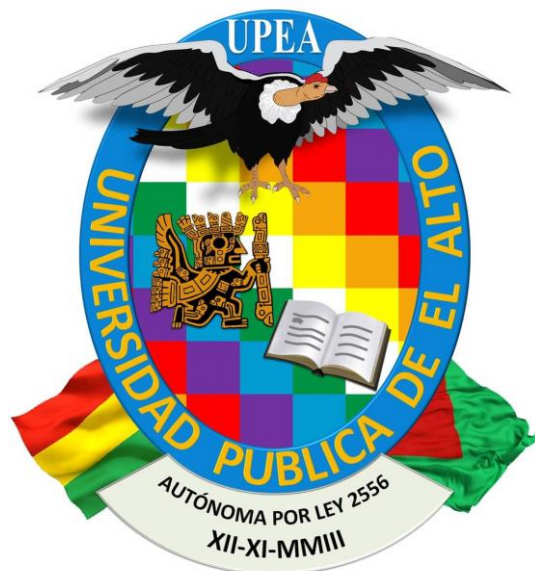


# UNIVERSIDAD PÚBLICA DE EL ALTO

## CARRERA INGENIERÍA DE SISTEMAS



### PROYECTO DE GRADO

**SISTEMA DE INFORMACIÓN WEB PARA EL CONTROL DE PERSONAL  
RESPECTO A LA PRODUCCIÓN Y ALMACENES**

**CASO: COOPERATIVA MINERA BOLSA NEGRA**

Para optar al título de Licenciatura en Ingeniería de Sistemas

**MENCIÓN: INFORMÁTICA Y COMUNICACIONES**

**Postulante: Marín Duran Quispe**

**Tutor Metodológico: M. Sc. Marisol Arguedas Balladares**

**Tutor Especialista: Ing. Rolando Alarcon Choquehuaca**

**Tutor Revisor: Lic. Freddy Salgueiro Trujillo**

**EL ALTO – BOLIVIA**

**2020**

## DEDICATORIA

A mis padres: Saturnino Duran y Sonia Quispe por el gran esfuerzo, amor y confianza que depositaron en mí, por sus sabios consejos y enseñanza. Son lo más importante que tengo en la vida, gracias por todo.

A mis hermanas por brindarme su apoyo incondicional, amor y confianza constante.

Con mucho cariño

Marin Duran Quispe

## **AGRADECIMIENTO**

A Dios por el don de conocimiento y las personas que intervinieron con su apoyo, guía y colaboración en la realización del presente proyecto.

Con mucho cariño a mis padres, a las personas que más aprecio en este mundo, por brindarme afecto, apoyo y comprensión de forma incondicional desde la infancia hasta hoy, por que dedicaron toda una vida entera en inculcarnos a mí y mis hermanas los buenos valores y principios por medio de una buena educación. Al Ing. Marisol Arguedas Balladares, docente de la materia de Taller de Licenciatura II - tutor metodológico, por impulsarme en la culminación del proyecto de grado, así mismo a mis docentes, por haber compartido sus experiencias y por enseñarme las herramientas necesarias para iniciar mi carrera. Al Ing. Rolando Alarcon Choquehuaca, como tutor especialista, por su constante apoyo guiándome a través de su conocimiento y experiencia en las correcciones y sugerencias durante el desarrollo del presente proyecto de grado. Al Ing. Freddy Salgueiro Trujillo. Por dedicar su tiempo en la revisión, orientación y colaboración en el presente proyecto como tutor revisor. Al Presidente de la Cooperativa minera Bolsa Negra Ladislao Prado Z., por brindarme su ayuda en todo el transcurso del desarrollo del proyecto.

## RESUMEN

Actualmente con el avance de la ciencia y tecnología contar con un sistema automatizado es una necesidad en cualquier entidad, por tal razón se ha propuesto el sistema de información web para el control personal con respecto a la producción y almacenes, para que brinde un mejor servicio en el control del personal, materiales de explotación minera y herramientas de la Cooperativa minera Bolsa Negra. Los sistemas de información han sido una ayuda importante en el manejo de gran cantidad de información, así que, tomando el Caso de la Cooperativa Minera, y el volumen de información que genera dicha Cooperativa tendría muchos conflictos a la hora de obtener información confiable oportuna. De manera que el presente proyecto tiene como objetivo desarrollar e implementar un Sistema de Información Web para el Control Personal con respecto a la Producción y Almacenes Caso: Cooperativa Minera Bolsa Negra, que permita un mejor control de personal, producción y almacén. Con el objeto de generar información oportuna y rápida.

Para el análisis y diseño de la aplicación Web se usó la metodología UWE UML (UML-Based Web Engineering) para la construcción y el diseño. UWE es el proceso de desarrollo para aplicaciones web, basada en las técnicas de UML. Para evaluar la calidad del software se utilizó la ISO 9126 y finalmente para la estimación del costo de producto se usó COCOMO II basado en el peso o líneas de código. La arquitectura MVC (Modelo, Vista, Controlador) ayuda al desarrollo a mantener separado los aspectos visuales de la lógica del negocio, PHP es el lenguaje de programación elegido para la crear la aplicación del lado del servidor, Laravel es un framework que utiliza la arquitectura MVC y MariaDB como gestor de base de datos.

## Índice de Contenido

<b>MARCO PRELIMINAR</b> .....	1
<b>1.1</b> Introducción .....	2
<b>1.2</b> Antecedentes .....	3
1.2.1  Antecedente institucionales .....	3
1.2.2  Antecedente académico .....	4
<b>1.3</b> Planteamiento del problema .....	6
1.3.1  Problema principal .....	6
1.3.2  Problemas específicos .....	6
<b>1.4</b> Objetivo .....	7
1.4.1  Objetivo general .....	7
1.4.2  Objetivos específicos .....	7
<b>1.5</b> Justificación .....	8
1.5.1  Técnica .....	8
1.5.2  Económica .....	8
1.5.3  Social .....	9
<b>1.6</b> Metodología .....	9
1.6.1  Método de ingeniería .....	9
<b>1.7</b> Herramientas .....	10
<b>1.8</b> Límites y alcances .....	11
1.8.1  Limites .....	11
1.8.2  Alcances .....	12
<b>1.9</b> Aportes .....	13
<b>MARCO TEORICO</b> .....	15
<b>2.1</b> Sistema .....	16
<b>2.2</b> Sistema de información web .....	17
<b>2.3</b> Producción .....	18
<b>2.4</b> Control personal .....	19
<b>2.5</b> Almacenes .....	21
<b>2.6</b> Ingeniería de requerimientos .....	22
2.6.1  Requerimientos funcionales .....	23
2.6.2  Requerimientos no funcionales .....	23

2.6.3	Características de un requerimiento .....	23
2.6.4	Importancia de la ingeniería de requerimientos .....	24
2.7	UWE UML (UML-Based Web Engineering) .....	25
2.7.1	Fases de la metodología UWE .....	26
2.7.2	Modelo de casos de uso .....	27
2.7.3	Modelo de Contenido .....	28
2.7.4	Modelo de usuario .....	28
2.7.5	Modelo de navegación .....	29
2.7.6	Modelo de procesos .....	29
2.7.7	Modelo de Presentación .....	31
2.8	UML (Unified Modeling Language) .....	32
2.8.1	Modelo .....	33
2.8.2	Vistas .....	33
2.8.3	Diagramas .....	35
2.9	Ingeniería de Software .....	35
2.9.1	Ciclo de vida .....	36
2.9.1.1	Modelo de ciclo de vida .....	38
2.10	Herramientas .....	40
2.10.1	Lenguaje PHP .....	40
2.10.1.1	Características del PHP .....	40
2.10.2	Framework Laravel .....	41
2.10.2.1	ORM .....	41
2.10.2.2	Blade .....	42
2.10.2.3	Composer .....	42
2.10.2.4	MVC .....	43
2.10.3	Servidor web apache .....	44
2.10.4	Base de datos .....	45
2.10.4.1	Generalidades de las bases de datos .....	46
2.10.5	MariaDB .....	46
2.11	Pruebas .....	48
2.11.1	Pruebas de caja negra o funcionales .....	48
2.12	Seguridad en aplicaciones Web .....	49
2.13	Métricas de calidad .....	51

2.13.1	Métricas de calidad establecidas por la norma iso 9126 .....	51
2.13.2	Punto de Función .....	59
2.14	Costos.....	63
2.14.1	Modelo para estimación de costos Cocomo II .....	63
<b>MARCO APLICATIVO .....</b>		<b>73</b>
3.1	Análisis de la situación actual.....	73
3.2	Análisis de requerimientos.....	75
3.2.1	Requerimientos de hardware .....	75
3.2.2	Requisitos de software .....	75
3.2.3	Requerimientos funcionales .....	76
3.2.4	Requerimientos no funcionales.....	77
3.3	Diseño .....	77
3.3.1	Fase de inicio.....	77
3.3.1.1	Identificación de actores .....	77
3.3.2	Fase de elaboración .....	79
3.3.2.1	Modelado de caso de uso .....	79
3.3.2.1.1	Diagrama de caso de uso general .....	80
3.3.2.1.2	Caso de uso: Administrador.....	80
3.3.2.1.3	Caso de uso: jefe de vigilancia .....	87
3.3.2.1.4	Caso de uso: Trabajador .....	90
3.3.2.1.5	Caso de uso: encargado de almacén.....	92
3.4	Desarrollo del modelo .....	95
3.4.1	Modelo de contenido .....	95
3.4.2	Modelo de navegación .....	96
3.4.3	Modelo de presentación.....	97
3.5	Fase de construcción .....	100
3.5.1	Modelo de implementación .....	100
3.5.2	Base de datos.....	111
3.6	Fase de transición .....	111
3.6.1	Prueba de la caja negra.....	111
<b>CAILIDAD Y SEGURIDAD .....</b>		<b>120</b>
4.1	Métricas de calidad.....	120
4.1.1	Usabilidad .....	120

4.1.2	Funcionalidad.....	122
4.1.3	Confiabilidad.....	126
4.1.4	Mantenibilidad .....	127
4.1.5	Portabilidad .....	128
4.2	Seguridad .....	130
4.2.1	Seguridad de software.....	130
ANALISIS DE COSTOS .....		133
5.1	Costo del Software desarrollado .....	133
CONCLUSIONES Y RECOMENDACIONES .....		138
6.1	Conclusiones .....	138
6.2	Recomendaciones.....	139
BIBLIOGRAFIA .....		140
WEBGRAFIA .....		141
ANEXOS.....		142

### Tabla de contenido de figuras

Figura N° 1	Diagrama de Caso de uso .....	27
Figura N° 2	Modelo de Contenido .....	28
Figura N° 3	Diagrama de Navegación.....	29
Figura N° 4	Diagrama de estructuras de procesos .....	30
Figura N° 5	Diagrama de Flujo de procesos.....	30
Figura N° 6	Diagrama de presentación .....	31
Figura N° 7	Capas de la ingeniería de software.....	36
Figura N° 8	Modelo en cascada .....	39
Figura N° 9	Modelo V .....	39
Figura N° 10	Modelo Vista Controlador .....	44
Figura N° 11	Características de la Norma ISO 9126.....	52
Figura N° 12	Factor de ponderación - Punto Función .....	60
Figura N° 13	Escala de Valores .....	62
Figura N° 14	Proceso de cálculo de puntos de función y transformación a líneas de código .....	63
Figura N° 15	Distribución del Mercado de Software Actual y Futuro. ....	64
Figura N° 16	Diagrama de flujo de datos .....	74
Figura N° 17	<i>Diagrama caso de uso general</i> .....	80
Figura N° 18	Caso de uso Administrador.....	81
Figura N° 19	Caso de uso jefe de vigilancia.....	87
Figura N° 20	Caso de uso trabajador.....	90
Figura N° 21	Caso de uso encargado de almacén.....	92
Figura N° 22	Modelo de contenido del sistema.....	95



<b>Figura N° 23</b> Modelo de navegación.....	96
<b>Figura N° 24</b> Modelo de Flujo de Proceso de Autenticar Usuario del Sistema. ....	97
<b>Figura N° 25</b> Pantalla de autenticación del sistema.....	98
<b>Figura N° 26</b> Diagrama de presentación para Control.....	99
<b>Figura N° 27</b> Modelo de Presentación de Modulo de Administración del sistema .....	99
<b>Figura N° 28</b> Modelo de Presentación de Registro.....	100
<b>Figura N° 29</b> Pantalla de Autenticación de usuario.....	101
<b>Figura N° 30</b> Pantalla de control de trabajador.....	101
<b>Figura N° 31</b> Pantalla de menú .....	102
<b>Figura N° 32</b> Pantalla de listado de almacén .....	103
<b>Figura N° 33</b> Pantalla de registro de almacén .....	103
<b>Figura N° 34</b> Pantalla de listado de herramientas .....	104
<b>Figura N° 35</b> Pantalla de listado de materiales .....	104
<b>Figura N° 36</b> pantalla de listado de material entrante.....	105
<b>Figura N° 37</b> Pantalla de listado de salida de material .....	105
<b>Figura N° 38</b> Pantalla de listado de entrada de herramientas.....	106
<b>Figura N° 39</b> Pantalla de listado de salida de herramientas .....	106
<b>Figura N° 40</b> Pantalla de listado de usuarios .....	107
<b>Figura N° 41</b> Pantalla de listado de trabajadores .....	107
<b>Figura N° 42</b> Pantalla de listado de asistencia.....	108
<b>Figura N° 43</b> Pantalla de listado de permiso .....	108
<b>Figura N° 44</b> Pantalla de listado de producción .....	109
<b>Figura N° 45</b> Pantalla de listado de cargos.....	109
<b>Figura N° 46</b> Pantalla de listado de secciones .....	110
<b>Figura N° 47</b> Pantalla de reportes.....	110
<b>Figura N° 48</b> Base de datos del sistema .....	111

## Índice de tablas

<b>Tabla 1</b> Distribución de diagramas.....	35
<b>Tabla 2</b> Coeficientes del modelo COCOMO .....	67
<b>Tabla 3</b> Coeficientes del modelo COCOMO .....	68
<b>Tabla 4</b> Atributo de Coste.....	71
<b>Tabla 5</b> Requerimientos funcionales .....	76
<b>Tabla 6</b> Requerimientos no funcionales.....	77
<b>Tabla 7</b> Especificación del Caso de Uso de Registrar, Modificar y Eliminar Trabajador .....	82
<b>Tabla 8</b> Especificación del Caso de Uso de Registrar, Modificar y Eliminar Material .....	83
<b>Tabla 9</b> Especificación del Caso de Uso de Registrar, Modificar y Eliminar Producción. ....	84
<b>Tabla 10</b> Especificación del Caso de Uso de Registrar, Modificar y Eliminar herramienta.....	85
<b>Tabla 11</b> Especificación del Caso de Uso de Generar Reportes. ....	86
<b>Tabla 12</b> Especificación del Caso de Uso de Registrar, Modificar y Eliminar Producción. ....	88
<b>Tabla 13</b> Especificación del Caso de Uso de Registrar asistencia. ....	89
<b>Tabla 14</b> Especificación del Caso de Uso de Generar Reportes. ....	89

<b>Tabla 15</b>	Especificación del Caso de Uso de Registrar asistencia del trabajador .....	91
<b>Tabla 16</b>	Especificación del Caso de Uso de movimiento de herramientas .....	93
<b>Tabla 17</b>	Especificación del Caso de Uso de movimiento de material .....	94
<b>Tabla 18</b>	Encuesta de usabilidad del sistema .....	121
<b>Tabla 19</b>	Factor de ponderación .....	123
<b>Tabla 20</b>	Valores de ajuste de complejidad .....	123
<b>Tabla 21</b>	Portabilidad .....	129
<b>Tabla 22</b>	Modelo de cocomo .....	134
<b>Tabla 23</b>	Conductores de costo.....	134

# **CAPITULO I**

## **MARCO PRELIMINAR**

## CAPITULO I

### 1.1 Introducción

Muchas cooperativas mineras consideran necesario realizar ajustes en su administración, pero a su vez son conscientes que no son especialistas en administración o saben muy poco de ello. “La administración es nuestro lado flaco” dicen algunos dirigentes del sector. Este problema, sin embargo, no es exclusivo de los cooperativistas mineros. Es un problema general, inclusive del sector privado donde se supone que se encuentran los mejores especialistas en administración porque pagan mejor por el servicio.

