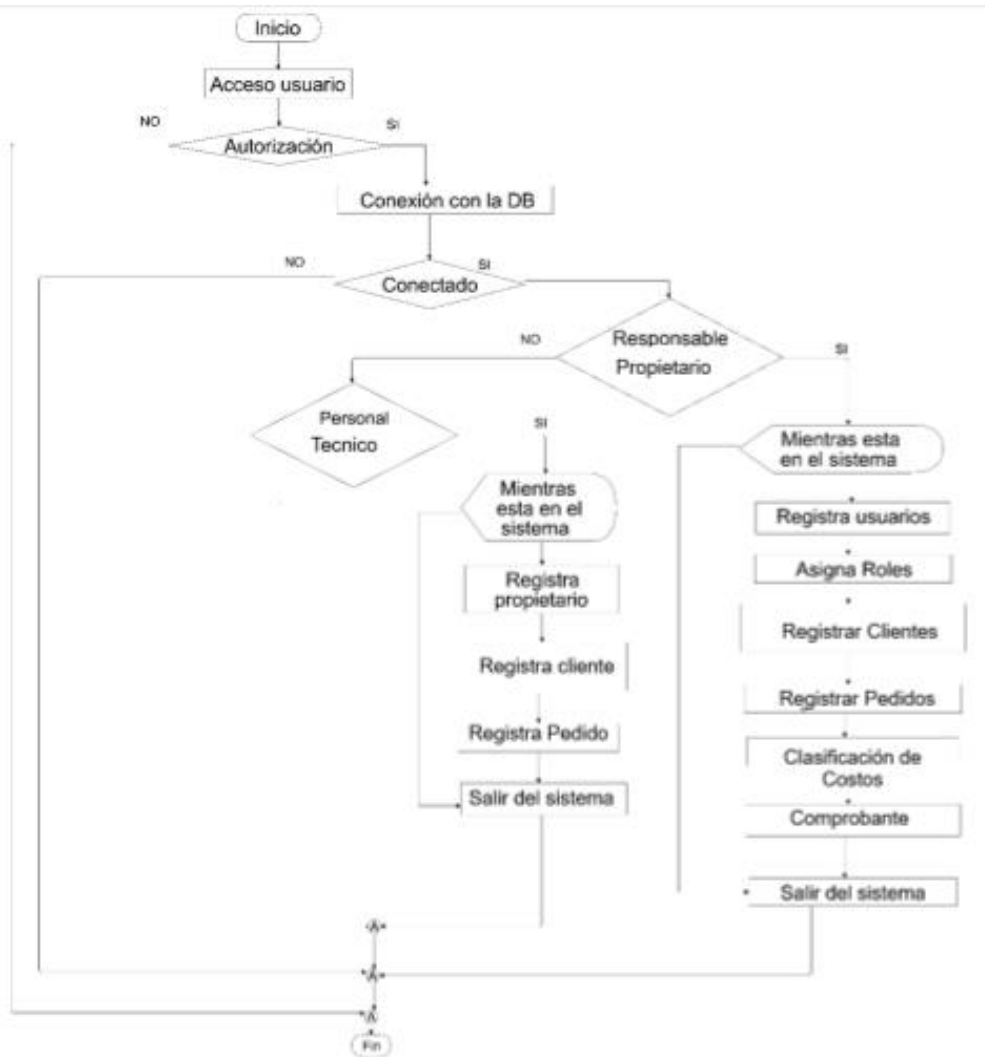
Se llevarán a cabo pruebas detalladas para que el software cumpla con los estándares de calidad y rendimiento requeridos, estas pruebas basadas en casos de uso del sistema, evaluarán la funcionalidad, el rendimiento y la usabilidad del software, además se realizarán pruebas de carga y estrés para verificar la estabilidad y capacidad del sistema bajo condiciones extremas.

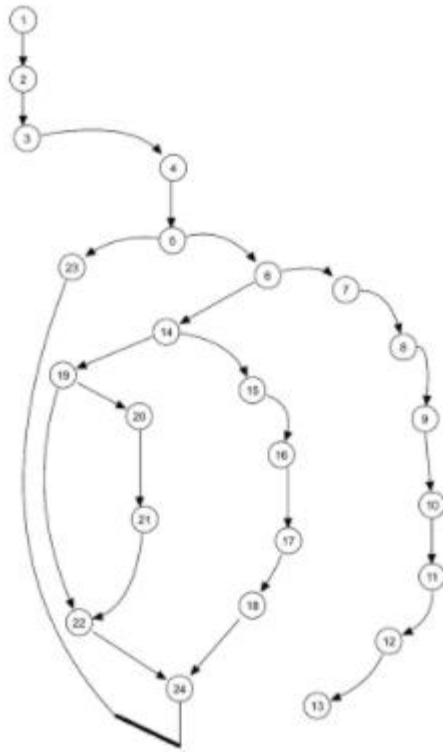
El objetivo es asegurar que la entrada de datos, la navegación por el sistema, los procedimientos y la generación de resultados se ejecuten de manera correcta y eficiente, cumpliendo con los requisitos establecidos.

### **4.4.1 Prueba de Caja Blanca**

Se llevarán a cabo pruebas de caja blanca para evaluar las condiciones lógicas del portal web y revelar su estructura de datos, este enfoque ayudará a reducir errores y a comprender la complejidad lógica del sistema determinando el número de caminos posibles en el código excluyendo los detalles de entradas y salidas, y proporcionando una visión detallada del flujo lógico del software.

**Figura 61**  
*Diagrama del Sistema*



**Figura 62***Grafo de Flujo del Sistema*

Después de revisar el diagrama de del sistema y haber generado el grafo correspondiente, el siguiente paso es determinar la complejidad ciclomática, se aplicará la fórmula adecuada el cual permite cuantificar la complejidad del código al medir el número de caminos independientes a través del grafo.

Donde:

1 :	Inicio del sistema	15 :	Gestión proveedores
2 :	Usuario y contraseña	16 :	Gestión de pedidos
3 :	Validación de usuario y contraseña	17 :	Gestión clientes
4 :	Conexión base de datos	18 :	Salir del usuario
5 :	Menú principal	19 :	Salir del usuario

6	:	Responsable de la empresa	20	:	Salir del sistema
7	:	Gestión de usuarios	21	:	Salir del sistema
8	:	Gestión de roles y permisos	22	:	Fin ciclo
9	:	Gestión de pedidos	23	:	Fin ciclo
10	:	Gestión de clientes	24	:	Fin ciclo
11	:	Gestión de producción		:	
12	:	Gestión de clasificación de costos		:	
13	:	Salir del usuario		:	
14	:	Personal técnico de producción		:	

**( 24 ) Ecuación Complejidad Ciclomática**

$$V(G) = A - N + 2 \quad ( 24 )$$

Donde:

$$A = \text{Número de aristas} = 31$$

$$N = \text{Número de nodos} = 24$$

$$V(G) = \text{Complejidad}$$

Calculamos:

$$V(G) = 31 - 24 + 6$$

$$\mathbf{V(G) = 13}$$

Identificar el conjunto esencial de trayectorias linealmente independientes, lo cual se requiere evaluar 6 caminos específicos, determinados por ciertas variables.

#### **4.4.2 Prueba de Caja Negra**

Las pruebas de caja negra se centran en verificar los requisitos funcionales del software sin examinar su código interno, evaluando la interfaz y asegurando que el sistema

responda correctamente a las entradas y cumpla con las especificaciones, el objetivo es identificar errores relacionados con el uso práctico del sistema, mejorando así la funcionalidad.