Actualmente la cooperativa minera bolsa negra cuenta con 400 personas que forman parte de ella. Y están divididas en 15 secciones por esta razón es difícil controlar lo que cada sección produce por día, porque no se sabe con exactitud los materiales y herramientas que se les entrega a los trabajadores en almacenes, que con el tiempo se pierde, ya que esa información se escribe en un cuaderno el mismo que se encuentra vulnerable a modificaciones, a causa de eso no se puede calcular el monto correcto que se debe pagar por la producción a cada trabajador, ocasionando conflicto entre ellos, por contar con información no confiable y con errores. Derivando en pérdidas económicas para la cooperativa

En ese sentido se propone el desarrollo de un sistema de información que permitirá llevar el control eficiente del personal como también del material y herramientas que están bajo responsabilidad de los trabajador, así mismo se podrá tener la información de forma oportuna y confiable respecto a la producción que cada uno de las secciones logra en una semana u otro periodo de tiempo, de esa manera la cooperativa podrá realizar los cálculos para pagar lo que corresponde a cada trabajador, evitando pagos que no corresponden.

El desarrollo del sistema web para el control personal y almacén estará basado en la metodología UWE UML (UML-Based Web Engineering).

Como herramienta se utilizará el lenguaje PHP por la gran adecuación satisfactoria que presenta con respecto a los requerimientos del sistema que se desea desarrollar, y para el sistema de gestión de base de datos se opta por MariaDB, que nos permitirá almacenar grandes cantidades de información.

## **1.2 Antecedentes**

### **1.2.1 Antecedente institucionales**

La Cooperativa Minera “Bolsa Negra” R.L., fue fundada el 23 de junio de 1977; cooperativa que fue reconocida mediante Resolución N° 00469 de 1 de marzo de 1966, emitida por la Dirección Nacional de Cooperativas, con Registro N° 446 y se encuentra ubicada en Lambate, Provincia Sud Yungas del departamento de La Paz.

#### **Misión. –**

Realizar la explotación de wólfam y oro con la mayor seguridad posible para los socios y trabajadores de la cooperativa, como también con un compromiso con la sociedad y el respeto por el medio ambiente del a región en la que se opera.

#### **Visión. -**

Ser una cooperativa minera boliviana segura y de mayor confianza a nivel nacional y mundial.

## 1.2.2 Antecedente académico

### **Antecedentes Internacionales**

**Maricela Sánchez López y Marcelino Vargas López (2011)** realizaron el trabajo titulado **Sistema de Información para el Control de Inventarios del Almacén del ITS** con el objetivo de diseñar, desarrollar e implementar un sistema de información que permita llevar el registro de artículos, el control de las entradas y salidas de los mismos, así como todas aquellas actividades requeridas en el almacén del Instituto Tecnológico de Saltillo. Para lo cual usaron como metodología el Proceso Unificado del Rational (RUP) apoyado con (UML) Lenguaje de Modelado Unificado, como herramientas utilizaron Visual Basic y Access 2000 como gestor de base de datos.

**Jhon Patrick Gonzales Guevara, Jose Diego Barbaran, Sandoval y Jeter Ali Panduro Panduro(2018)** realizaron el trabajo titulado **Sistema De Información Web Para El Control Del Personal Para Mejorar La Gestión De Recursos Humanos Del Gobierno Regional De Loreto** con el objetivo de mejorar la gestión de procesos del área de recursos humanos de la entidad pública, debido a que habrá una reducción de tiempos y recursos de escritorio, motivo por el cual serán más eficientes. De igual manera el sistema de información web le proporcionará al usuario final la facilidad de emitir a tiempo las planillas del personal y tener un control de asistencias, horarios, suspensiones, asignación de vacaciones entre otros. Para el presente proyecto utilizo como metodología Proceso Unificado del Rational (RUP) juntamente con UML y como lenguajes de programación utilizaron lo que es el HTML y PHP y como gestor de base de datos utilizaron SQL server.

## **Antecedentes Nacionales**

**Jhonny Rubén Monrroy Casillo (2014)** realiza el trabajo titulado **Sistema Web Para El Control Y Administración De Recursos Humanos Caso: Empresa De Limpieza Industrial “Totes Ltda”**. Con el objetivo de realizar reportes del estado de la documentación de cada funcionario, una mejor administración y control de las faltas y permisos del personal, informar sobre contratos en donde existen puestos vacantes, generación de planillas y almacenamiento de la documentación de cada funcionario y contrato en su formato digital para que esta pueda ser consultada de forma inmediata. Para el presente trabajo uso una metodología basada en la metodología de Desarrollo Ágil de Programación Extrema XP (*eXTREME PROGRAMMING*), utilizando java como lenguaje de programación, Framework *Spring* y como gestor de base de datos PostgreSQL.

Por otra parte, también mencionar el trabajo realizado por **Bernardo Angel Villanueva Escobar (2010)** con el título **Sistema de información para el control de almacén central de la caja nacional de salud regional santa cruz**. El proyecto, soluciona los problemas que se tienen en el Almacén Central de la Institución. Ya que su antiguo sistema era mono usuario, duplicaba información, no tenía un mantenimiento adecuado. Bajo estas falencias es que se realizó el proyecto, que proporciona una solución más integrada y mucho más eficiente. Se utilizó como metodología el Proceso Unificado del Rational (RUP) apoyado con (UML) Lenguaje de Modelado Unificado, utilizando como herramientas de desarrollo Microsoft visual Studio, .NET Framework y como gestor de base de datos Microsoft Sql server.

### **1.3 Planteamiento del problema**

#### **1.3.1 Problema principal**

En la cooperativa minera “Bolsa Negra”, el proceso de control se realiza de forma manual, con respecto a: el control de asociados y trabajadores ya que estas se registran en planillas, la entrega materiales y herramientas los cuales se llenan en un formulario o en algunos sin registro alguno, el control de la producción por secciones se anota en planillas de actas o en algunos casos solo se hace el pesado del oro o wólfam y no se registra , lo que genera pérdida de tiempo a la hora de realizar pagos, demora en la obtención de información , la misma no es confiable y como consecuencia genera pérdidas económicas.

#### **1.3.2 Problemas específicos**

- No se realiza el control del personal adecuadamente, lo que ocasiona que no se sepa quiénes están activos en un determinado tiempo.
- El control de los materiales que se utiliza para la explotación no se realiza de forma adecuada, generando demora en el abastecimiento.
- El registro de las salidas de herramientas es manual lo que genera que exista confusión y pérdida de los mismo por no saber a quién se le entrego.
- No existe información confiable respecto a la producción de oro o wólfam de cada sección lo que ocasiona que no se pueda calcular el monto a pagar.
- La elaboración de informes es manual lo que genera que no se tenga información oportuna para poder tomar de decisiones adecuada.



## **1.4 Objetivo**

### **1.4.1 Objetivo general**

Desarrollar un Sistema de Información Web para el Control personal respecto a Producción y Almacenes para la cooperativa minera Bolsa Negra que permitirá llevar el control eficiente del personal como también del material y herramientas que están bajo responsabilidad de cada trabajador, tal que genere información de forma oportuna y confiable para la correcta toma de decisiones.

### **1.4.2 Objetivos específicos**

- Automatizar el control del personal tal que sea adecuado y se pueda saber quiénes están activos en un determinado tiempo.
- Automatizar el control de materiales usados en la explotación adecuado que permita conocer de forma oportuna la información para el abastecimiento.
- Automatizar el registro de las salidas de herramientas de tal manera que apropiado y se pueda identificar a quien se le entrego y no exista confusiones.
- Generar información oportuna y confiable sobre la producción que permita calcular el monto adecuado a pagar.
- Optimizar el tiempo de procesamiento y búsqueda de datos que permita una buena toma de decisiones.

## **1.5 Justificación**

### **1.5.1 Técnica**

Dada la situación actual de la cooperativa minera en la que el desarrollo de control personal y de los materiales necesarios para la explotación de oro y wólfram genera un elevado volumen de información tanto en la elaboración de reportes de salida de materiales, así como tipos de materiales que se adquieren, nos deja una clara necesidad de implementar un sistema de información web para el control de la cooperativa, para lo cual se utilizará metodologías y herramientas como ser UML, UWE, PHP.

La cooperativa minera Bolsa Negra actualmente cuenta con la infraestructura, recursos hardware y software para la implementación del Sistema de Información Web, lo que hace que el proyecto sea viable.

### **1.5.2 Económica**

La cooperativa minera cuenta con el capital para poder implementar el sistema a realizarse.

El Sistema de información web reduce en gran medida los gastos de tiempo a la hora de realizar reportes, empleado en los procesos de registro y verificación realizados diariamente, de modo que se logre mayor eficiencia en la obtención de información, de tal manera que les permita una buena toma de decisiones a la hora de realizar los pagos correspondientes a los trabajadores.

### 1.5.3 Social

La implementación del sistema beneficiara a la cooperativa minera Bolsa Negra tanto al personal operativo como al directivo, permitiendo búsquedas de información rápida y oportuna para la toma de decisiones y por ende mejorar la eficiencia en el control personal. El sistema también será beneficioso para la parte de contabilidad a la hora de pagar los salarios y no exista problemas sociales ya que existirá un registro de los días trabajados, como también a la hora de realizar la entrega de herramientas será más fácil registrar a los trabajadores que se apersonen por una herramienta.

## 1.6 Metodología

### 1.6.1 Método de ingeniería

**UWE - UML-based WEB Engineering.** UWE es un método, de ingeniería WEB orientada a objetos basada en UML, que puede ser utilizado para la especificación de aplicaciones WEB.

La aproximación propuesta por UWE provee:

Una notación específica de dominio, un proceso de desarrollo basado en el modelo, y una herramienta de soporte para la ingeniería de aplicaciones WEB.

La principal característica de UWE es el hecho de ser una aproximación basada en estándares, la cual no se limita al uso de UML, además integra:

XMI como modelo de intercambio de formatos, MOF para los metamodelos, los principios de la aproximación MDA (dirigida por el modelo), el modelo de transformación del lenguaje QVT y XML

La razón principal para extender UML en lugar de crear una técnica de modelamiento propietaria, es la aceptación de UML en el proceso de desarrollo de software, la flexibilidad para la definición de un lenguaje de modelamiento específico en el dominio WEB, también llamado perfil UML, y un gran soporte del modelo de visualización con las herramientas existentes de UML CASE. (Mínguez, 2007).

**El Lenguaje de Modelamiento Unificado (UML - *Unified Modeling Language*)** es un lenguaje gráfico para visualizar, especificar y documentar cada una de las partes que comprende el desarrollo de software. UML entrega una forma de modelar cosas conceptuales como lo son procesos de negocio y funciones de sistema, además de cosas concretas como lo son escribir clases en un lenguaje determinado, esquemas de base de datos y componentes de software reusables. Es un lenguaje de modelado formado por símbolos y es utilizado por muchas metodologías. De hecho, es un lenguaje para comunicar cosas, por lo tanto, se utiliza en la documentación. Por otro lado, la herencia, no es lo más usado en Java. Lo más usado es composición. En una clase siempre se hereda una sola vez, pero siempre se componen de muchos objetos de tipos muy variados. Es un error suponer que la herencia es la característica más importante de la POO, y el error más común es utilizarla a diestra y siniestra sin sentido solo porque es un buen recurso. (Mylena Rodriguez, 2012).

## **1.7 Herramientas**

**PHP (Hypertext Pre-Processor):** Es un lenguaje de programación muy potente que, junto con html, permite crear sitios web dinámicos. Php se instala en el servidor y funciona con versiones de Apache, Microsoft IIS, Netscape Enterprise Server y otros.

La forma de usar php es insertando código php dentro del código html de un sitio web. Cuando un cliente (cualquier persona en la web) visita la página web que contiene éste código, el servidor lo ejecuta y el cliente sólo recibe el resultado. Su ejecución, es por tanto en el servidor, a diferencia de otros lenguajes de programación que se ejecutan en el navegador.

Php permite la conexión a numerosas bases de datos, incluyendo MySQL, Oracle, ODBC, etc. Y puede ser ejecutado en la mayoría de los sistemas operativos (Windows, Mac OS, Linux, Unix (Lina Nova, 2011)).

**MariaDB:** Es un sistema de gestión de bases de datos derivado de MySQL con licencia GPL (General Public License). Es desarrollado por Michael (Monty) Widenius fundador de MySQL, la fundación MariaDB y la comunidad de desarrolladores de software libre introduce dos motores de almacenamiento nuevos, uno llamado Aria que reemplaza a MyISAM y otro llamado XtraDB en sustitución de InnoDB. Tiene una alta compatibilidad con MySQL ya que posee las mismas órdenes, interfaces, API y bibliotecas, siendo su objetivo poder cambiar un servidor por otro directamente.

Este SGBD surge a raíz de la compra de Sun Microsystems compañía que había comprado previamente MySQL, por parte de Oracle. MariaDB es una bifurcación directa de MySQL que asegura la existencia de una versión de este producto con licencia GPL. (Michael Widenius, 2009).

## **1.8 Límites y alcances**

### **1.8.1 Límites**

- Solo se podrá acceder al sistema si son personas autorizadas.

- El sistema solo estará orientado para la cooperativa minera Bolsa Negra.
- El sistema se concentra específicamente en la ejecución de los procesos que conciernen al control personal respecto a producción y almacenes.

### 1.8.2 Alcances

El presente proyecto mejorara el manejo de información de la cooperativa minera, para todos los accionistas y trabajadores que tiene la cooperativa, aumentando a si la eficiencia y la eficacia del control personal con respecto a los que es la producción y material, herramientas existentes en almacenes.

Por lo que se realizara una buena organización de los datos que se obtengan para el realizado del sistema, a si también realizar una interfaz sencilla y de fácil manejo para el administrador, con lo que se lograra mejorar el tiempo de la administración de datos bajo los siguientes módulos:

- **Módulo usuario**, se otorgará acceso al sistema y se aplicará medidas de seguridad en cuanto al ingreso al sistema.
- **Módulo personal**, en este módulo estará registrado el personal que existe en la cooperativa.
- **Módulo producción**, se creará registros para almacenar información sobre la producción que se hace por secciones en la cooperativa
- **Módulo almacén**, donde se registrará la cantidad de material y herramientas que existente en el almacén.

- **Módulo entrega material**, se registrará todo el material que se entrega a los encargados de las secciones que trabajan en la cooperativa.
- **Modulo ingreso material**, Donde se podrá registrar los materiales que ingresan al almacén.
- **Modulo entrega herramientas**, donde se registrará las herramientas que se entrega a los trabajadores de la cooperativa.
- **Modulo entrada herramientas**, donde se registrará las nuevas herramientas que entra al almacén.
- **Modulo reportes**, en la cual se pueden ver los reportes de: existencia material, ingreso material, entrega material y herramientas como también la producción.

## 1.9 Aportes

Los aportes que se brindan con este proyecto se puede enfocar de dos puntos de vista que son: aporte del proyecto y aporte académico.

En aporte académico es mejorar los procesos de control personal, producción y materiales existentes en el almacén de la cooperativa, utilizando para el desarrollo del sistema metodologías y herramientas como ser el método de ingeniería web con uso de la metodología UWE (UML Basado en Ingeniería Web) y las herramientas libres para el desarrollo del software: PHP, MariaDb, framework Laravel que facilita la elaboración del software.

Los aportes que se brindan con este proyecto será automatizar sus procesos rutinarios, minimizar y optimizar tiempos de ejecución generando información que coadyuve a la fácil

y al correcto control de la cooperativa. El sistema permitirá mejorar los procesos de control personal, producción y material existentes en el almacén implementando diferentes módulos lo cual facilitará la emisión de resultados más rápidos y completos.



**CAPITULO II**

**MARCO TEORICO**

## CAPITULO II

### MARCO TEORICO

En este capítulo se presentan los conceptos más relevantes sobre las metodologías, herramientas utilizadas para el desarrollo del presente proyecto.

#### 2.1 Sistema

Un sistema es un conjunto de partes o elementos organizados y relacionados que interactúan entre sí para lograr un objetivo. Cabe aclarar que las cosas o partes que componen al sistema, no se refieren al campo físico (objetos), sino más bien al funcional. De este modo las cosas o partes pasan a ser funciones básicas realizadas por el sistema. Los sistemas reciben (entrada) como ser datos, (proceso) es lo que transforma una entrada en salida y (salida) son los resultados que se obtienen de procesar las entradas, es decir información. (Von Bertalanffy,1968).

La teoría General de Sistemas, más que teoría se trata de una concepción estructurada o metodología que tiene como propósito estudiar el sistema como un todo, de forma íntegra, tomando como base sus componentes y analizando las relaciones e interrelaciones existentes entre éstas y mediante la aplicación de estrategias científicas, conducir al entendimiento globalizante y generalizado del sistema. La Metodología General de Sistemas reúne los elementos necesarios para difundir y hacer extensiva su propia aplicación, es una metodología que permite elaborar modelos y pronosticar como se comportarán antes de su puesta en marcha mediante la aplicación de procesos de simulación, permitiendo seleccionar la mejor alternativa a la problemática analizada. (Alonzo Tamayo A.,1999).

Analizando los dos conceptos anteriores un sistema es un conjunto de elementos relacionados entre sí que funciona como un todo, si bien cada uno de los elementos de un sistema puede funcionar de manera independiente, siempre formará parte de una estructura mayor, del mismo modo un sistema puede ser a su vez un componente de otro sistema.

## **2.2 Sistema de información web**

Los sistemas de información han ido evolucionando a lo largo de los años empleando diferentes programas y entornos hasta encontrar, en la Web, el hábitat ideal. La World Wide Web emplea un conjunto de recomendaciones, técnicas y tecnologías muy robustas que ofrecen un gran rendimiento y fiabilidad, así como un mecanismo de navegación muy familiar para la mayoría de usuarios. Esto favorece los procesos de enseñanza de los programas basados en la Web, ya que se parte de conceptos y herramientas ampliamente conocidos, como puede suceder con los navegadores o los métodos de búsqueda y navegación.

Estos sistemas de información cuentan con una arquitectura común, al contrario de lo que sucede con otras soluciones creadas específicamente para trabajar en sistemas aislados. Esto favorece que se puedan retroalimentar unas aplicaciones de otras, sin que necesariamente pertenezcan a la misma empresa o incluso sin que hayan sido diseñadas para realizar las mismas funciones, Además se facilita la migración de datos de una aplicación a otra, así como la interconexión de la información procedente de diferentes fuentes, pero que empleen la misma arquitectura. Al estar dentro de un entorno tan dinámico y cambiante, la mayoría de estos sistemas tienen la necesidad de estar en constante evolución, presentando continuas versiones que actualizan las funciones y posibilidades de las anteriores. (José A. Senso, 2016).

El sistema de información, es el conjunto formal de procesos que, operando sobre una colección de datos estructurada de acuerdo con las necesidades de una empresa, recopila, elabora y distribuye la información necesaria para la operación de dicha empresa y para las actividades de dirección y control correspondientes, apoyando, al menos en parte, los procesos de toma de decisiones necesarios para desempeñar las funciones de negocio de la empresa de acuerdo con su estrategia.(Andreu, Ricart y Valor, 1991).

Sintetizando las definiciones anteriores todo sistema de información utiliza como materia prima los datos, los cuales almacena, procesa y transforma para obtener como resultado final información, la cual será suministrada a los diferentes usuarios del sistema, existiendo además un proceso de retroalimentación, en la cual se ha de valorar si la información obtenida se adecua a lo esperado.

### **2.3 Producción**

La producción de una empresa puede medirse en un determinado volumen, la diferencia entre el volumen de lo producido en términos de dinero en relación a los bienes consumidos da cuenta del valor que se ha añadido a esos recursos. Así, según la diferencia que se haga de la utilización de los factores de producción con respecto a los valores de producción final se tendrá referencia a la rentabilidad o ganancia de la organización comercial. Las empresas están continuamente midiendo, reorganizando y combinando estos factores de modo cada vez más novedoso a efectos de bajar los costos o dar cuenta de bienes o servicios cuya alta demanda ofrezca un precio superior y por lo tanto las ganancias sean más elevadas. (Editorial Definición MX, 2016).

La producción es la actividad económica que se encarga de transformar los insumos para convertirlos en productos, la producción es cualquier actividad que aprovecha los recursos y las materias primas para poder elaborar o fabricar bienes y servicios, que serán utilizados para satisfacer una necesidad.

También se podría decir que la producción es una actividad dirigida a la satisfacción de las necesidades humanas, a través del procesamiento de las materias primas, hasta generar productos o mercancías, que serán intercambiadas dentro del mercado.

Por lo tanto, la definición económica de producción es muy amplia, ya que se pueden incluir como producción cualquier actividad humana donde se está elaborando un satisfactor de una necesidad determinada. (Miriam Quiroa, 2007).

La producción constituye la actividad a través de la cual el trabajo humano genera beneficios para el sector económico de una empresa o de la persona mismo. En este sentido, está conformada por la utilidad que crea la elaboración, fabricación u obtención de determinados productos, bienes o servicios, una caída en la producción puede afectar negativamente su economía, mientras que un alza de producción aumenta mayores ganancias para la empresa o en este caso para una cooperativa.

## **2.4 Control personal**

El control de personal o control de colaboradores permite integrar una serie de procedimientos con la finalidad de registrar y controlar al personal que trabaja en una empresa o institución.

Los terminales de control de personal también se denominan terminales de control de presencia, control de asistencia, control horario o relojes de fichar. Estos sistemas tienen

definidos para cada usuario registrado, unos calendarios, horarios, vacaciones, turnos de trabajo, etc. en que el usuario registrado tiene permiso.

Otra característica común de los terminales de control de personal es la posibilidad de descargar el registro o logs del histórico de accesos, donde hay información de la identidad de cada usuario, el código de incidencia de acceso, la hora del acceso.

La autenticación de cada usuario puede realizarse mediante tecnología biométrica, proximidad, tarjeta inteligente, banda magnética, código de barras o password. (kimaldi, 2017).

El control de personal es un proceso (una serie de etapas ordenadas) y que tiene como finalidad conocer las incidencias sobre la asistencia del personal que se dan en la empresa, para lo que se sirve de una serie de herramientas de recogida, registro y tratamiento de la información. Su finalidad se puede concretar en dos objetivos:

- Evaluación del desempeño. Detectar los problemas de funcionamiento del personal y determinar las causas que los producen para tomar decisiones que mejoren la situación.
- Cumplimiento de la disciplina. Controlar el cumplimiento de las normas por parte de los trabajadores respecto a las entradas y salidas, el cumplimiento del horario de trabajo, la realización de horas extras, los permisos, las vacaciones, las licencias y los retrasos. (Rafael Castaño, 2005).

Del análisis de las definiciones sobre control personal se puede concluir que el control personal es una forma de controlar las funciones que cumple un trabajador en una entidad, como en este caso una cooperativa minera, el control personal se aplica para que los objetivos

de la empresa se cumplan y el ambiente de trabajo sea adecuado, es preciso que el capital humano de la empresa cumpla sus tareas en el tiempo estipulado. Por tanto, es necesario llevar a cabo un proceso de control de personal.

## **2.5 Almacenes**

Son aquellos lugares donde se guardan los diferentes tipos de mercancía. La formulación de una política de inventario para un departamento de almacén depende de la información respecto a tiempos de adelantes, disponibilidades de materiales, tendencias en los precios y materiales de compras, es la fuente mejor de esta información. Esta función controla físicamente y mantiene todos los artículos inventariados, se deben establecer resguardo físico adecuados para proteger los artículos de algún daño de uso innecesario debido a procedimientos de rotación de inventarios defectuosos de rotación de inventarios defectuosos y a robos. Los registros se deben mantener, lo cual facilitan la localización inmediata de los artículos. (Reynier, 2017).

El almacén es una unidad de servicio en la estructura orgánica y funcional de una empresa comercial o industrial, con objetivos bien definidos de resguardo, custodia, control y abastecimiento de materiales y productos. El éxito en el desarrollo de los programas de trabajo de una empresa depende, en gran medida del adecuado suministro de los equipos, materiales o artículos de la calidad requerida, en las cantidades solicitadas y en el tiempo oportuno. En el estudio y la aplicación de la administración moderna, el buen manejo de un almacén es un medio para lograr eficiencia en los procesos de recepción, control, consolidación y envío de productos disminuyendo mermas y deterioro generando un valor agregado en los insumos, este concepto ahuyenta la idea de que un almacén es un mal

necesario cuya función principal es la de agregar gastos y disminuir utilidades. (García, Cantu, 1996).

Analizando las definiciones anteriores se llega a la conclusión de que un almacén es un lugar que sirve para el resguardo de las mercaderías, materiales, etc., como también para que un buen control a la hora de recepcionar o despachar lo necesario para los trabajadores en caso de una empresa industrial o como en este caso una cooperativa minera.

## **2.6 Ingeniería de requerimientos**

La ingeniería de requerimientos sirve como una base sólida en el proceso de desarrollo de software, por lo que antes de pasar a tratar los aspectos referentes a la administración adecuada de los requerimientos, es importante primero definir lo que es un requerimiento y cuáles serían las características deseables que deberían de tener.

La Ingeniería de Requerimientos (IR) cumple un papel primordial en el proceso de producción de software, ya que se enfoca un área fundamental: la definición de lo que se desea producir. Su principal tarea consiste en la generación de especificaciones correctas que describan con claridad, sin ambigüedades, en forma consistente y compacta, las necesidades de los usuarios o clientes; de esta manera, se pretende minimizar los problemas relacionados por la mala gestión de los requerimientos en el desarrollo de sistemas. Sirve como una base sólida en el proceso de desarrollo de software, por lo que antes de pasar a tratar los aspectos referentes a la administración adecuada de los requerimientos. Los requerimientos de software pueden dividirse en 2 categorías: requerimientos funcionales y requerimientos no funcionales. (Michael Arias C., 2006).



### **2.6.1 Requerimientos funcionales**

Los requerimientos funcionales son los que definen las funciones que el sistema será capaz de realizar, describen las transformaciones que el sistema realiza sobre las entradas para producir salidas. Es importante que se describa el ¿Qué? y no el ¿Cómo? se deben hacer esas transformaciones. Estos requerimientos al tiempo que avanza el proyecto de software se convierten en los algoritmos, la lógica y gran parte del código del sistema.

### **2.6.2 Requerimientos no funcionales**

Los requerimientos no funcionales tienen que ver con características que de una u otra forma puedan limitar el sistema, como, por ejemplo, el rendimiento (en tiempo y espacio), interfaces de usuario, fiabilidad (robustez del sistema, disponibilidad de equipo), mantenimiento, seguridad, portabilidad, estándares, etc.

### **2.6.3 Características de un requerimiento**

Es importante no perder de vista que un requerimiento debe ser:

- Especificado por escrito: Como todo contrato o acuerdo entre dos partes.
- Posible de probar o verificar: Si un requerimiento no se puede comprobar, entonces ¿cómo se sabe si se cumplió con él o no?
- Conciso: Un requerimiento es conciso si es fácil de leer y entender. Su redacción debe ser simple y clara para aquellos que vayan a consultarlo en un futuro.
- Completo: Un requerimiento está completo si no necesita ampliar detalles en su redacción, es decir, si se proporciona la información suficiente para su comprensión.

- **Consistente:** Un requerimiento es consistente si no es contradictorio con otro requerimiento.
- **No ambiguo:** Un requerimiento no es ambiguo cuando tiene una sola interpretación. El lenguaje usado en su definición, no debe causar confusiones al lector.

#### **2.6.4 Importancia de la ingeniería de requerimientos**

Según la autora Lizka Johany Herrera (2003) en su documento de la ingeniería de requerimientos, los principales beneficios que se obtienen de la Ingeniería de Requerimientos son:

- **Permite gestionar las necesidades del proyecto en forma estructurada:** Cada actividad de la IR consiste de una serie de pasos organizados y bien definidos.
- **Mejora la capacidad de predecir cronogramas de proyectos, así como sus resultados:** La IR proporciona un punto de partida para controles subsecuentes y actividades de mantenimiento, tales como estimación de costos, tiempo y recursos necesarios.
- **Disminuye los costos y retrasos del proyecto:** es sabido que reparar errores por un mal desarrollo no descubierto a tiempo, es sumamente caro; especialmente aquellas decisiones tomadas durante la IR, ya que es una de las etapas de mayor importancia en el ciclo de desarrollo de software y de las primeras en llevarse a cabo.