**Figura 63**

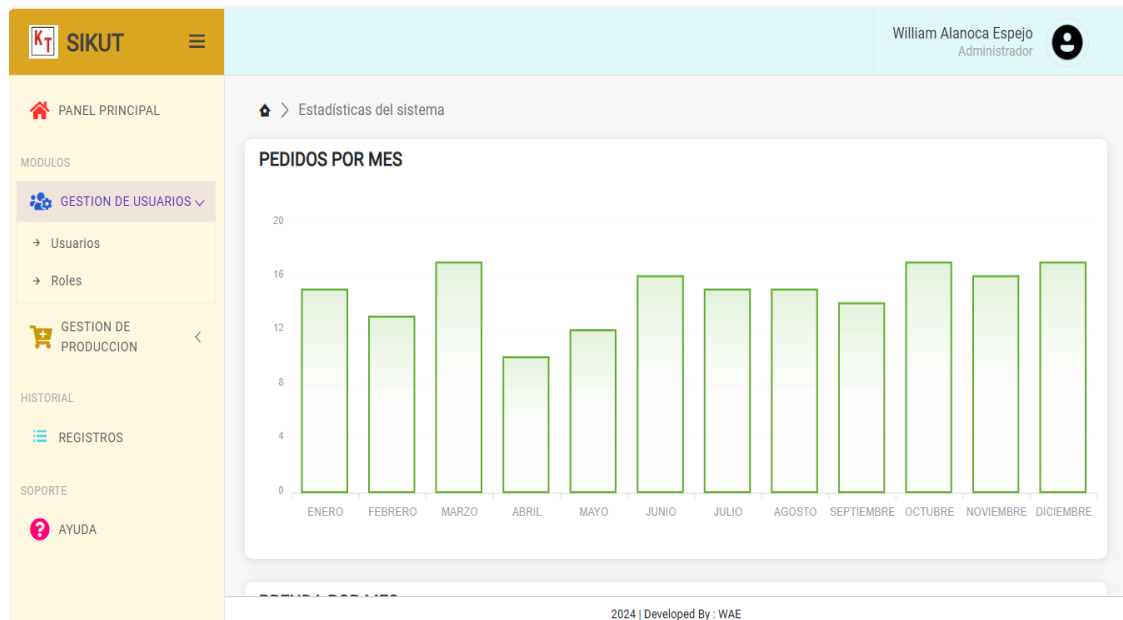
*Prueba de Caja Negra Iniciar Sesión*



**Tabla 30**

*Valores de Caja Negra Login*

CAMPOS	TIPO DE DATOS	ENTRADA INVÁLIDA
Usuario	Cadena de texto	Caracteres especiales, espacios en blanco.
Contraseña	Cadena de texto	Espacios en blanco.
Captcha	Check	-

**Figura 64***Caja Negra Ingreso al Sistema***Tabla 31***Prueba de Caja Negra Ingreso al Sistema*

ENTRADA		RESULTADO
Usuario	admin	El sistema verifica que no queden campos vacíos y al introducir datos correctos, concede el acceso.
Contraseña	*****	
Captcha	✓	

A continuación, se muestra la interfaz de acuerdo a la prueba de caja negra

**Figura 65***Prueba de Caja Negra – Agregar Cliente*
**Tabla 32***Valores Limite - Registrar Pedido*

ENTRADA	TIPO DE DATOS	ENTRADA INVÁLIDA
Descripción	Cadena de texto	Caracteres Especiales, espacios en blanco
Cantidad total	Número	Caracteres Especiales, espacios en blanco
Listado cliente	Cadena de texto	Caracteres Especiales, espacios en blanco
Fecha de pedido	Selección	Caracteres especiales, espacio de selección
Fecha entrega	Selección	Caracteres Especiales, espacios en blanco

**Figura 66***Prueba de Caja Negra – Registro de cliente*

**REGISTRAR NUEVO CLIENTE**

DATOS PERSONALES

NOMBRES: JUANITO ✓

APELLIDO PATERNO: PEREZ ✓

APELLIDO MATERNO: OCHOA ✓

NRO. CI: 9885672 ✓

COMPLEMENTO: LP ✓

EXPEDIDO EN: LA PAZ

NRO. CELULAR: 70503044 ✓

DESCRIPCION: COLBASILLER ✓

DIRECCION: VILLABALLIVIAN ✓

REGISTRAR CANCELAR

**Tabla 33***Prueba de Caja Negra - Registrar pedido*

	CAMPOS	VALORES
<b>DATOS DE ENTRADA</b>	Nombres	juanito
	Ap. Paterno	perez
	Ap. Materno (opcional)	Ochoa
	C.I.	9885672
	Expedido	La paz
	Complemento	
	Dirección	villaballivian
<b>PROCESO DE LA INFORMACIÓN</b>	El sistema requiere solo datos requeridos del propietario si lo es necesario.	Cuando el usuario ingresa datos correctos, el sistema guarda la información en la base de datos.
<b>RESULTADO ESPERADO EN EL SISTEMA</b>	El sistema valida y registra al nuevo cliente	

Después de llevar a cabo la prueba de caja negra en la interfaz de registro de los propietarios, se confirmó que esta funciona según lo previsto, garantizando así que se ingresen los datos necesarios para el proceso, esta validación asegura que no se omita información importante, cumpliendo con los requisitos funcionales establecidos para el correcto funcionamiento del registro, de esta manera el sistema responde adecuadamente a las entradas del usuario mostrando mensajes de “Completa este campo” cuando los campos obligatorios no se completan.

**Figura 67**

*Validación de Campos en el Formulario*

The screenshot shows a web form titled "REGISTRAR NUEVO CLIENTE" under the heading "DATOS PERSONALES". The form contains several input fields with validation errors indicated by red boxes and error messages:

- NOMBRES:** Input field with "EJ: CARLOS MARTIN" and a red error message "INGRESE EL/LOS NOMBRE(S)".
- APELLIDO PATERNO:** Input field with "EJ: OCHOA".
- APELLIDO MATERNO:** Input field with "EJ: SOTO".
- NRO. CI:** Input field with "EJ: 8154770" and a red error message "INGRESE EL NRO. DE CI".
- COMPLEMENTO:** Input field with "EJ: 1A" and a green checkmark.
- EXPEDIDO EN:** Dropdown menu with "SELECCIONE" and a red error message "SELECCIONE UNA OPCION".
- NRO. CELULAR:** Input field with "EJ: 7518998" and a red error message "INGRESE EL NRO. DE CELULAR".
- DESCRIPCION:** Input field with "EJ: INSTITUCIÓN, ETC." and a red error message "INGRESE DESCRIPCION DE CLIENTE".
- DIRECCION:** Input field with "EJ: AV. SIMON BOLIVAR" and a red error message "INGRESE DIRECCION DE CLIENTE".

At the bottom right, there are two buttons: "REGISTRAR" (blue) and "CANCELAR" (pink).

#### 4.4.3 Pruebas de Estrés

Las pruebas de estrés evalúan el comportamiento de una aplicación Web bajo condiciones de carga extrema llevando el sistema al límite de su capacidad operativa.

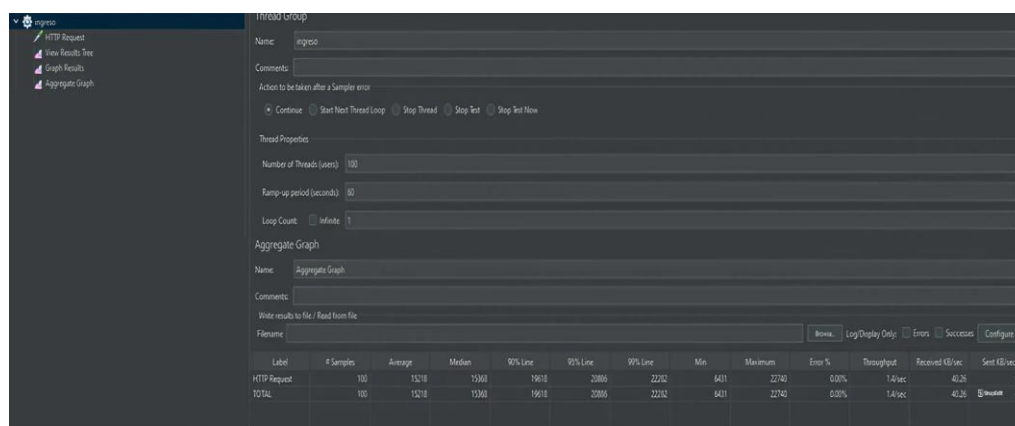
Este proceso permite identificar cómo responde la aplicación en diversos escenarios o situaciones límite, durante esta fase se realizaron pruebas de estrés específicamente en el módulo de reservas.

Para esta prueba, se empleó un servidor interno de la institución, caracterizado por ser de media gama en comparación con los servidores de alta gama, cabe mencionar que se hicieron pocas modificaciones en el sistema operativo, conforme a las requerimientos de la institución, a pesar de estas limitaciones, los resultados obtenidos estuvieron en línea con los promedios de rendimiento exigidos para el sistema, demostrando que el servidor y la configuración empleada fueron adecuadas para las necesidades de la prueba.

Usuarios virtuales: 100, Tiempo de respuesta o rendimiento: 1.4/sec, Solicitudes realizadas Max: 22740, Error HTTP: 0

### Figura 68

*Evaluando Pruebas de Estrés.*



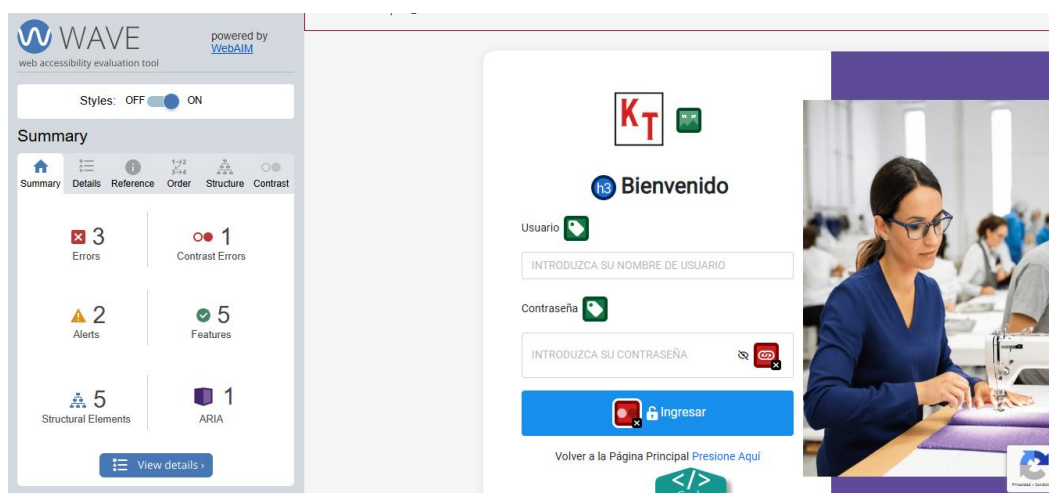
Los datos obtenidos permitirán analizar la estabilidad y la eficiencia del sistema, identificando cualquier posible problema de rendimiento que pueda surgir cuando la página de inicio es accedida por múltiples usuarios al mismo tiempo.

#### 4.4.4 Pruebas de Accesibilidad

En resumen, las pruebas de accesibilidad son una práctica fundamental para optimizar las aplicaciones web y móviles, asegurando que sean utilizables por una amplia audiencia, incluyendo personas que estén en situación de discapacidad, ya sea visual, auditiva u otras situaciones físicas o cognitivas.

**Figura 69**

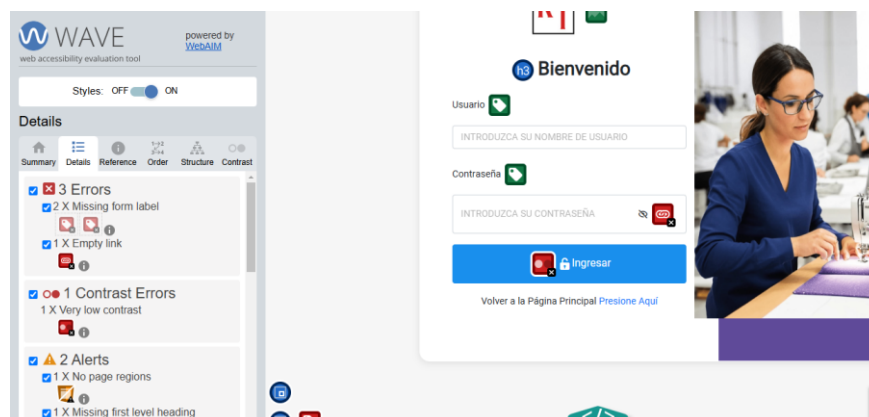
*Prueba de Accesibilidad*



Según la Prueba de Accesibilidad WAVE se puede evidenciar los resultados del software con los siguientes detalles:

**Tabla 34***Resultado de la Prueba de Accesibilidad*

<b>3</b>	<b>errores</b>
<b>2</b>	Etiqueta de formulario faltante
<b>1</b>	Errores de contraste Muy bajo contraste
<b>1</b>	Sin regiones de página
<b>1</b>	Falta el encabezado del primer nivel
<b>1</b>	Falta el título 1 de primer nivel

**Figura 70***Resultado del Testeo*

Según la prueba de accesibilidad se ve los resultados , cual está permitido en el rango de lo que se puede tener fallas en el sistema este margen de errores permite la funcionalidad del sistema con normalidad que no son de consideración.

#### 4.4.5 Pruebas Unitarias

Una prueba unitaria de software —también conocida como *unit testing*— es el instrumento utilizado para validar un fragmento de código fuente. Los desarrolladores aíslan

una línea del lenguaje codificado para saber si el sistema está operando correctamente en una función, proceso o actividad específica.

La palabra unidad alude a un componente individual del sistema que, a su vez, es desglosado por el programa de *testing* para obtener información detallada sobre el funcionamiento y los comportamientos que lo definen.

Según la Prueba Unitaria PHPUnit se pueden evidenciar los resultados del software con los siguientes detalles:

### Figura 71

#### Prueba Unitaria

```

/Applications/XAMPP/xamppfiles/htdocs/curso/65. unit testing/tutorial > master -> ./vendor/bin/phpunit tests
PHPUnit 9.2.2 by Sebastian Bergmann and contributors.

..
                                         2 / 2 (100%)
Time: 00:00.008, Memory: 6.00 MB
OK (2 tests, 2 assertions)
/Applications/XAMPP/xamppfiles/htdocs/curso/65. unit testing/tutorial > master -> ./vendor/bin/phpunit tests
PHPUnit 9.2.2 by Sebastian Bergmann and contributors.

..E
                                         3 / 3 (100%)
Time: 00:00.009, Memory: 6.00 MB

There was 1 error:
]
1) OperationsTest::testSumWithNotNumericValues
A non-numeric value encountered

/Applications/XAMPP/xamppfiles/htdocs/curso/65. unit testing/tutorial/src/Operations.php:18
/Applications/XAMPP/xamppfiles/htdocs/curso/65. unit testing/tutorial/tests/OperationsTest.php:24

ERRORS!
Tests: 3, Assertions: 2, Errors: 1.
/Applications/XAMPP/xamppfiles/htdocs/curso/65. unit testing/tutorial > master ->

```

Según la Prueba Unitaria se puede ver que el sistema se acepta con la medida de errores que presenta ya que no son muy graves para el sistema y nos permite trabajar con normalidad.