- Mejora la calidad del software: La calidad en el software tiene que ver con cumplir un conjunto de requerimientos (funcionalidad, facilidad de uso, confiabilidad, desempeño, etc.).
- Mejora la comunicación entre equipos: La especificación de requerimientos representa una forma de consenso entre clientes y desarrolladores. Si este consenso no ocurre, el proyecto no será exitoso.
- Evita rechazos de usuarios finales: La ingeniería de requerimientos obliga al cliente a considerar sus requerimientos cuidadosamente y revisarlos dentro del marco del problema, por lo que se le involucra durante todo el desarrollo del proyecto.

## **2.7 UWE UML (UML-Based Web Engineering)**

UML-Based Web Engineering (UWE) es una metodología de desarrollo de aplicaciones web, utilizada en la ingeniería web, dedicado a la sistematización y personalización, es decir realizar sistemas adaptativos.

Debemos también destacar las características relevantes del proceso UWE como la utilización del paradigma orientado a objetos, su orientación al usuario.

UWE es un proceso, iterativo e incremental, incluye flujos de trabajo y puntos de control, las fases coinciden con las propuestas en el Proceso Unificado de Modelado y UML, pero adaptada a la web. (Mínguez, 2007).

El modelo que propone UWE está compuesto por 6 etapas o sub-modelos:

- Modelo de Casos de Uso: modelo para capturar los requisitos del sistema.

- Modelo de Contenido: es un modelo conceptual para el desarrollo del contenido.
- Modelo de Usuario: es modelo de navegación, en el cual se incluyen modelos estáticos y modelos dinámicos.
- Modelo de estructura: en el cual se encuentra la presentación del sistema y el modelo de flujo.
- Modelo Abstracto: incluye el modelo a de interfaz de usuario y el modelo de ciclo de vida del objeto.
- Modelo de Adaptación.

### **2.7.1 Fases de la metodología UWE**

UWE proporciona guías para la construcción de modelos de forma sistemática enfocándose en la personalización y en el estudio de caso de uso.

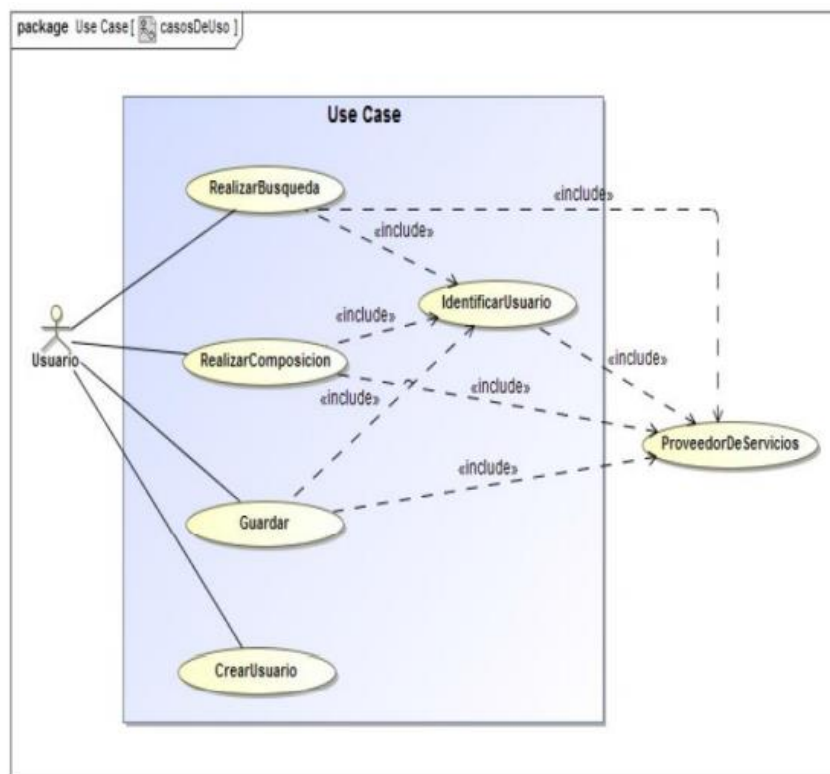
Por otro lado, UWE nos propone una extensión de UML que se divide en 4 fases.

- Análisis de requisitos. Su objetivo es encontrar los requisitos funcionales de la aplicación Web para representarlos como casos de uso. Da lugar a un diagrama de casos de uso.
- Diseño de Contenido. Su objetivo es construir un modelo conceptual del dominio de la aplicación considerando los requisitos reflejados en los casos de uso. Da como resultado un diagrama de clases de dominio.

- Diseño navegacional. Se obtienen el modelo de espacio de navegación y modelo de estructura de navegación, que muestra cómo navegar a través del espacio de navegación. Se obtienen diagramas de clases que representan estos modelos.
- Diseño de presentación. De este paso se obtienen una serie de vistas de interfaz de usuario que se presentan mediante diagramas de interacción UML.

### 2.7.2 Modelo de casos de uso

Los casos de uso se utilizan para visualizar las diferentes funcionalidades que la aplicación tiene que proporcionar, como son: crear a un nuevo usuario, identificar al usuario, realizar una búsqueda, realizar la composición de un nuevo objeto y guardar el objeto compuesto.

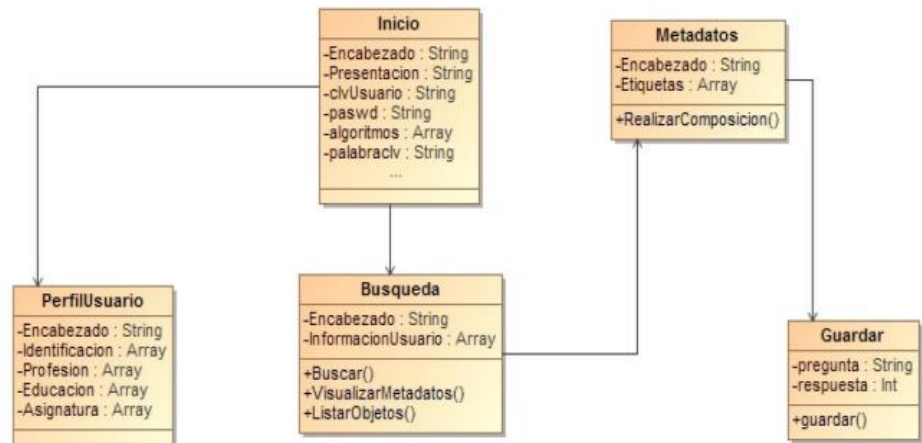


**Figura N° 1** Diagrama de Caso de uso

**Fuente:** (Mínguez, 2012)

### 2.7.3 Modelo de Contenido

El objetivo del modelo de contenido es proporcionar una especificación visual de la información en el dominio relevante para la aplicación Web. Este es un diagrama UML normal de clases, por ello se debe pensar en las clases que son necesarias para el caso de estudio presentado.



**Figura N° 2** *Modelo de Contenido*

**Fuente:** (Minguez,2012)

### 2.7.4 Modelo de usuario

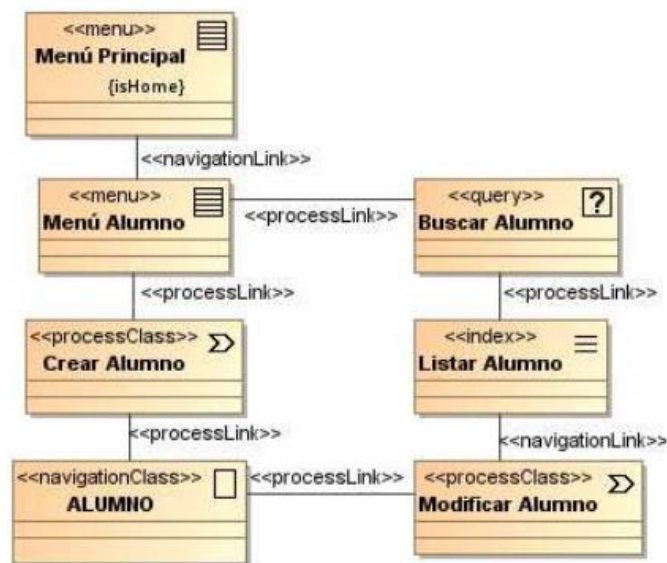
La diferencia entre el modelo de Contenido y este modelo no explicado anteriormente, es que el modelo de Contenido define el contenido de los datos almacenados por la aplicación, mientras que el modelo de Usuario tiene dos objetivos diferenciados:

- Contiene las clases que define qué información es almacenada en el contexto de una sesión. En este caso práctico una sesión está formada por el usuario actual y sus discos.

- Estas clases contenidas proveen de operaciones que puede ser usados en el proceso de negociación de procesos. (Mínguez, 2012).

### 2.7.5 Modelo de navegación

Este modelo indica como el sistema de páginas web del sitio está relacionado internamente es decir cómo se enlazan los elementos de navegación. Para ello se utilizan unidades de navegación llamadas “nodos” conectadas por enlaces de navegación estos nodos pueden ser mostrados en la misma página web, no tiene que estar en páginas diferentes. (Mínguez, 2012).



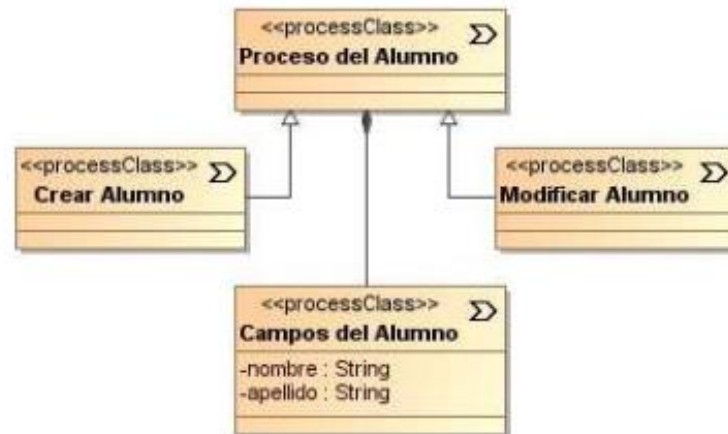
**Figura N° 3** Diagrama de Navegación

**Fuente:** (Mínguez,2012)

### 2.7.6 Modelo de procesos

Esta modelo especifica las acciones que realiza cada clase de proceso, en este modelo se incluye:

- **Modelo de Estructura de Procesos:** que define las relaciones entre las diferentes clases proceso. Un ejemplo de diagrama de clases de este modelo siguiendo el caso de la Agenda de contactos sería:



*Figura N° 4 Diagrama de estructuras de procesos*

Fuente: (Minguez,2012)

- **Modelo de Flujo de Procesos:** que especifica las actividades conectadas con cada proceso. Describe los comportamientos de una clase proceso. Lo que ocurre en detalle dentro de cada una. Por ejemplo, para la operación de borrado de contactos tenemos el siguiente diagrama.



*Figura N° 5 Diagrama de Flujo de procesos*

Fuente: (Mínguez, 2012)



### 2.7.7 Modelo de Presentación

El modelo de presentación ofrece una visión abstracta de la interfaz de usuario de una aplicación Web. Se basa en el modelo de navegación y en los aspectos concretos de la interfaz de usuario (IU). Describe la estructura básica de la IU, es decir, ¿qué elementos de interfaz de usuario (por ejemplo, texto, imágenes, enlaces, formularios) se utilizan para presentar los nodos de navegación?

Su ventaja es que es independiente de las técnicas actuales que se utilizan para implementar un sitio Web, lo que permite a las partes interesadas discutir la conveniencia de la presentación antes de que realmente se aplique.

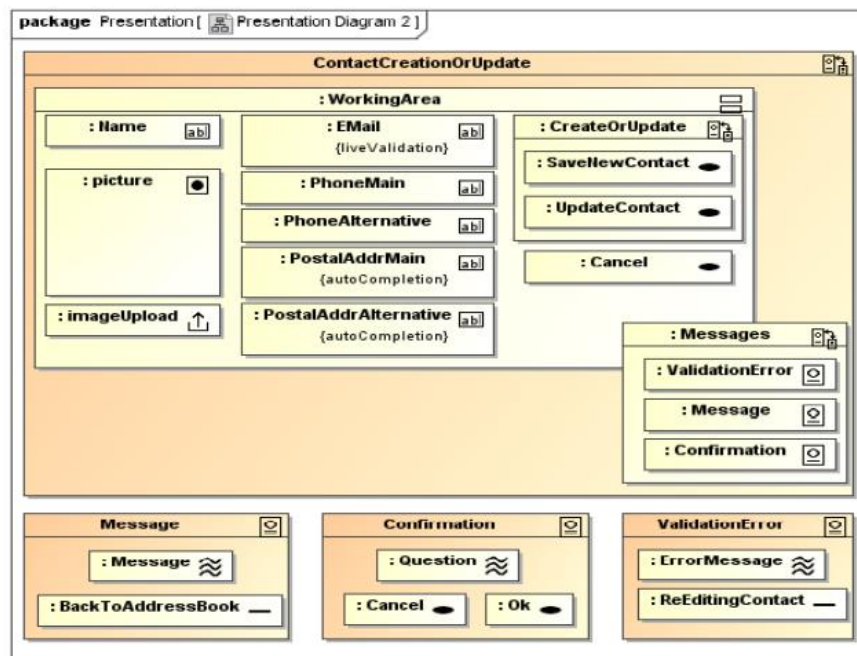


Figura N° 6 Diagrama de presentación

Fuente: (Mínguez, 2012)

## 2.8 UML (Unified Modeling Language)

El UML (Lenguaje Unificado de Modelado) es una herramienta que ayuda a capturar la idea de un sistema para comunicarla a quién esté implicado en el proceso de desarrollo esto se hace mediante un conjunto de símbolos y diagramas. Lo que permite este lenguaje es generar diseños para la comprensión fácil para que otros trabajen en el sistema. Es necesario entender por sistema como la combinación de hardware y software para solucionar algún tipo de negocio.

Presentamos algunas definiciones sobre lo que es UML.

El Lenguaje Unificado de Modelado (UML) es un lenguaje de modelado visual de propósito general que se utiliza para especificar, visualizar, construir y documentar los

artefactos de un sistema software. Captura decisiones y conocimiento sobre sistemas que deben ser construidos. Se usa para comprender, diseñar, ojear, configurar, mantener y controlar la información sobre tales sistemas.

Está pensado para ser utilizado con todos los métodos de desarrollo, etapas del ciclo de vida, dominios de aplicación y medios. El lenguaje de modelado pretende unificar la experiencia pasada sobre las técnicas de modelado e incorporar las mejores prácticas de software actuales en una aproximación estándar. UML incluye conceptos semánticos, notación y principios generales. Tiene partes estáticas, dinámicas, de entorno y organizativas. Está pensado para ser apoyado por herramientas de modelado visuales e interactivas que dispongan de generadores, tanto de código, como de informes. La especificación de UML no define un proceso estándar, pero está pensado para ser útil en un proceso de desarrollo

iterativo. Pretende dar apoyo a la mayoría de los procesos de desarrollo orientados a objetos existentes. (Rumbaugh, Jacobson, Booch, 2007).