## 4.5 ESTIMACIÓN DE COSTO COCOMO II

COCOMO II es un método que proporciona una estimación precisa de los costos asociados con el desarrollo de proyectos de software, este enfoque facilita la elaboración de presupuestos, la planificación y la mejora en la ejecución del proyecto que evalúa el tamaño del software basándose en las líneas de código producidas durante el desarrollo lo que permite

una estimación más exacta de los costos y ayuda a una planificación y a una adecuada asignación de recursos.

El cálculo de los costos del proyecto se realizará utilizando el modelo COCOMO II que se divide en tres niveles: Orgánico, semi acoplado y empotrado, estos niveles se ajustan a diferentes tipos de software y etapas del desarrollo permitiendo una estimación precisa de los costos y el esfuerzo necesario para completar el proyecto con éxito (Galvan, 2014, p. 13).

#### 4.5.1 Modelo Orgánico

Abarca proyectos simples con menos de 5000 líneas de código, centrados en el procesamiento de datos y el uso de bases de datos para transacciones y recuperación de datos.

#### 4.5.2 Modelo Semi Acoplado

Se aplica a proyectos de complejidad y tamaño intermedios, con experiencia variable y restricciones moderadas.

#### 4.5.3 Modelo Empotrado

Aborda proyectos altamente complejos, con poca experiencia y en un entorno de innovación técnica significativa.

Para estimar el esfuerzo requerido en un proyecto de desarrollo de software, es crucial determinar la variable KLDC (líneas de código).

**Tabla 35***Complejidad y Eficiencia del Lenguaje.*

LENGUAJE	NIVEL	FACTOR LDC/PF
C	2.5	128
JAVA	6	53
ANSI COBOL 74	3	107
ASP	9.00	36
PHP	11.00	29
VISUAL C++	9.50	34

*Nota. Conversión de puntos función a KLDC. (Pressman, 2023)*

Esta métrica ofrece una base sólida y cuantificable para valorar el trabajo necesario, facilitando así una planificación más precisa y eficaz del esfuerzo requerido para concluir el proyecto.

**( 25 ) Ecuación Líneas de Código**

$$KLDC = \frac{FLDC * PF}{1000} \quad ( 25 )$$

De acuerdo con la información presentada en la Tabla 30, la evaluación de la complejidad y eficiencia se centró en el lenguaje PHP, se obtuvo un factor LDC/PF de 29, lo cual es significativo para un sistema de tamaño mediano ya que el valor indica el número de líneas de código por punto de función, proporcionando una medida cuantitativa que ayuda a evaluar el esfuerzo y los recursos necesarios para el desarrollo del sistema en PHP.

Al llevar a cabo los cálculos correspondientes, se obtiene el siguiente resultado:

Factor de conversión FLDC = 29

Punto de Función PF = 910.6

Calculando:

$$KLDC = \frac{29 * 910.6}{1000}$$

$$KLDC = 26,40$$

Después de sumar las líneas de código desarrolladas en el framework Laravel, así como las librerías incorporadas en el proyecto, se ha obtenido un total de 4900 líneas de código. Este número se calculó utilizando la fórmula específica para determinar el KLDC (líneas de código) dentro del modelo COCOMO II.

La metodología aplicada permitió obtener una cifra precisa del volumen de código, lo cual es fundamental para estimar el esfuerzo, los costos y los recursos necesarios para el desarrollo del proyecto. Al contar con esta medida cuantitativa, se facilita una planificación más detallada y una gestión eficiente del proyecto, asegurando que se pueda abordar adecuadamente la complejidad y las demandas asociadas al desarrollo del software en Laravel.

$$KLDC = \frac{4900}{1000}$$

$$KLDC = 4.900$$

El siguiente paso en los cálculos es evaluar las características y determinar el Factor de Ajuste de Esfuerzo (FAE), una métrica crucial en el modelo COCOMO II que ajusta la estimación de esfuerzo del desarrollo de software, considerando factores adicionales como la experiencia del equipo, la complejidad del proyecto y el entorno tecnológico.

Para calcular el FAE, se deben analizar características del proyecto como el tamaño del sistema, la complejidad y los requisitos específicos lo que hace que estos ajustes permiten

una estimación más precisa del esfuerzo y los recursos necesarios, ajustando el modelo básico para reflejar mejor las particularidades del proyecto.

**Tabla 36**

*Tabla de Atributos Factores del Esfuerzo*

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		
<b>ATRIBUTOS DEL PROYECTO</b>						
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
<b>TOTAL</b>			<b>0.4933</b>		

La tabla ilustra el cálculo del Factor de Ajuste de Esfuerzo (FAE) mediante la multiplicación de los valores de los atributos relevantes, este procedimiento ajusta la estimación del esfuerzo del proyecto al tener en cuenta factores como la complejidad del sistema de la experiencia del equipo y las condiciones del entorno, como resultado de este ajuste se obtiene un FAE final de 0.7933.

**Tabla 37**

*Coefficientes Cocomo II – Modo Orgánico*

<b>MODO</b>	<b>A</b>	<b>b</b>	<b>C</b>	<b>D</b>
ORGÁNICO	2.40	1.05	2.50	0.38
SEMI ENCAJADO	3.00	1.12	2.50	0.35
EMPOTRADO	3.60	1.20	2.50	0.32

**Nota.** La tabla ilustra los valores para sustituir mediante el modelo básico, Galván, 2014, Modo Orgánico

#### 4.5.4 Esfuerzo Requerido del Desarrollo

Se emplearán los coeficientes simples de la tabla para calcular el esfuerzo necesario, ya que se ajustan de manera óptima al proyecto el cual se establece así una relación directa entre los valores seleccionados y el cálculo del esfuerzo.

**( 26 ) Ecuación Esfuerzo**

$$E = a * KLDC^b * FAE \quad (26)$$

Donde:

<i>a</i> :	Coeficiente Cocomo II	2.40
<i>b</i> :	Coeficiente Cocomo II	1.05
<i>d</i> :	Coeficiente Cocomo II	0.38
<i>FAE</i> :	Factores de atributos del esfuerzo	0.4933
<i>KLDC</i> :	líneas de código	4.900

Calculando:

$$E = 2,40 * 4.900^{1.05} * 0.4933$$

$$E = 6,10 \text{ meses/persona}$$

**4.5.5 Tiempo de Desarrollo Requerido del Sistema**

Para determinar el tiempo de desarrollo del proyecto se utilizarán los datos obtenidos, incluyendo el factor de ajuste de esfuerzo (FAE) y el tamaño del proyecto en líneas de código.

**( 27 ) Ecuación Tiempo Requerido**

$$Tdev = c(E)^d \quad (27)$$

Donde:

<i>c</i>	:	Coeficiente Cocomo II	2.50
<i>d</i>	:	Coeficiente Cocomo II	0.38
<i>E</i>	:	Esfuerzo	7,18

Calculando:

$$Tdev = 2.50(6.10)^{0.38} \text{ meses}$$

$$Tdev = 4,28 \text{ meses}$$

$$Tdev = 4 \text{ meses}$$

#### 4.5.6 Número de Desarrolladores

Se determinará el número de desarrolladores requeridos para el sistema, considerando el esfuerzo estimado del proyecto y el tiempo disponible, para asegurar una asignación adecuada de recursos y cumplimiento de plazos.

**( 28 ) Ecuación Número de Desarrolladores**

$$P = \frac{E}{Tdev} \quad (28)$$

Donde:

$E$  : Esfuerzo 6.10

$Tdev$  : Tiempo requerido 5

Calculando:

$$P = \frac{6.10}{5}$$

$$P = 1.30 \text{ personas}$$

$$P = 2 \text{ personas}$$

#### 4.5.7 Costo Total

Posteriormente se calculará el monto total a pagar por el software utilizado en este proyecto específico, el cálculo se basa en el salario mínimo nacional en Bolivia, establecido en 2,500 Bs, este valor sirve como referencia para estimar los costos asociados al software permitiendo una planificación financiera precisa y adaptada a la economía local del proyecto.

Considerando el salario mínimo, se podrán calcular de manera detallada los costos laborales y otros gastos relacionados, facilitando una gestión adecuada del presupuesto y asegurando que los recursos se utilicen de manera eficiente en el contexto económico del proyecto.

##### ( 29 ) Ecuación Costo del Software

$$C_{sof} = S_{mes} * P * T_{dev} \quad ( 29 )$$

Donde:

$S_{mes}$  : Sueldo mes = 2500 Bs.

$P$  : Número de personas = 2

$T_{dev}$  : Tiempo requerido = 5

Calculando:

$$C_{sof} = 2500 * 2 * 5$$

$$C_{sof} = 25.000 \text{ Bs.}$$

$$C_{sof} = 3.570 \$$$

Según los cálculos realizados se estima que se necesitarán dos programadores trabajando durante ocho meses para completar el desarrollo del software, durante este período el costo total del proyecto se proyecta en 3.570 \$ considerando que el valor del dólar

actualmente en Bolivia es de 6,96 Bs realizando una conversión del costo total del proyecto a moneda local, resultando en una estimación de 25000 Bs. Bs.

- Se requiere 2 desarrolladores
- Un tiempo de desarrollo de 5 meses
- Costo total estimado 3. 570\$ -> 25000 Bs.

# **CAPÍTULO V**

## **CONCLUSIÓN Y**

### **RECOMENDACIONES**

**INGENIERÍA**  
**DE SISTEMAS**  
UNIVERSIDAD PÚBLICA DE EL ALTO



## 5 CAPÍTULO V CONCLUSIONES Y RECOMENDACIONES

### 5.1 CONCLUSIONES

Después de culminar con el Proyecto de Grado denominado “SISTEMA DE INFORMACIÓN WEB DE COSTOS DE PRODUCCIÓN Y DETERMINACIÓN DE UTILIDADES”, desarrollado para la EMPRESA “KUBIC TEXTILES” de acuerdo a los objetivos establecidos, cumpliendo con los requerimientos.

A continuación, a partir de los objetivos planteados, se llegaron a las siguientes conclusiones:

- Se diseño una página web con interfaz amigable al usuario, combinando colores de la empresa donde muestra toda la información correspondiente a la empresa KUBIC TEXTILES.
- Se crearon los perfiles para todos los niveles de acceso y permisos dependiendo del rol a la cual pertenece los usuarios, aplicando la seguridad correspondiente para los accesos.
- Se centralizo toda la información de la empresa, en un gestor de base de datos de licencia gratuita.
- Se implemento el sistema, con un panel de administración de amigable navegación y acceso a la información de acuerdo al rol establecido.
- El acceso a la información de registros se optimizo a través del acceso a la información en tiempo real y en cualquier momento, con la conexión a internet correspondiente.
- Se genera los reportes estadísticos necesarios para la empresa para mejorar la determinación de costos coadyuben en la toma de decisiones a futuro.

## 5.2 RECOMENDACIONES

Ya concluido el presente Proyecto de Grado, se proponen las siguientes recomendaciones a tomar en cuenta:

- Realizar el cambio continuo de contraseñas de accesos para todos los usuarios del sistema.
- Realizar las copias de seguridad de manera periódica, para garantizar el respaldo de la información en caso de pérdida de datos del sistema.
- El mantenimiento periódico del software es muy recomendable al menos en un periodo de cada seis meses.
- El acceso a internet es de vital importancia para la conexión y acceso a la plataforma, por lo que se recomienda tener el servicio estable y de alta disponibilidad.
- Para la actualización de nuevos módulos o nuevas tecnologías, se debe tomar en cuenta el manual de usuario y el manual técnico del Software.

# BIBLIOGRAFÍA

## BIBLIOGRAFÍA

Alarcón, J. (2023). *Cocomo y punto de fusion*.

Barish, R. A., & Arnold, T. (12 de Septiembre de 2022). *Mordeduras MSD*. Obtenido de <https://www.msdmanuals.com/es/hogar/traumatismos-y-envenenamientos/mordeduras-y-picaduras/mordeduras-de-animales>

Bartholomew, D. (2014). *libro de MariaDB*. Packt Publishing.

Brinck, T., & Gergle, D. (2002). *Usability for the Web: Designing Web Sites that Work*. Obtenido de <https://www.uxlumen.com/modelos-de-navegacion/>

Bustamante, P. (2021). *Estimación de esfuerzo de trabajo y planificación de proyectos de desarrollo con restricciones de recursos*. Obtenido de repositorio UCHILE: <https://repositorio.uchile.cl/>

Calero, W. (16 de Octubre de 2010). *Ingeniería de software*. Obtenido de Ingeniería de software: <https://ingenieraupoliana.blogspot.com/2010/10/cocomo.html>

Clutton-Brock. (1999). *Una Historia Natural de los Mamíferos Domesticados*. Prensa de la Universidad de Cambridge.

Flores, A. (20 de Febrero de 2024). *Ecosistemas*. Obtenido de <https://ecosistemas.win/que-animales-silvestres-y-domesticos/>

Galvan, H. (24 de Septiembre de 2014). *SlideShare*. Obtenido de SlideShare: <https://es.slideshare.net/slideshow/cocomo-39479541/39479541>

Gardey, J. (2021). Definicion.de. Obtenido de <https://definicion.de/web-2-0/>.

Gómez, J. (8 de Diciembre de 2013). *Estimación de Costes con COCOMO 81 (II)*. Obtenido de Estimación de Costes con COCOMO 81 (II): <https://www.laboratorioti.com/2013/04/22/estimacion-de-costes-con-cocomo-81-ii/>

Gonzales, E. (2021). Desarrollo Web con PHP y MySQL.

González, E. (2006). [https://www.aprenderaprogramar.com/index.php?option=com\\_content&view=article&i](https://www.aprenderaprogramar.com/index.php?option=com_content&view=article&i)

d=492:ique-es-php-y-ipara-que-sirve-un-potente-lenguaje-de-programacion-para-crear-paginas-web-cu00803b&catid=70&Itemid=193.

Jaimes, J., & Mitzi, N. (2014). *Modelo de estimación del proyecto Cocomo*.