UML (Unified Model Language) o Lenguaje de Modelado Unificado posee tres fuentes distintas en sus orígenes. Se trata de la “Técnica de Modelado de Objetos” (OMT), de James Rumbaugh, el “Método Booch”, de Grady Booch, y “Objetory”, de Ivar Jacobson (creador de OOSE - Object Oriented Software Engineer). Estos tres reconocidos expertos en 1994 realizaron la primera versión de UML para la empresa Rational Software Corporation (adquirida en el 2003 por IBM en 2.100 millones de dólares), haciendo cada uno aportes desde las perspectivas que venían desarrollando. (Anyer Martinez, 2011).

### **2.8.1 Modelo**

Un modelo es una representación en un cierto medio de algo en el mismo u otro medio. El modelo capta los aspectos importantes de lo que se está modelando, desde un cierto punto de vista, y simplifica u omite el resto. La ingeniería, la arquitectura y muchos otros campos creativos utilizan modelos.

Un modelo de un sistema software está construido en un lenguaje de modelado, como UML.

El modelo tiene, tanto semántica, como notación, y puede adoptar diversos formatos que incluyen el texto y los gráficos. Se pretende que el modelo sea más fácil de utilizar, para ciertos propósitos, que el sistema final. (Rumbaugh, Jacobson, Booch, 2007).

### **2.8.2 Vistas**

El modelaje de un sistema complejo es una tarea extensiva. Idealmente, el sistema completo sería descrito con un sólo gráfico que describa al sistema entero sin ambigüedades,

y que sea fácil de comunicar y entender. Sin embargo, esto es usualmente imposible. Un sólo gráfico no puede capturar toda la información necesaria para describir un sistema. Un sistema es descrito con una serie de aspectos funcionales (su estructura estática e interacciones dinámica) y no funcionales (requerimientos de tiempo, confiabilidad, despliegue, etc.). Es por ello que un sistema es descrito en una serie de vistas, donde cada vista representa una proyección de la descripción completa del sistema, mostrando un aspecto particular del sistema. Cada vista es descrita en una serie de diagramas que contienen información que enfatiza un aspecto particular del sistema. Hay un pequeño traslape, de manera que un diagrama puede actualmente ser parte de una o más vistas. Viendo al sistema de diferentes vistas, es posible concentrarse en un aspecto del sistema a la vez. Un diagrama en una vista particular debería ser suficientemente simple para ser fácilmente comunicado, y ser coherente con los otros diagramas y vistas, de manera que una imagen completa del sistema es descrita por las vistas en conjunto (a través de sus respectivos diagramas). Un diagrama contiene símbolos gráficos que representan los elementos de modelo del sistema. Las vistas son:

Vista de Casos de Uso: Es una vista que muestra la funcionalidad de un sistema como es percibida por los actores externos.

Vista Lógica: Es una vista que muestra como es diseñada la funcionalidad dentro del sistema, en términos de las estructuras estáticas del sistema y su comportamiento dinámico.

Vista de Componentes: Es una vista que muestra la organización de los componentes de código.

Vista de Procesos: Es una vista que muestra la concurrencia en el sistema, resolviendo problemas de comunicación y sincronización que estén presentes en un sistema concurrente.

Vista de Despliegue: Es una vista que muestra el despliegue de un sistema dentro de una arquitectura física con computadoras y dispositivos llamados nodos. (Mendoza Navarro Javier, 2015).

### 2.8.3 Diagramas

Los diagramas son los gráficos actuales que muestran los símbolos de los elementos del modelo arreglados para ilustrar una parte particular o aspecto del sistema. Un modelo del sistema típicamente tiene varios diagramas de cada tipo. Un diagrama es una parte de una vista específica; y cuando es dibujado, es usualmente adecuado para una vista. Algunos tipos de diagramas pueden ser parte de varias vistas, dependiendo de los contenidos del diagrama. (Enrique Hernández, 2014).

**Tabla 1**  
*Distribución de diagramas*

Análisis	Diseño
Diagrama de clases	Diagrama de clases
Diagrama de casos de uso	Diagrama de secuencia
Diagrama de estados	Diagrama de colaboración
Diagrama de actividades	Diagrama de componentes
	Diagrama de despliegue

**Fuente:** (Enrique Hernández, 2014).

## 2.9 Ingeniería de Software

Es el estudio de los principios y metodologías para el desarrollo y mantenimiento de sistemas de software. (Roger Pressman, 2007).

Ingeniería del software trata del establecimiento de los principios y métodos de la ingeniería a fin de obtener software de modo rentable que sea fiable y trabaje en máquinas reales. (Bauer, 1972).



**Figura N° 7** *Capas de la ingeniería de software*

**Fuente:** (Pressman, 2007).

### 2.9.1 Ciclo de vida

El término ciclo de vida del software describe el desarrollo de software, desde la fase inicial hasta la fase final. El propósito de este programa es definir las distintas fases intermedias que se requieren para validar el desarrollo de la aplicación, es decir, para garantizar que el software cumpla los requisitos para la aplicación y verificación de los procedimientos de desarrollo: se asegura de que los métodos utilizados son apropiados.

Estos programas se originan en el hecho de que es muy costoso rectificar los errores que se detectan tarde dentro de la fase de implementación. El ciclo de vida permite que los errores

se detecten lo antes posible y, por lo tanto, permite a los desarrolladores concentrarse en la calidad del software, en los plazos de implementación y en los costos asociados.

El ciclo de vida básico de un software consta de los siguientes procedimientos:

- Definición de objetivos: define la finalidad del proyecto y su papel en la estrategia global.
- Análisis de los requisitos y su viabilidad: recopila, examina y formula los requisitos del cliente y examina cualquier restricción que se pueda aplicar.
- Diseño general: requisitos generales de la arquitectura de la aplicación.
- Diseño en detalle: definición precisa de cada subconjunto de la aplicación.
- Programación (programación e implementación): implementación de un lenguaje de programación para crear las funciones definidas durante la etapa de diseño.
- Prueba de unidad: prueba individual de cada subconjunto de la aplicación para garantizar que se implementaron de acuerdo con las especificaciones.
- Integración: garantiza que los diferentes módulos se integren con la aplicación. Este es el propósito de la prueba de integración que está cuidadosamente documentada.
- Prueba beta (o validación): garantiza que el software cumple con las especificaciones originales.
- Documentación: sirve para documentar información necesaria para los usuarios del software y para desarrollos futuros.

- Implementación
- Mantenimiento: comprende todos los procedimientos correctivos (mantenimiento correctivo) y las actualizaciones secundarias del software (mantenimiento continuo).

El orden y la presencia de cada uno de estos procedimientos en el ciclo de vida de una aplicación dependen del tipo de modelo de ciclo de vida acordado entre el cliente y el equipo de desarrolladores. (Villagómez, 2017).

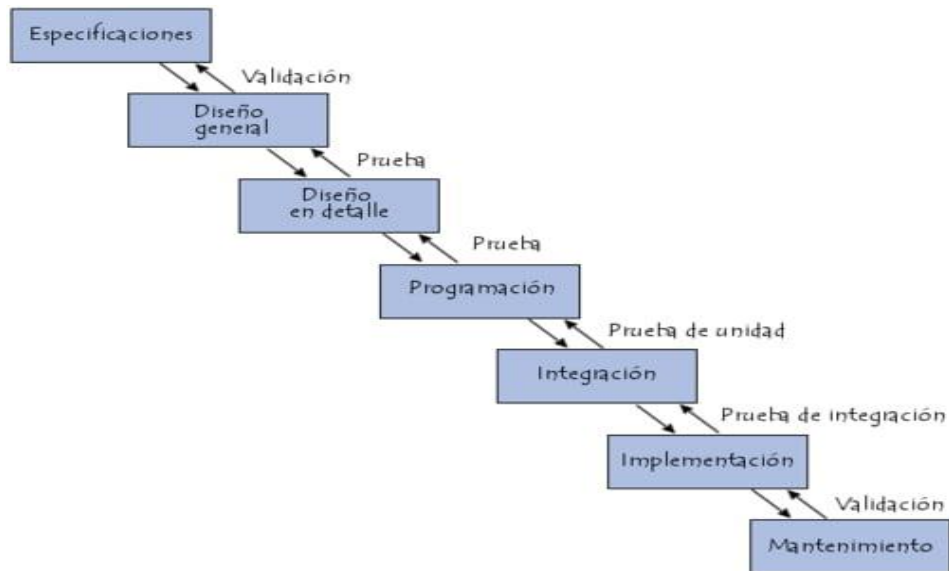
### **2.9.1.1 Modelo de ciclo de vida**

Para facilitar una metodología común entre el cliente y la compañía de *software*, los modelos de ciclo de vida se han actualizado para reflejar las etapas de desarrollo involucradas y la documentación requerida, de manera que cada etapa se valide antes de continuar con la siguiente.

#### **Modelo en cascada**

El modelo de ciclo de vida en cascada se comenzó a diseñar en 1966 y se terminó alrededor de 1970. Se define como una secuencia de fases donde al final de cada una de ellas se reúne la documentación para garantizar que cumple las especificaciones y los requisitos antes de pasar a la fase siguiente:



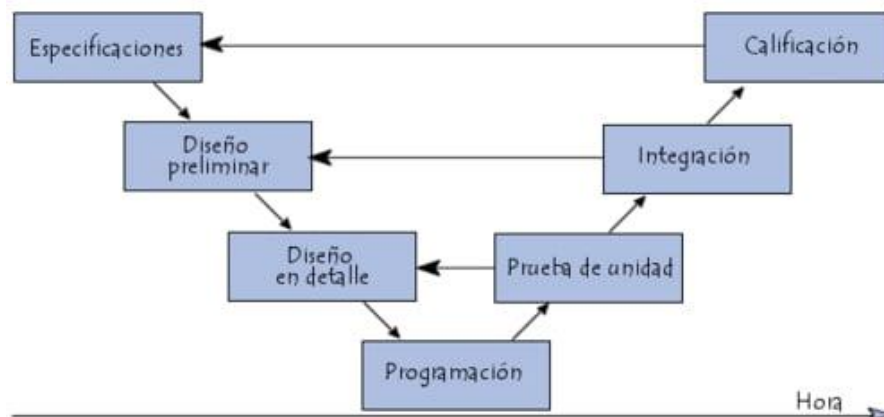


**Figura N° 8** Modelo en cascada

Fuente: (Villagómez, 2017).

### Modelo V

El modelo de ciclo de vida V proviene del principio que establece que los procedimientos utilizados para probar si la aplicación cumple las especificaciones ya deben haberse creado en la fase de diseño.



**Figura N° 9** Modelo V

Fuente: (Villagómez, 2017).

## **2.10 Herramientas**

### **2.10.1 Lenguaje PHP**

PHP es un lenguaje de código abierto muy popular, adecuado para desarrollo web y que puede ser incrustado en HTML. Es popular porque un gran número de páginas y portales web están creadas con PHP. Código abierto significa que es de uso libre y gratuito para todos los programadores que quieran usarlo. Incrustado en HTML significa que en un mismo archivo vamos a poder combinar código PHP con código HTML, siguiendo unas reglas. PHP se utiliza para generar páginas web dinámicas. Recordar que llamamos página estática a aquella cuyos contenidos permanecen siempre igual, mientras que llamamos páginas dinámicas a aquellas cuyo contenido no es el mismo siempre. Por ejemplo, los contenidos pueden cambiar en base a los cambios que haya en una base de datos, de búsquedas o aportaciones de los usuarios, etc. (E. Gonzales, 2006).

PHP lenguaje de script incrustado dentro del HTML (acrónimo recursivo de PHP: Hypertext Preprocessor), es un lenguaje de código abierto muy popular especialmente adecuado para el desarrollo web y que puede ser incrustado en HTML. PHP se caracteriza por ser un lenguaje gratuito y multiplataforma. Además de su posibilidad de acceso a muchos tipos de bases de datos, también es importante destacar su capacidad de crear páginas dinámicas, así como la posibilidad de separar el diseño del contenido de una web. (Rasmus Lerdorf, 1995).

#### **2.10.1.1 Características del PHP**

- Más Rápido que ASP.
- Es un lenguaje multiplataforma.

- Lenguaje más fácil y potente.
- Integración perfecta más de ocho servidores HTTP.
- Diseño Modular de fácil ampliación.
- Licencia Abierta.
- Acceso a 20 tipos de Bases de Datos.

### **2.10.2 Framework Laravel**

Laravel es un framework de código abierto para desarrollar aplicaciones y servicios web con PHP 5. En Laravel opcionalmente podemos usar el patrón de diseño MVC (ModeloVista-Controlador) tradicional, donde al igual que otros frameworks PHP, el controlador es programado como una clase.

Laravel incluye un sistema de mapeo de datos relacional llamado Eloquent ORM que facilita la creación de modelos. Este ORM se funda en patrón active record y su funcionamiento es muy sencillo. Laravel también incluye de paquete un sistema de procesamiento de plantillas llamado Blade. Este sistema de plantillas favorece un código mucho más limpio en las Vistas, además de incluir un sistema de Caché que lo hace mucho más rápido. El sistema Blade de Laravel, permite una sintaxis mucho más reducida en su escritura.

#### **2.10.2.1 ORM**

Un ORM (Object Relational Mapper) en PHP es un software que permite tratar la capa de persistencia de los datos, como simples accesos a métodos de una Clase u Objeto en PHP. La funcionalidad interna del ORM es mapear los objetos de PHP a las tablas en la base de datos,

para el caso en que la persistencia de los datos de la aplicación es proporcionada por una DB. La forma de interactuar con los datos en un patrón de diseño MVC, es mediante la creación de Modelos. Los Modelos son clases en PHP que encapsulan todo el trabajo con los datos de una aplicación. (Taylor Otwell, 2011).

### **2.10.2.2 Blade**

Blade nos provee de muchas ventajas (*asi como casi todo en Laravel*), además de modularizar nuestras vistas de una forma sorprendente, también nos permite usar estructuras de control y variables de PHP directamente en ellas, aunque esto ya era posible antes usando las etiquetas de php, por ejemplo:

```
<? php echo $var?>
```

```
<? = $var?>
```

Pero esto además de ser un poco incómodo de escribir deja nuestras vistas mucho más difíciles de entender y sucias por la mezcla de tanto código.

Entonces para el ejemplo anterior usamos el siguiente código

```
{{ $pastel }}
```

Esto es el equivalente a `<? = $pastel?>` y aunque con un ejemplo tan sencillo no se ve demasiada diferencia, a medida que se vaya realizando más los proyectos se notara la gran diferencia.

### **2.10.2.3 Composer**

Composer es un manejador de dependencias, no un gestor de paquetes. Pero es cierto que trata con paquetes y librerías, la instalación siempre es local para cualquier proyecto, las

librerías se instalan en un directorio por defecto (normalmente es /vendor). Composer es capaz de instalar las librerías que requiere tu proyecto con las versiones que necesiten.