Lucena, P. (Junio de 2023). *Universidad CESUMA*. Obtenido de Universidad CESUMA:  
[https://www.cesuma.mx/blog/que-son-los-sistemas-de-informacion-y-por-que-son-necesarios.html#abh\\_posts](https://www.cesuma.mx/blog/que-son-los-sistemas-de-informacion-y-por-que-son-necesarios.html#abh_posts)

Marrugo, Y. (23 de Junio de 2010). *Modelo Cocomo*. Obtenido de  
<https://es.slideshare.net/jedaro/modelo-cocomo-4593779>

Mendoza, L. E. (18 de Marzo de 2022). *Goconqr*. Obtenido de Goconqr:  
[https://www.goconqr.com/flashcard/35942991/metodologia-extreme-programming-xp#google\\_vignette](https://www.goconqr.com/flashcard/35942991/metodologia-extreme-programming-xp#google_vignette)

Miss, F. G., & Baena, R. (05 de 06 de 2019). *ENUMED.NET*. Obtenido de ENUMED:  
<http://www.eumed.net/rev/ce/2019/norma-iso-eic.html>

*Modelo Cocomo*. (19 de Julio de 2008). Obtenido de  
<https://acevedodelacru.wordpress.com/cocomo-ii/>

Osmosis Latina. (15 de Abril de 2024). *Osmosis Latina*. Obtenido de Osmosis Latina:  
<https://www.osmosislatina.com/jmeter/basico.htm>

Parry, C. M. (2008). *Resistencia a los antimicrobianos en Salmonella: Una Perspectiva Global Enfermedades infecciosas clínicas*.

Platzi. (9 de septiembre de 2018). Obtenido de <https://platzi.com/blog/pruebas-esenciales-para-evaluar-el-rendimiento-de-software/>

Pressman, R. (2023). *Ingeniería de software*.

Risso, A., Ruiz Yanzi, M. V., & Volij, C. (12 de Diciembre de 2022). *evidencia.org*. Obtenido de evidencia.org: <https://www.evidencia.org/index.php/Evidencia/article/view/7049>

Smith, J. S. (1994). *Virus de la Rabia en Microbiología Veterinaria*.

Tenter, A. M. (2000). *Toxoplasma gondii: de los Animales a los Humanos.*" *Revista Internacional de Parasitología.*

Thompson, R. C. (2019). *Zoonoses: Biology and Emerging Issues.*

Vargas, M. S. (25 de Junio de 2017). *Ingeniería basada en modelos.* Obtenido de Ingeniería basada en modelos: <http://marcelosalasvargas.blogspot.com/2017>

Wormser, G. P. (2006). *Guías de práctica de la sociedad de Enfermedades Infecciosas de Estados Unidos para el Tratamiento de la Enfermedad de Lyme.*

Zaptest. (07 de 03 de 2024). *Copyright 2024.* Obtenido de <https://www.zaptest.com/es/pruebas-de-caja-negra-que-son-tipos-procesos-enfoques-herramientas-y-mucho-mas>

# ANEXOS



## Anexo C. Informe

Tabla de precios y cantidades						
Fecha	Materia	Unidad	Cantidad	Porcentaje	Costo	Observaciones
Puede	PA	Tela	100mts	100%	20000	Material directo
	DA	Tela	100mts	100%	20000	
DA	Botones	100mts	100%	100%	5000	Indirectos
	Cierre	20mts	100%	100%	4000	
	Cuerpo	1 cada	100%	100%	2000	
Indice						

Fecha: con fecha el pedido de materiales

Materia: PA: para el pedido del mes DA el segundo pedido DA fabrica directa

Material: → el material es tela - 100mts 100% Botones 100mts 100% etc

Observaciones: → Total en materia la tela 200mts Botones 100mts Cierre 20mts Cuerpo 1 cada 100% etc

Tipo: → el tipo de tela que se usa para el pedido

Costo Directo: → Tela: 200 metros 2-4000 DA → 80000 = 80000 DA

Costo Total: → la multiplicación del costo unitario X la cantidad

## Anexo D. Máquina de Producción



**Anexo E. Maquinas de la Empresa**



**Anexo F. Corte de Produccion**



**Anexo G. Producto Terminado**



# **MANUAL DE USUARIO**



# MANUAL DE USUARIO

**“SISTEMA DE INFORMACIÓN WEB DE COSTOS DE  
PRODUCCIÓN Y DETERMINACION DE UTILIDADES”**

**EMPRESA UNIPERSONAL DE ROPA DEPORTIVA KUBIK TEXTILES**

Versión V1

## **1. INTRODUCCIÓN.**

El sistema SIKUT “Sistema de información web de costos de producción y determinación de utilidades” es un sistema que permite tener un registro de todos sus pedidos y costos que se presentan bajo el cargo de la Empresa KUBIK TEXTILES.

## **2. OBJETIVOS.**

Instruir al usuario en el uso correcto del sistema y la solución de los problemas que puedan suceder en la cada operación.

## **3. REQUERIMIENTOS.**

### **3.1 CONOCIMIENTOS BÁSICOS.**

Manejo de Navegadores de Internet.

### **3.2 REQUERIMIENTOS DE HARDWARE.**

Computadora de escritorio o personal.

Conexión a internet.

### **3.3 REQUERIMIENTOS DE SOFTWARE.**

Sistema Operativo: Windows (7 o superior)

Navegadores: Google Chrome (3.5 o superior), Opera (3.5 o superior), Microsoft Edge (3.5 o superior), Mozilla Firefox (3.5 o superior), o navegadores de distribuciones Linux.

### **3.4 REQUERIMIENTOS DE RED.**

Conexión a la intranet.

## 4 GENERALIDADES.

### 4.1 INGRESO AL SISTEMA.

En la url de su navegador ingrese el siguiente enlace: , inmediatamente le aparecerá una ventana como la mostrada en la Figura 1. Figura 1.

Dirección URL para el acceso del sistema kubik

### 4.2 INICIO DE SESIÓN.

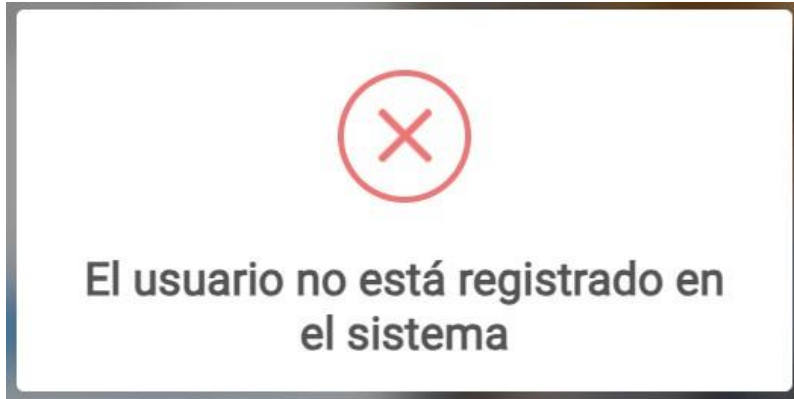
En la ventana de la Figura 2 es donde el usuario deberá ingresar sus datos de autenticación para ingresar al sistema. Figura 2.

*Login (Inicio de sesión) del Sistema SiCat.*



Si los datos ingresados son los correctos, se le permitirá el acceso al sistema como se puede observar en la Figura 5 (cabe recalcar que esta ventana principal es diferente para cada usuario) y si no son los datos correctos le aparecerá el mensaje de la Figura 3. Figura 3.

Usuario no Autenticado



#### 4.3 CERRAR SESIÓN.

Para cerrar la sesión nos dirigimos a la parte superior derecha de la ventana para desplegar las opciones y elegimos Cerrar Sesión.

Tal y como se puede observar en la Figura 4. Figura 4.

Opciones del menú de usuario: Cerrar Sesión



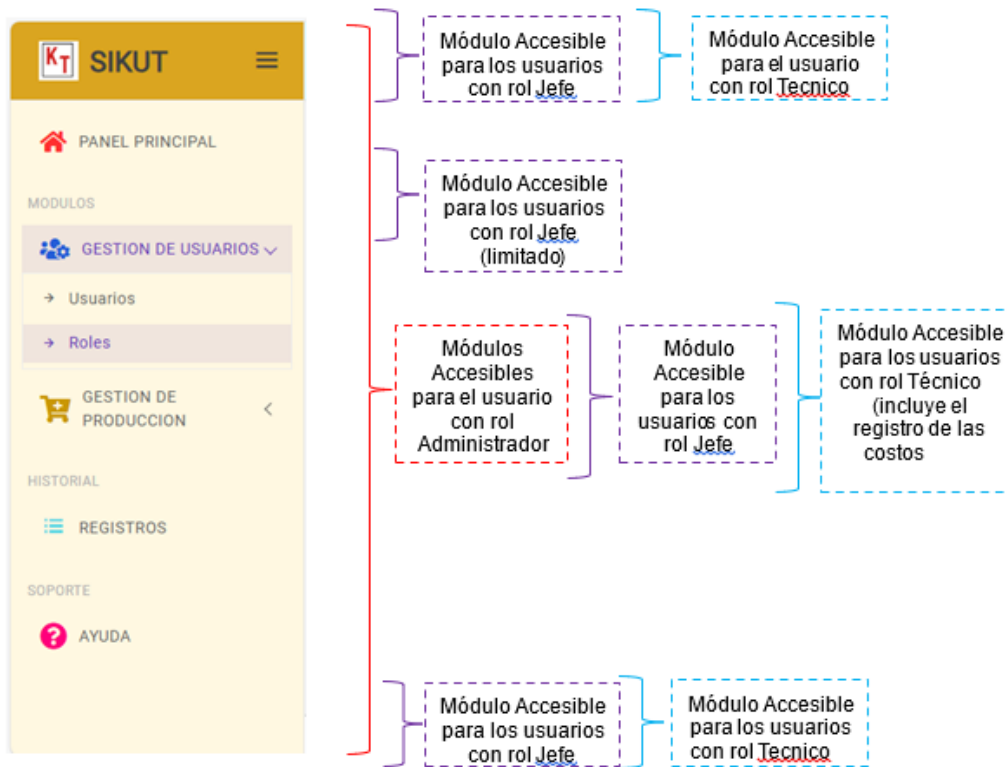
## 5 MENÚ LATERAL IZQUIERDO.

En esta sección se encontrará el menú de vistas disponibles, para los usuarios.

Las opciones están limitadas según el rol del usuario, como se puede ver en la

Figura 5.

*Módulos disponibles para todos los usuarios*

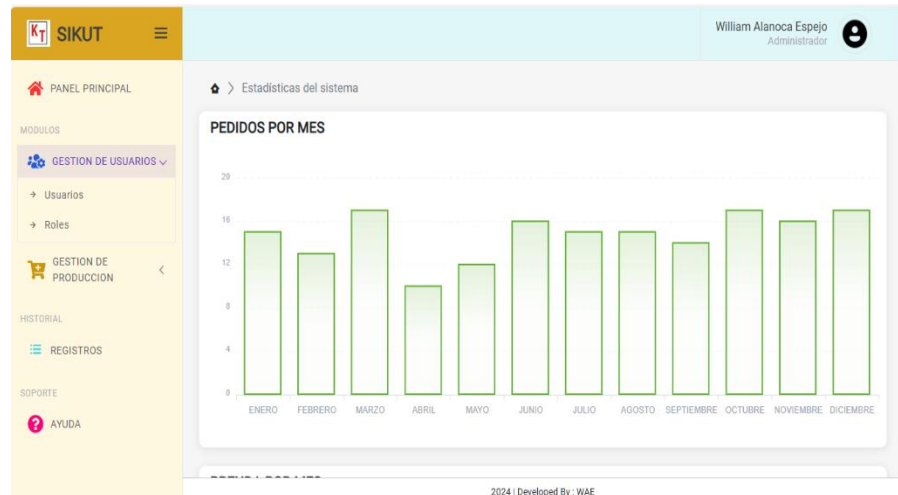


## 5.1 PÁGINA PRINCIPAL.

Como página principal se tiene algunos gráficos que representan los datos que se tienen en el sistema, con respecto a la cantidad de casos que se reciben. Cabe recalcar que estos datos se muestran acorde a los roles de usuario, es decir, los Administradores y jefe pueden ver los gráficos de todos los datos del sistema, mientras que los Terapeutas solo podrán ver los gráficos de todos los casos a los que hayan sido asignados.

Figura 6.

Página principal del sistema, gráficos acerca de los Pedidos



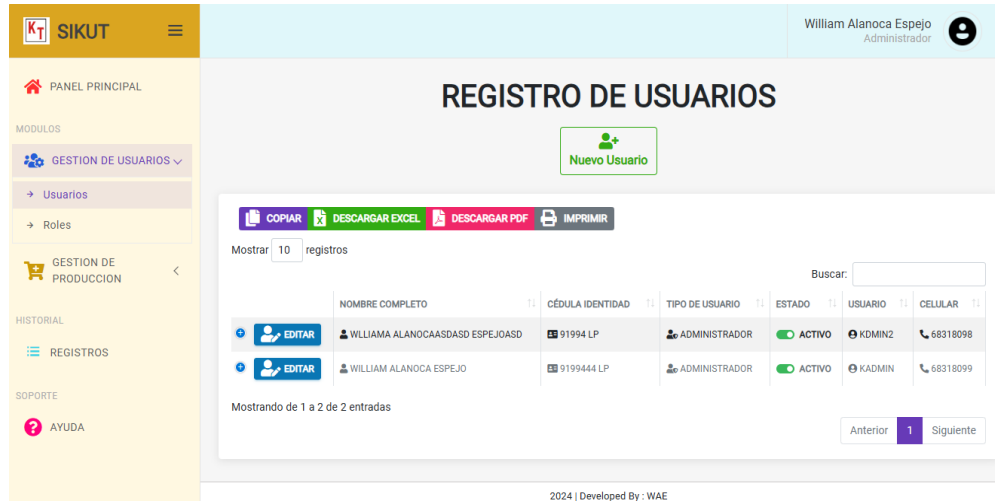
## 5.2 MÓDULO DE USUARIOS / MÓDULO DE TECNICO.

En este módulo se puede visualizar la tabla de registros y el estado de los usuarios registrados en el sistema.

Esta vista solo es accesible para usuarios con el rol: Administrador o Jefe, siendo que los primeros pueden registrar a cualquier usuario en el sistema y los segundos solo pueden registrar Terapeutas.

Figura 8.

### Vista principal del Módulo de Usuarios



#### - 5.2.1 Registrar un Nuevo Usuario:

- Paso 1: Para registrar a un nuevo Usuario primero debe hacer clic al botón

#### **Nuevo Usuario. Figura 9.**

- Botón para el registro de un Nuevo Usuario



- Paso 2: Debe rellenar los campos disponibles en la ventana emergente, el sistema verificará el llenado de los campos obligatorios y dependiendo de los mismos, le permitirá guardar el registro.

**Figura 10.**

- Ventana donde se encuentra el formulario de registro de usuarios



- Paso 3: Presionar el Botón de Registrar para guardar el registro, caso contrario presione Cancelar.

**Figura 11.**

- Botones para Registrar o Cancelar el registro de un nuevo usuario



- 5.2.2 Actualizar Usuario:

- Paso 1: Para actualizar los datos de un Usuario seleccionado primero debe hacer clic al botón correspondiente Figura 12.

- Botones para la actualización de datos de un usuario



- Paso 2: Debe editar los campos disponibles en la ventana emergente, el sistema verificará el llenado de los campos obligatorios y dependiendo de los mismos, le permitirá guardar el registro.