Es un gestor de paquetes a nivel de aplicación para el PHP lenguaje de programación que proporciona un formato estándar para la gestión de dependencias de software PHP y requeridas bibliotecas.

Fue desarrollado por Nils Adermann y Jordi Boggiano, que siguen para administrar el proyecto. (Taylor Otwell, 2011).

#### **2.10.2.4 MVC**

El MVC o Modelo-Vista-Controlador es un patrón de arquitectura de software que, utilizando 3 componentes (Vistas, Models y Controladores) separa la lógica de la aplicación de la lógica de la vista en una aplicación. Es una arquitectura importante puesto que se utiliza tanto en componentes gráficos básicos hasta sistemas empresariales; la mayoría de los frameworks modernos utilizan MVC (o alguna adaptación del MVC) para la arquitectura, entre ellos podemos mencionar a Ruby on Rails, Django, AngularJS y muchos otros más. En este pequeño artículo intentamos introducirte a los conceptos del MVC.

##### **Modelo**

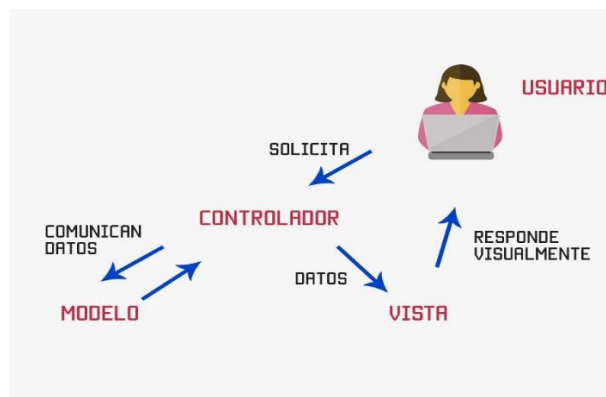
Se encarga de los datos, generalmente (pero no obligatoriamente) consultando la base de datos. Actualizaciones, consultas, búsquedas, etc. todo eso va aquí, en el modelo.

##### **Controlador**

Se encarga de controlar, recibe las órdenes del usuario y se encarga de solicitar los datos al modelo y de comunicárselos a la vista.

## Vistas

Son la representación visual de los datos, todo lo que tenga que ver con la interfaz gráfica va aquí. Ni el modelo ni el controlador se preocupan de cómo se verán los datos, esa responsabilidad es únicamente de la vista. (Uriel Hernandez,2015).



**Figura N° 10** *Modelo Vista Controlador*

**Fuente:** (Uriel Hernandez,2015)

### 2.10.3 Servidor web apache

Es un programa especialmente diseñado para transferir datos de hipertexto, es decir, páginas web con todos sus elementos (textos, widgets, banners, etc). Estos servidores web utilizan el protocolo http. Apache Es un servidor web HTTP de código abierto para la creación de páginas y servicios web. Es un servidor multiplataforma, una de las ventajas más grandes de Apache, es que es un servidor web multiplataforma, gratuito, muy robusto y que destaca por su seguridad y rendimiento. (Robert McCool. 1995).

Principales características de Apache. Entre las principales características de Apache, se encuentran las siguientes:

- Soporte de seguridad SSL y TLS.

- Puede realizar autenticación de datos utilizando SGDB.
- Puede dar soporte a diferentes lenguajes, como Perl, PHP, Python y tcl.

### **Ventajas**

- Instalación/Configuración. Software de código abierto.
- Coste. El servidor web Apache es completamente gratuito.
- Funcional y Soporte. Alta aceptación en la red y muy popular, esto hace que muchos programadores de todo el mundo contribuyen constantemente con mejoras, que están disponibles para cualquier persona que use el servidor web y que Apache se actualice constantemente.
- Multi-plataforma. Se puede instalar en muchos sistemas operativos, es compatible con Windows, Linux y MacOS.
- Rendimiento. Capacidad de manejar más de un millón de visitas/día.
- Soporte de seguridad SSL y TLS.

#### **2.10.4 Base de datos**

Es un sistema que almacena datos que están relacionados. Una base de datos (BD) es una entidad en la cual se pueden almacenar datos de manera estructurada, con la menor redundancia posible. Se define como un conjunto de datos que se encuentran organizados y relacionados entre sí, con el fin de satisfacer tratamientos de información implicados en las actividades de una empresa. Una base de datos proporciona a los usuarios el acceso a datos, que pueden visualizar, ingresar o actualizar, en concordancia con los derechos de acceso que

se les hayan otorgado. Se convierte más útil a medida que la cantidad de datos almacenados crece. Por este motivo a las BD's también se les puede identificar como un Almacén de Datos. (J. López, 2011).

#### **2.10.4.1 Generalidades de las bases de datos**

**Base datos relacional:** Es aquella que está compuesta por un conjunto de datos almacenados y relacionados entre sí, permitiendo manipular los datos de una empresa. La función principal de una base de datos es mantener la integridad y seguridad de los datos ante cualquier incidente, además es considerada como un sistema de datos integrados los cuales pueden ser manipulados directamente por una serie de aplicaciones.

**Sistema gestor de bases de datos (SGBD):** También denominado por sus siglas en inglés como Data Base Management System (DBMS), es una herramienta que sirve como interfaz entre la base de datos, el usuario y las aplicaciones. Además, permite definir, construir, almacenar y manejar una base de datos, permitiendo el control de los tipos de datos y restricciones sobre la información. (Zea M, Honores J, Rivas W,2015).

#### **2.10.5 MariaDB**

MariaDB es un sistema gestor de bases de datos (SGBD), es decir, un conjunto de programas que permiten modificar, almacenar, y extraer información de una base de datos. Disponiendo de otro tipo de funcionalidades como la administración de usuarios, y recuperación de la información si el sistema se corrompe, entre otras.

MariaDB surge a raíz de la compra, de la compañía desarrolladora de otro (SGBD) llamado MySQL, por la empresa Sun Microsystems. El desarrollador original, decide tomar el código fuente original de MySQL y genera un derivado con mejoras y cambios a los que



llama MariaDB. Permitiendo así la existencia de una versión de este producto con licencia GPL (General Public License).

Como hemos comentado anteriormente, MariaDB se trata de una “sustituto” de MySQL que corre bajo una licencia GPL. Por lo tanto, incorpora todas las mejoras de la versión de *MySQL* en la que se encuentre y además incluye diferentes mejoras:

Incorpora nuevos motores de almacenamiento mucho más eficientes que son **Aria** y **XtraDB**, los cuales han sido desarrollados para ser los sustitutos de **MyISAM** e **InnoDB** respectivamente. Estos permiten ejecutar consultas más complejas y almacenarlas en caché y no en disco duro.

Aparte de incluir los sustitutos de MyISAM e InnoDB también incorpora nuevos motores de almacenamiento:

**FederatedX**: Reemplaza a Federated que incorpora mysql.

**OQGRAPH**: Nos permite que nuestro sistema de base de datos soporte el uso de jerarquías de estructuras y graphs complejos.

**SphinxSE**: Nos permite usar searchd bajo Sphinx para permitir búsquedas de texto.

**Cassandra Storage Engine**: Nos permite acceder a un clúster de datos. Cabe destacar que por defecto no viene instalado y deberá activarse por separado.

También incorpora mejoras de rendimiento y versiones de seguridad más rápidas y más transparentes.

Es de código libre y cuenta con el soporte de la comunidad de desarrolladores, pero también cuenta con el soporte de Oracle. (G. Auladell, 2016).

## 2.11 Pruebas

### 2.11.1 Pruebas de caja negra o funcionales

También conocidas como Pruebas de Comportamiento, estas pruebas se basan en la especificación del programa o componente a ser probado para elaborar los casos de prueba. El componente se ve como una “Caja Negra” cuyo comportamiento sólo puede ser determinado estudiando sus entradas y las salidas obtenidas a partir de ellas. No obstante, como el estudio de todas las posibles entradas y salidas de un programa sería impracticable se selecciona un conjunto de ellas sobre las que se realizan las pruebas. Para seleccionar el conjunto de entradas y salidas sobre las que trabajar, hay que tener en cuenta que en todo programa existe un conjunto de entradas que causan un comportamiento erróneo en nuestro sistema, y como consecuencia producen una serie de salidas que revelan la presencia de defectos. Entonces, dado que la prueba exhaustiva es imposible, el objetivo final es pues, encontrar una serie de datos de entrada cuya probabilidad de pertenecer al conjunto de entradas que causan dicho comportamiento erróneo sea lo más alto posible.

Al igual que ocurría con las técnicas de Caja Blanca, para confeccionar los casos de prueba de Caja Negra existen distintos criterios. Algunos de ellos son:

- Particiones de Equivalencia.
- Análisis de Valores Límite.
- Métodos Basados en Grafos.
- Pruebas de Comparación.
- Análisis Causa-Efecto.

De ellas, las técnicas que estudiaremos son las dos primeras, esto es, Particiones de Equivalencia y Análisis de Valores Límite. (J. Sánchez, 2015).

## **2.12 Seguridad en aplicaciones Web**

La seguridad de la aplicación web es un tema crítico y complejo para desarrolladores web. Un sistema seguro requiere una cuidadosa planificación, y los administradores y desarrolladores de sitios web deben tener un conocimiento previo de ello, ya que requiere realizar un estudio y tener entendimiento sobre los puntos vulnerables en cada aplicación web.

Los puntos más importantes se pueden clasificar en:

### **Seguridad en el servidor**

- Servidor Web.
- Servidor de Base de Datos.
- Lenguaje de Programación.

### **Seguridad en Aplicaciones Web**

- Control de Acceso
- Validación de Datos de Entrada
- Pruebas de Código

### **Seguridad en la Comunicación**

- Protocolos de seguridad

- Certificados de Comunicación

### **Copias de Seguridad**

- Backup de la Base de Datos
- Backup del control de Acceso

Las pruebas de seguridad están diseñadas para probar la vulnerabilidad en el ambiente del lado del cliente, las comunicaciones de red que ocurren mientras los datos pasan del cliente al servidor pueden recibir ataques y es labor de quien prueba la seguridad descubrir las debilidades que pueden explotar quienes tengan la intención de hacerlo. (Pressman, 2007).

**Cortafuegos:** Mecanismo de filtrado combinación de hardware y software para examinar cada paquete de información entrante para garantizar que no sea un dato sospechoso.

**Uso de Autenticación:** Una contraseña o clave es una forma de autenticación que utiliza informa secreta para controlar el acceso hacia recursos. A aquellos que desean acceder a la información se les solicita una clave; si conocen o no conocen la contraseña se concede o se niega el acceso a la información según sea el caso.

**Modelo de Criptografía:** La encriptación<sup>4</sup> está asociada con la transformación de un mensaje inteligente a una forma no inteligente con la ayuda de una clave secreta antes de que sea colocada en un medio inseguro.

**Algoritmo básico del MD5:** El mensaje se divide en bloques de 512 bits, añadiendo bits si es necesario al último bloque (de 1 a 448, por una parte, más 64 fijos de un valor  $k \bmod 264$ ). Con cada bloque de 512 se realiza 64 iteraciones. En cada iteración la función MD5 se

alimenta con el bloque correspondiente y con el resultado de la iteración anterior, un vector de cuatro elementos de 32 bits cada uno (con valores predeterminados en la primera iteración). La salida de cada iteración se convierte en el nuevo vector (128 bits). Algoritmo de encriptación Bcrypt: es una cadena de un solo sentido. Utiliza el algoritmo estándar basado en DES. Tiene como parámetro opcional una variable llamada salt, que es la que permite generar contraseñas muy fuertes.

### **2.13 Métricas de calidad**

Las Métricas de Calidad proporcionan una indicación de cómo se ajusta el software, a los requerimientos implícitos y explícitos del cliente. La calidad del software comprende distintos aspectos como estética (que sea agradable a la vista), funcionalidad (que sea fácil de usar), eficiencia (que ejecute con rapidez y precisión los procesos), etc. La obtención de un software con calidad implica la utilización de metodologías o procedimientos estándares para el análisis, diseño, programación y prueba del software, con el propósito de lograr una mayor confiabilidad, mantenibilidad, facilidad de prueba y uso. (R. Pressman, 2007).

#### **2.13.1 Métricas de calidad establecidas por la norma iso 9126**

La norma - ISO 9126 es un estándar internacional para la evaluación del Software, fue originalmente desarrollado en 1991 para proporcionar un esquema para la evaluación de calidad del software.



**Figura N° 11** Características de la Norma ISO 9126

Fuente: (R Pressman, 2007)

Las definiciones se dan para cada característica y subcaracterística de calidad del software que influye en la calidad. Para cada característica y subcaracterística, la capacidad del software es determinada por un conjunto de atributos internos que pueden ser medidos. Las características y subcaracterísticas se pueden medir externamente por la capacidad del sistema que contiene el software.

#### **a). Funcionalidad**

La funcionalidad es la capacidad del software de cumplir y proveer las funciones para satisfacer las necesidades explícitas e implícitas cuando es utilizado en condiciones específicas.

A continuación, se muestra la característica de Funcionalidad y las sub-características que cubre:

#### **Característica de funcionalidad**

La funcionalidad se divide en 5 criterios:

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

#### **b). Confiabilidad**

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

La confiabilidad se divide en 4 criterios:

- **Madurez:** La capacidad que tiene el software para evitar fallas cuando encuentra errores. Ejemplo, la forma como el software advierte al usuario cuando realiza operaciones en la unidad de diskett vacía, o cuando no encuentra espacio suficiente el disco duro donde esta almacenando los datos.

- **Tolerancia a errores:** La capacidad que tiene el software para mantener un nivel de funcionamiento en caso de errores.
- **Recuperabilidad:** La capacidad que tiene el software para restablecer su funcionamiento adecuado y recuperar los datos afectados en el caso de una falla.
- **Conformidad de la fiabilidad:** La capacidad del software de cumplir a los estándares o normas relacionadas a la fiabilidad.

### c). Usabilidad

La usabilidad es la capacidad del software de ser entendido, aprendido, y usado en forma fácil y atractiva. Algunos criterios de funcionalidad, fiabilidad y eficiencia afectan la usabilidad, pero para los propósitos de la ISO/IEC 9126 ellos no clasifican como usabilidad.

La usabilidad está determinada

por los usuarios finales y los usuarios indirectos del software, dirigidos a todos los ambientes, a la preparación del uso y el resultado obtenido.

### Característica de Usabilidad

La usabilidad se divide en 5 criterios:

- **Entendimiento:** La capacidad que tiene el software para permitir al usuario entender si es adecuado, y de una manera fácil como ser utilizado para las tareas y las condiciones particulares de la aplicación. En este criterio se debe tener en cuenta la documentación y de las ayudas que el software entrega.
- **Aprendizaje:** La forma como el software permite al usuario aprender su uso. También es importante considerar la documentación.



- **Operabilidad:** La manera como el software permite al usuario operarlo y controlarlo.
- **Atracción:** La presentación del software debe ser atractiva al usuario. Esto se refiere a las cualidades del software para hacer más agradable al usuario, ejemplo, el diseño gráfico.
- **Conformidad de uso:** La capacidad del software de cumplir los estándares o normas relacionadas a su usabilidad.

#### **d). Eficiencia**

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

#### **Característica de Eficiencia**

La eficiencia se divide en 3 criterios:

- **Comportamiento de tiempos:** Los tiempos adecuados de respuesta y procesamiento, el rendimiento cuando realiza su función en condiciones específicas. Ejemplo, ejecutar el procedimiento más complejo del software y esperar su tiempo de respuesta, realizar la misma función, pero con más cantidad de registros.
- **Utilización de recursos:** La capacidad del software para utilizar cantidades y tipos adecuados de recursos cuando este funciona bajo requerimientos o

condiciones establecidas. Ejemplo, los recursos humanos, el hardware, dispositivos externos.

- **Conformidad de eficiencia:** La capacidad que tiene el software para cumplir con los estándares o convenciones relacionados a la eficiencia.

#### **e). Capacidad de Mantenimiento**

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

#### **Característica de Mantenimiento**

El mantenimiento se divide en 5 criterios:

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

- **Conformidad de facilidad de mantenimiento:** La capacidad que tiene el software para cumplir con los estándares de facilidad de mantenimiento.

#### f). Portabilidad

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

#### Característica de portabilidad

La usabilidad se divide en 5 criterios:

- **Adaptabilidad:** Es como el software se adapta a diferentes entornos especificados (hardware o sistemas operativos) sin que implique reacciones negativas ante el cambio. Incluye la escalabilidad de capacidad interna (Ejemplo: Campos en pantalla, tablas, volúmenes de transacciones, formatos de reporte, etc.).
- **Facilidad de instalación:** La facilidad del software para ser instalado en un entorno específico o por el usuario final.
- **Coexistencia:** La capacidad que tiene el software para coexistir con otro o varios softwares, la forma de compartir recursos comunes con otro software o dispositivo.
- **Reemplazabilidad:** La capacidad que tiene el software para ser reemplazado por otro software del mismo tipo, y para el mismo objetivo. Ejemplo, la reemplazabilidad de una nueva versión es importante para el usuario, la propiedad de poder migrar los datos a otro software de diferente proveedor.

- **Conformidad de portabilidad:** La capacidad que tiene el software para cumplir con los estándares relacionados a la portabilidad.

#### **g). Calidad en Uso**

Calidad en uso es la calidad del software que el usuario final refleja, la forma como el usuario final logra realizar los procesos con satisfacción, eficiencia y exactitud. La calidad en uso debe asegurar la prueba o revisión de todas las opciones que el usuario trabaja diariamente y los procesos que realiza esporádicamente<sup>5</sup> relacionados con el mismo software.

#### **Característica Calidad de uso**

La calidad de uso se divide en 4 criterios:

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

- **Satisfacción:** La satisfacción es la respuesta del usuario a la interacción con el software, e incluye las actitudes hacia el uso del mismo. A continuación, se describe un cuadro donde podemos resumir las características y cada uno de sus atributos, este cuadro le ayudara a visualizar el proceso de evaluación. (R. Pressman, 2007).

### 2.13.2 Punto de Función

La técnica de medición del tamaño en punto-función consiste en asignar una cantidad de "puntos" a una aplicación informática según la complejidad de los datos que maneja y de los procesos que realiza sobre ellos, siempre tratando de considerarlo desde el punto de vista del usuario.

Los valores de los dominios de información se definen de la forma siguiente:

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

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

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

**Número de archivos.** Se cuenta cada archivo maestro lógico (esto es, un grupo lógico de datos que puede ser una parte de una gran base de datos o un archivo independiente).

**Número de interfaces externas.** Se cuentan todas las interfaces legibles por la máquina (por ejemplo: archivos de datos de cinta o disco) que se utilizan para transmitir información a otro sistema.

Una vez que se han recopilado los datos anteriores, a la cuenta se asocia un valor de complejidad. Las organizaciones que utilizan métodos de puntos de función desarrollan criterios para determinar si una entrada en particular es simple, media o compleja. No obstante, la determinación de la complejidad es algo subjetiva.

Parámetros de medición	Factor de ponderación				=	Cuenta
	Cuenta	Simple	Medio	Complejo		
Número de entradas de usuario	<input type="text"/>	× 3	4	6	=	<input type="text"/>
Número de salidas de usuario	<input type="text"/>	× 4	5	7	=	<input type="text"/>
Número de peticiones de usuario	<input type="text"/>	× 3	4	6	=	<input type="text"/>
Número de archivos	<input type="text"/>	× 7	10	15	=	<input type="text"/>
Número de interfaces externas	<input type="text"/>	× 5	7	10	=	<input type="text"/>
Cuenta total	→					<input type="text"/>

**Figura N° 12** Factor de ponderación - Punto Función

**Fuente:** (R. Pressman, 2007).

Para calcular puntos de función, se utiliza la relación siguiente:

$$FP = \text{cuenta-total} \times [0,65 + 0,01 \times \Sigma (Fi)] \quad (1)$$

Donde **cuenta-total** es la suma de todas las entradas obtenidas de la Figura N° 12. Fi donde i puede ser de uno hasta 14 los valores de ajuste de complejidad basados en las respuestas a las siguientes preguntas:

1. ¿Requiere el sistema copias de seguridad y de recuperación fiables?
2. ¿Se requiere comunicación de datos?
3. ¿Existen funciones de procesamiento distribuido?
4. ¿Es crítico el rendimiento?
5. ¿Se ejecutaría el sistema en un entorno operativo existente y fuertemente utilizado?
6. ¿Requiere el sistema entrada de datos interactiva?
7. ¿Requiere la entrada de datos interactiva que las transacciones de entrada se lleven a cabo sobre múltiples pantallas u operaciones?
8. ¿Se actualizan los archivos maestros de forma interactiva?
9. ¿Son complejas las entradas, las salidas, los archivos o las peticiones?
10. ¿Es complejo el procesamiento interno?
11. ¿Se ha diseñado el código para ser reutilizable?
12. ¿Están incluidas en el diseño la conversión y la instalación?
13. ¿Se ha diseñado el sistema para soportar múltiples instalaciones en diferentes organizaciones?
14. ¿Se ha diseñado la aplicación para facilitar los cambios y para ser fácilmente utilizada por el usuario?

Cada una de las preguntas anteriores es respondida usando una escala con rangos desde 0 (no importante o aplicable) hasta 5 (absolutamente esencial).

0	1	2	3	4	5
Sin influencia	Incidental	Moderado	Medio	Significativo	Esencial

**Figura N° 13** *Escala de Valores*

Fuente: (R. Pressman, 2007)

Los valores constantes de la ecuación (1) y los factores de peso que se aplican a las cuentas de los dominios de información se determinan empíricamente<sup>7</sup>. Una vez que se han calculado los puntos de función, se utilizan de forma análoga a las líneas de código como forma de normalizar las medidas de, productividad, calidad y otros atributos del software.

Una vez calculado los puntos de función se usan de forma análoga a las líneas de código como medida de la productividad, calidad y otros productos del software. (R. Pressman, 2007).

$$\text{Productividad} = \text{PF} / \text{persona-mes}$$

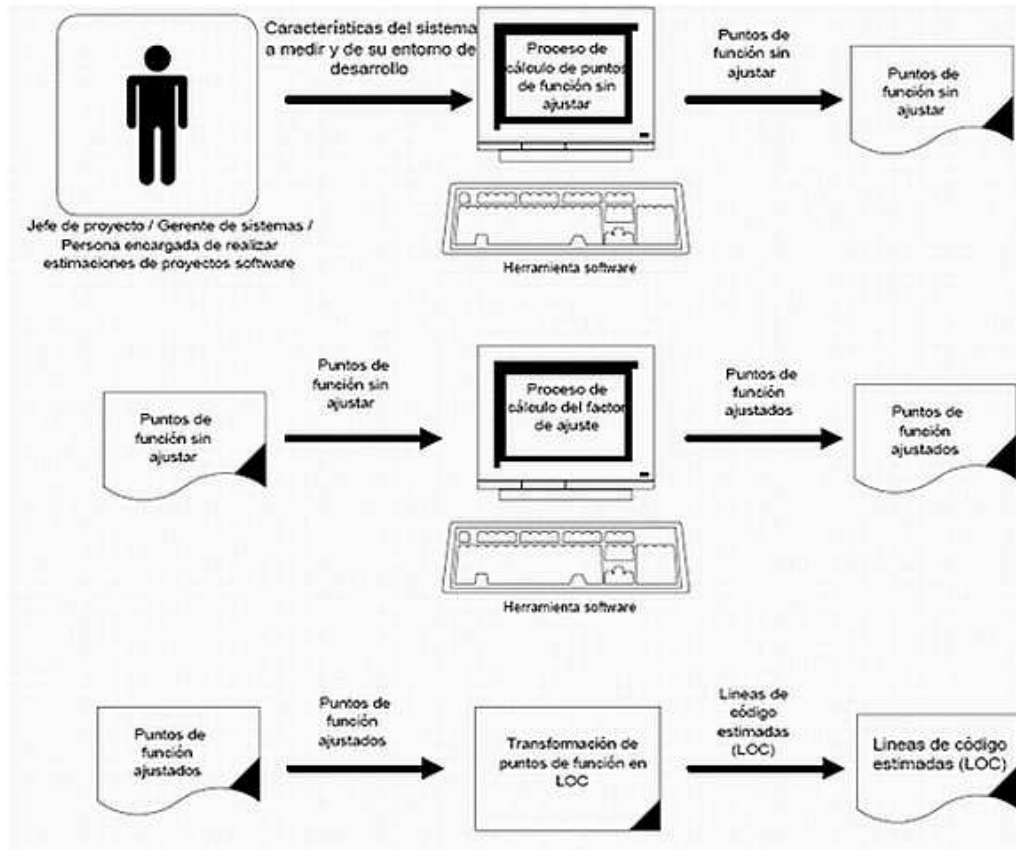
$$\text{Calidad} = \text{Errores} / \text{PF}$$

$$\text{Costo} = \text{Dólares} / \text{PF}$$

$$\text{Documentación} = \text{Pags. Doc} / \text{PF}$$

El proceso de cálculo y obtención de puntos de función y su posible transformación a líneas de código se esquematiza en la siguiente figura:





**Figura N° 14** *Proceso de cálculo de puntos de función y transformación a líneas de código*

Fuente: (R. Pressman – 2007)

## 2.14 Costos

### 2.14.1 Modelo para estimación de costos Cocomo II

#### Definición del modelo

Los objetivos principales que se tuvieron en cuenta para construir el modelo COCOMO

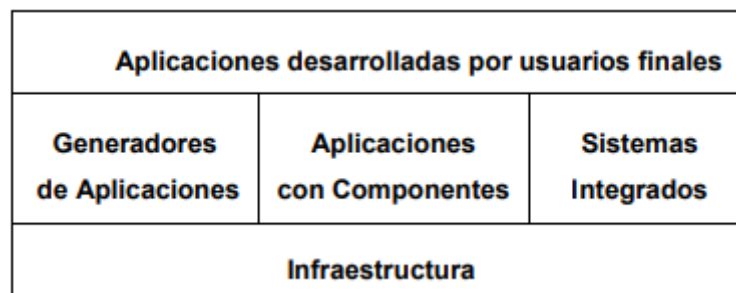
II fueron:

- Desarrollar un modelo de estimación de costo y cronograma de proyectos de software que se adaptara tanto a las prácticas de desarrollo de la década del 90 como a las futuras.

- Construir una base de datos de proyectos de software que permitiera la calibración continua del modelo, y así incrementar la precisión en la estimación.
- Implementar una herramienta de software que soportara el modelo.
- Proveer un marco analítico cuantitativo y un conjunto de herramientas y técnicas que evaluaran el impacto de las mejoras tecnológicas de software sobre los costos y tiempos en las diferentes etapas del ciclo de vida de desarrollo.

COCOMO II está compuesto por tres modelos denominados: Composición de Aplicación, Diseño Temprano y Post-Arquitectura.

Éstos surgen en respuesta a la diversidad del mercado actual y futuro de desarrollo de software. Esta diversidad podría representarse con el siguiente esquema (Figura 15).



**Figura N° 15** *Distribución del Mercado de Software Actual y Futuro.*

Fuente: (A. Gómez, María del C.López, 2010)

- **Aplicaciones desarrolladas por Usuarios Finales:** En este sector se encuentran las aplicaciones de procesamiento de información generadas directamente por usuarios finales, mediante la utilización de generadores de aplicaciones tales como planillas de cálculo, sistemas de consultas, etc. Estas aplicaciones surgen debido

al uso masivo de estas herramientas, conjuntamente con la presión actual para obtener soluciones rápidas y flexibles.

- **Generadores de Aplicaciones:** En este sector operan firmas como Lotus, Microsoft, Novell, Borland con el objetivo de crear módulos pre-empaquetados que serán usados por usuarios finales y programadores.
- **Aplicaciones con Componentes:** Sector en el que se encuentran aquellas aplicaciones que son específicas para ser resueltas por soluciones pre-empaquetadas, pero son lo suficientemente simples para ser construidas a partir de componentes interoperables. Componentes típicas son constructores de interfases gráficas, administradores de bases de datos, buscadores inteligentes de datos, componentes de dominio-específico (medicina, finanzas, procesos industriales, etc.). Estas aplicaciones son generadas por un equipo reducido de personas, en pocas semanas o meses.
- **Sistemas Integrados:** Sistemas de gran escala, con un alto grado de integración entre sus componentes, sin antecedentes en el mercado que se puedan tomar como base. Porciones de estos sistemas pueden ser desarrolladas a través de la composición de aplicaciones. Entre las empresas que desarrollan software representativo de este sector, se encuentran grandes firmas que desarrollan software de telecomunicaciones, sistemas de información corporativos, sistemas de control de fabricación, etc.
- **Infraestructura:** Área que comprende el desarrollo de sistemas operativos, protocolos de redes, sistemas administradores de bases de datos, etc.

Incrementalmente este sector direccionará sus soluciones, hacia problemas genéricos de procesamiento distribuido y procesamiento de transacciones, a soluciones middleware. Firmas representativas son Microsoft, Oracle, SyBase, Novell y NeXT.

### Modelo Básico

Se basa en el tamaño expresado en miles de líneas de código (KLOC). Las fórmulas del modelo son las siguientes:

$$E = a \cdot (KLOC)^b$$

$$T_{dev} = c \cdot (E)^d$$

$$P = \frac{E}{T_{dev}}$$

Donde:

- **E.** Es el esfuerzo medido en personas/mes
- **Tdev.** Es el tiempo estimado en meses
- **P.** Es el número de personas requerido para el proyecto
- **a, b, c, d.** Son constantes con valores definidos según cada modo y cada modelo
- **KLOC.** Son el número de miles de líneas de código fuente que tiene el software que estamos intentado estimar

Los valores de las constantes son los siguientes:

**Tabla 2**  
*Coefficientes del modelo COCOMO*

<b>Modo</b>	<b>A</b>	<b>B</b>	<b>C</b>	<b>D</b>
Orgánico	2.40	1.05	2.50	0.38
Semilibre	3.00	1.12	2.50	0.35
Rígido	3.60	1.20	2.50	0.32

Fuente: (Fenton, 1991).