- 5.2.3 Cambiar el Estado de un Usuario:

- Paso 1: Para actualizar el estado de un Usuario (actualizar el estado, permite denegar los permisos de acceso al sistema a cualquier usuario dependiendo de la situación), debe hacer clic en el switch asignado.

**Figura 15.**

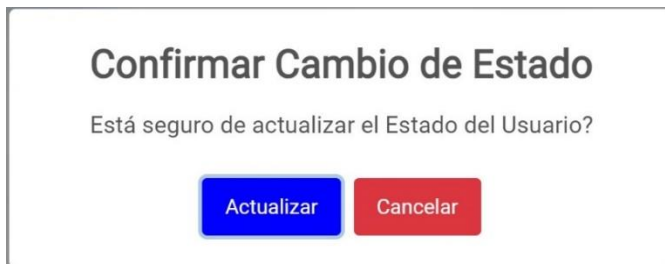
- Botón para actualizar el estado de un usuario



- Paso 2: Presione el botón Actualizar para confirmar el Cambio de Estado, caso contrario presione Cancelar.

**Figura 16.**

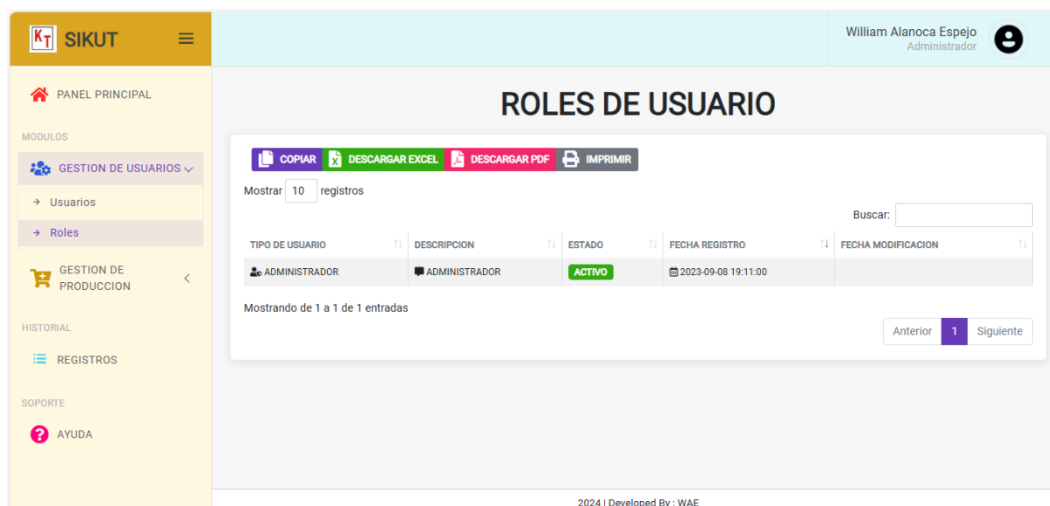
- Botones para la confirmación del cambio estado de un usuario



- 5.3 MÓDULO DE ROLES DE USUARIO.

En este módulo solo es visible para el usuario con rol Administrador, donde se puede visualizar la tabla de registros con los roles de usuario establecidos. Figura 17.

Vista principal del módulo de Roles de Usuario



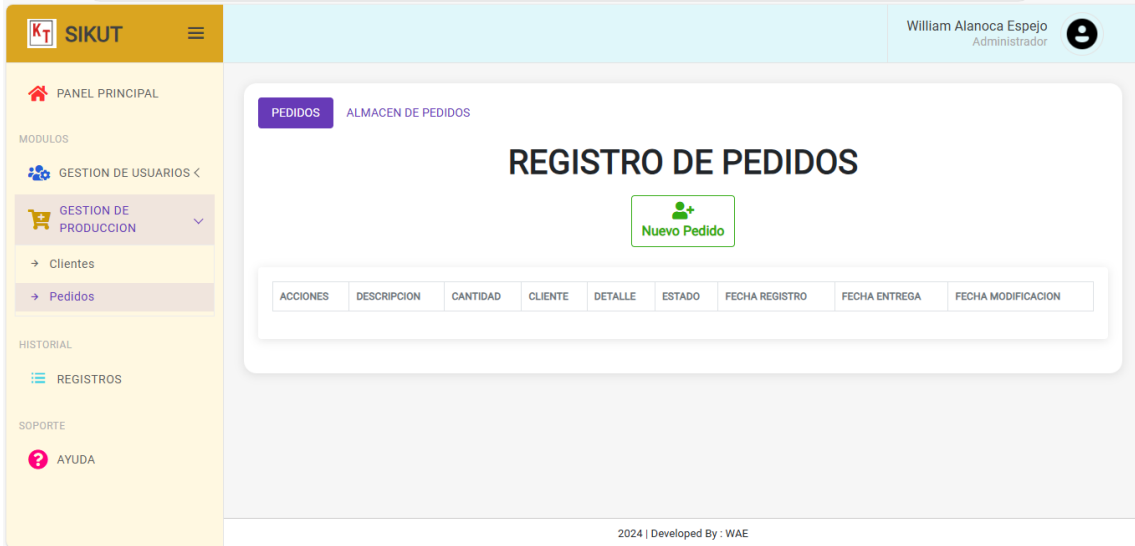
#### 5.2.1.1.1.1.1 5.4 MÓDULO DE GESTIÓN DE PRODUCCION

En este módulo se puede visualizar la tabla de registros (como se puede observar la tabla de registros de pedidos está organizada según su estado, así puede revisar el estado de cada caso según presente cambios con el tiempo) y el estado de los casos que se registren.

Este módulo es accesible para todos los roles de usuario, sin embargo, los Administradores y Jefes pueden realizar mas acciones que los Terapeutas. (Los Administradores y Jefes pueden registrar nuevos casos, actualizar los datos del caso, registrar los afectados que cada caso puede tener, asignar a un terapeuta para dicho caso y cancelar o finalizar el caso dependiendo de la situación, los Terapeutas a los que se les asignaron casos pueden ver los detalles del mismo, y de la lista de afectados que tengan cada caso seleccionar cuales de ellos están dispuestos a tomar las sesiones de terapia y sean pacientes).

#### Figura 18.

- Vista principal del módulo de gestión de Produccion



#### - 5.4.1 Registrar un Nuevo pedidos:

- Paso 1: Para registrar un nuevo Caso primero debe hacer clic al botón

#### **Nuevo Caso.**

#### **Figura 19.**

- Botón para el registro de un nuevo pedido

- Paso 2: Debe rellenar los campos disponibles en la ventana emergente, el sistema verificará el llenado de los campos obligatorios y dependiendo de los mismos, le permitirá guardar el registro.



**Figura 20.**

Ventana donde se encuentra el formulario de registro de pedidos

REGISTRAR NUEVO PEDIDO

DATOS DE PEDIDO

DESCRIPCION PEDIDO: SELECCIONE

CANTIDAD TOTAL: EJ: 50

LISTADO DE CLIENTES: SELECCIONE

FECHA DE ENTREGA: EJ: 2010-03-20

IMÁGEN DEL DISEÑO: Elegir archivo NO SE ELIGIÓ NINGÚN ARCHIVO

REGISTRAR CANCELAR

ACCIONES DESCRIPCION CANTIDAD CLIENTE DETALLE ESTADO FECHA REGISTRO

- Paso 3: Presionar el Botón de Registrar para guardar el registro, caso contrario presione Cancelar.

**Figura 21.**

- Botones para Registrar o Cancelar el registro de un nuevo Pedido



- 5.4.2 Actualizar Pedido:

- Paso 1: Para actualizar los datos de un Pedido seleccionado primero debe hacer clic al menú de acciones disponibles y luego al botón correspondiente.