### **Modelo Intermedio**

El modelo COCOMO intermedio calcula el esfuerzo del desarrollo de software en función del tamaño del programa y de un conjunto de “conductores de costo” que incluyen la evaluación subjetiva del producto, del hardware, del personal y de los atributos del proyecto.

Los modelos COCOMO están establecidos para tres prototipos de proyectos de software que empleando la terminología de Boehm son:

- **Modo orgánico:** aquellos proyectos de software que son respectivamente pequeños y sencillos en donde trabajan pequeños equipos que poseen buena experiencia en la aplicación, sobre un conjunto de requisitos poco rígidos.
- **Modo semiacoplado:** son los proyectos de software intermedios hablando de tamaño y complejidad, en donde los equipos tienen diversos niveles de experiencia, y además deben satisfacer requerimientos poco o medio rígidos.
- **Modo empotrado:** son proyectos de software que deben ser desarrollados en un conjunto de hardware, software y restricciones operativas muy restringido.

La ecuación del modelo COCOMO intermedio toma la forma:

$$E = a \cdot (KLOC)^b \cdot m(X)$$

$$T_{dev} = c \cdot (E)^d$$

$$P = \frac{E}{T_{dev}}$$

Donde:

- **E**. Es el esfuerzo medido en personas/mes
- **Tdev**. Es el tiempo estimado en meses
- **P**. Es el número de personas requerido para el proyecto
- **a, b, c, d**. Son constantes con valores definidos según cada modo y cada modelo
- **m(X)**. Es un multiplicador que se calcula con los 15 atributos de coste que se indican más abajo
- **KLOC**. Son el número de miles de líneas de código fuente que tiene el software que estamos intentado estimar.

**Tabla 3**  
*Coefficientes del modelo COCOMO*

MODO	A	B	C	D
Orgánico	3.20	1.05	2.50	0.38
Semi-libre	3.00	1.12	2.50	0.35
Rígido	2.80	1.20	2.50	0.32

Fuente: (Fenton, 1991).

Se puede observar que los exponentes son los mismos que los del modelo básico, confirmando el papel que representa el tamaño, mientras que los coeficientes de los modos

orgánico y rígido han cambiado, para mantener el equilibrio alrededor del Semi-libre con respecto al efecto multiplicador de los atributos de costes.

**a). Ecuaciones nominadas de costes.**

Para ecuaciones nominales de coste para el modelo intermedio son:

Modo Orgánico  $Kn = 3.2$  Sk 1.05

Modo semiencajado  $Kn = 3.0$  Sk 1.12

Modo empotrado  $Kn = 3.8$  Sk 1.20

**b). Atributo de Costes**

Cada atributo se cuantifica para un entorno de proyecto. La escala es muy baja – nominal – alto – muy alto – extremadamente alto. Dependiendo de la calificación de cada atributo se asigna un valor para usar de multiplicar en la formula el significado de los atributos es el siguiente, según el tipo de:

**Software**

- RELY garantía de funcionamiento requerida al software. Indica las posibles consecuencias para el usuario en el caso que existan defectos en el producto. Va desde la sola convivencia de corregir un fallo (muy bajo) hasta la posible pérdida de vidas humanas (extremadamente alto, software de alta criticidad).
- DATA tamaño de una base de datos en relación con el tamaño del programa. El valor del modificador se define por la relación  $D/K$  donde D corresponde al

tamaño de la base de datos en bytes y  $K$  es el tamaño del programa en cantidad de líneas de código.

- CPLK representa la complejidad del producto.

### **Hardware**

- IIME: limitaciones en el porcentaje del uso de la CPU.
- STOR: limitaciones en el porcentaje del uso de la memoria.
- VIRT: volatilidad de la máquina virtual.
- TURN: tiempo de repuesta requerido.

### **Personal**

- ACAP: calificación de los analistas.
- AEXP: experiencia del personal en aplicaciones similares.
- PCAP: calificación de los programadores.
- VEXP: experiencia del personal en la máquina virtual.
- LEXP: experiencia en el lenguaje de programación a usar.

### **Proyecto**

- MODP: uso de prácticas modernas de programación.
- TOOL: uso de herramientas de desarrollo de software.
- SCED: limitaciones en el cumplimiento de la planificación.



El valor de cada atributo de acuerdo a su calificación, se muestra en la Tabla

**Tabla 4**  
*Atributo de Coste.*

Atributos	Valores					
	Muy bajo	Bajo	Normal	Alto	Muy Alto	Extra Alto
<b>Atributos de Software</b>						
Facilidad	0.75	0.88	1.00	1.15	1.49	
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>						
restricción de tiempo de ejecución			1.00	1.11	1.30	1.66
Restricción de memoria virtual.			1.00	1.06	1.21	1.56
Volatilidad de la máquina virtual.		0.87	1.00	1.15	1.15	
Tiempo de respuesta.		0.87	1.00	1.07	1.07	
<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	
Experiencias en la máquina virtual.	1.21	1.10	1.00	0.90		
Experiencia en el lenguaje.	1.14	1.17	1.00	0.95		
<b>Atributo 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.	1.24	1.08	1.00	1.04	1.10	

**Fuente:** (Aguilar, Cruz, 2011).

**CAPITULO III**

**MARCO APLICATIVO**

## CAPITULO III

### MARCO APLICATIVO

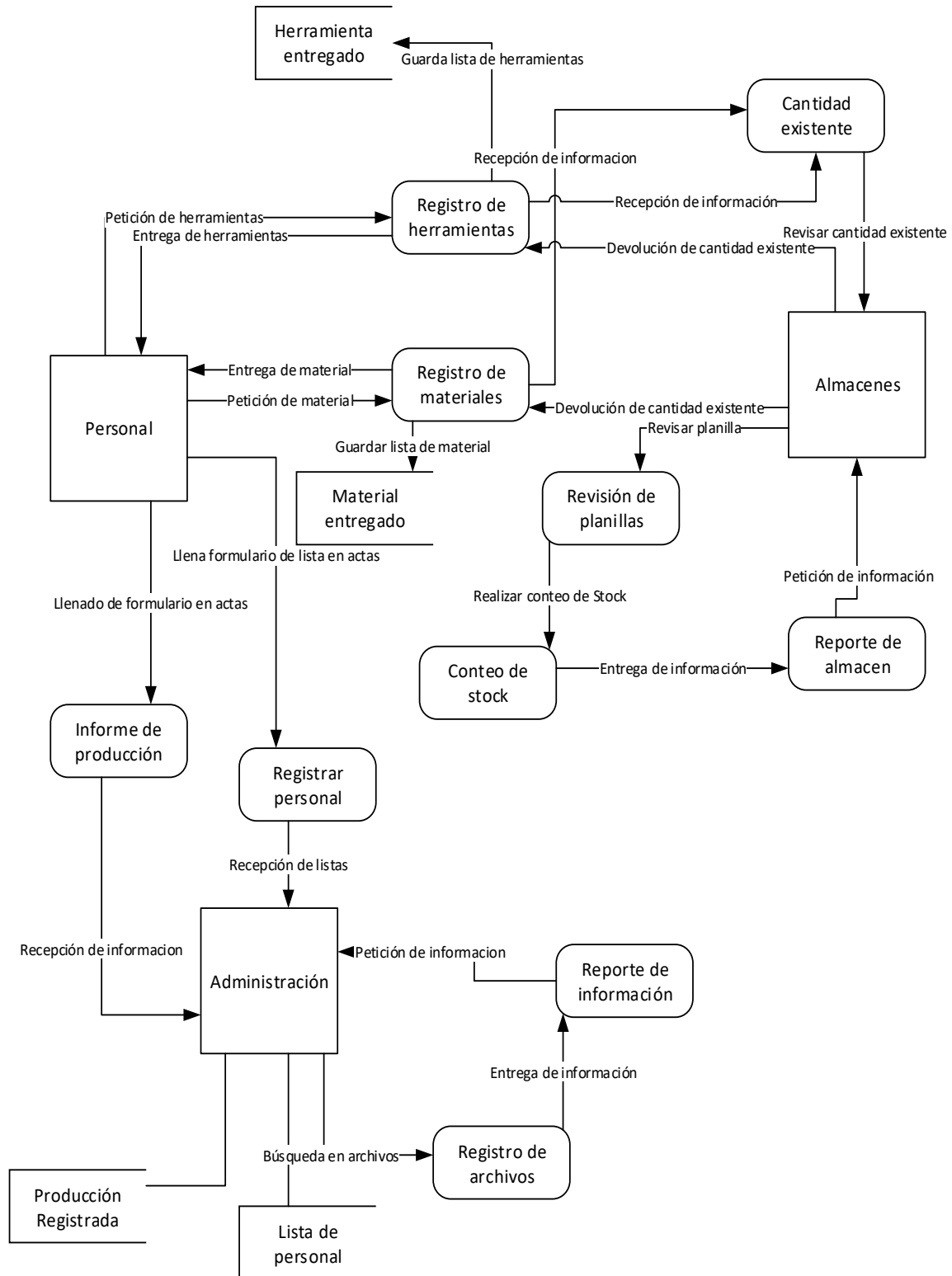
En El Presente Capitulo se mostrará el desarrollo de las etapas y modelos correspondientes, a las fases de análisis y diseño de sistema, siguiendo el proceso de desarrollo de la metodología UWE, detallado en el capítulo anterior (marco teórico).

#### **3.1 Análisis de la situación actual**

El administrador se encarga de registrar las planillas del personal que existe en la cooperativa minera Bolsa negra, como también los registros de la producción que se realiza. Estos procesos se realizan manualmente en planillas impresas, los que representa un gran esfuerzo para en administrador y para el personal que tiene la cooperativa, ya que se demora a la hora de hacer el registro, como también a la hora de verificar algún dato sobre la producción o sobre el personal, si contaran con sistema de información los procesos que realizan todos los días se realizarían en un menor tiempo.

También existe problemas a la hora de realizar la petición de materiales como también de herramientas ya que para ello se tiene que hacer una lista en un papel, después ir a revisar si aún se tiene el stock del material o las herramientas que se requiere, una vez revisado en almacenes, recién se llena el formulario de herramientas y materiales salientes para que se pueda registrar.

Una vez finalizado el día, el encargado del almacén realiza un reporte de los materiales y herramientas que fueron utilizados, por lo que para el encarado del almacén es otra tarea realizar el reporte y tendría una pérdida de tiempo.



**Figura N° 16** Diagrama de flujo de datos

Fuente: (elaboración propia)

## **3.2 Análisis de requerimientos**

### **3.2.1 Requerimientos de hardware**

Para el presente proyecto se necesitará una computadora que cumplirá las funciones de un servidor la cual tiene que estar conectado a una red intranet y como también a internet.

La computadora tiene que tener las siguientes características mínimas para un buen funcionamiento.

1.1 Procesador ryzen 5 3200

1.2 Placa madre b450m

1.3 Tarjeta ram de 8gigas

1.4 Disco duro de 1 tera

1.5 Fuente de energía

### **3.2.2 Requisitos de software**

Para que el presente proyecto sea puesto en funcionamiento se necesita software como un sistema operativo Windows en donde correrá todos los de más programas que se necesita, gestor de base de datos MariaDB desde la cual estará corriendo la base de datos del sistema de información, también se requiere cualquier gestor de navegación como por ejemplo puede ser el opera o el Chrome los cuales nos servirán para que el usuario pueda interactuar con el interfaz del sistema de información, también se necesita el servidor apache la cual nos permite servir contenido a las peticiones que vienen desde los clientes web (navegadores).

### 3.2.3 Requerimientos funcionales

Los requerimientos funcionales para el modelado del sistema se detallarán en la siguiente tabla.

**Tabla 5**  
*Requerimientos funcionales*

Ref.	Función	Categoría
R.1.1	El sistema debe tener seguridad en el acceso a la información del sistema.	Evidente
R.1.2	Acceder al sistema por tipos de usuarios (Administrador, Encargado de almacén y Trabajador)	Evidente
R.1.3	Registro de asistencia del trabajador	Evidente
R.1.4	Registro de ingreso de materiales y herramientas	Evidente
R.1.5	Registrar entrega de materiales al trabajador	Evidente
R.1.6	Registrar entrega de herramientas al trabajador	Evidente
R.1.7	Registrar cantidad de producción de oro y/o wólfam	Evidente
R.1.8	Validación del entrada de texto	Oculto
R.1.9	Generar reporte de material existente	Evidente
R.1.10	Generar reporte de ingreso material y herramientas	Evidente
R.1.11	Generar reporte de entrega de materiales y herramientas	Evidente
R.1.11	Generar reportes de asistencia del trabajador	Evidente
R.1.12	Generar reporte de producción de oro y/o wólfam	Evidente
R.1.13	Inicio y cierre del sistema	Oculto

Fuente: (Elaboración propia)

### 3.2.4 Requerimientos no funcionales

En la siguiente tabla se describirá los requerimientos no funcionales

**Tabla 6**  
*Requerimientos no funcionales*

Ref.	Función	Categoría
<b>R.1.1</b>	El sistema debe visualizarse y funcionar correctamente en cualquier navegador como ser Chrome, Mozilla, Opera, etc.	Evidente
<b>R.1.2</b>	Mantenimiento adecuado a la red local	Evidente
<b>R.1.3</b>	Respaldo energético del servidor, para asegurar la disponibilidad del sistema.	Evidente
<b>R.1.4</b>	Soporte y mantenimiento periódico para asegurar el buen rendimiento del sistema.	Evidente

Fuente: (Elaboración propia)

## 3.3 Diseño

### 3.3.1 Fase de inicio

En la fase de inicio identificaremos a los diferentes actores que interactúan en el uso del sistema de información.

#### 3.3.1.1 Identificación de actores

Una vez realizado el estudio preliminar y determinada la situación actual de la cooperativa, se pudo observar e identificar los posibles actores.

La identificación de actores en términos generales son usuarios del sistema los cuales interactúan, aportan y reciben información del sistema para coadyuvar a sus tareas cotidianas o necesidades demandadas.

A continuación, se describe de los actores o usuarios identificados.

### **Presidente de la cooperativa (Administrador)**

El presidente de la cooperativa es el que está en conocimiento de toda la información.

Cuando es necesario puede realizar todos los niveles de tarea existente en la cooperativa.

Es la persona que necesita tener la mayor parte de la información para así tener un control y seguimiento de todas las actividades de la cooperativa. Sus funciones son:

- Asignar a cada sección una tarea específica y supervisar el avance.
- Solicita reportes de materiales y herramientas existentes para la toma de decisiones.
- Aprueba o reprueba el ingreso de nuevos trabajadores.

### **Encargado de almacén**

Se en carga de controlar los materiales, herramientas que se encuentra dentro de los almacenes de la cooperativa.

- Registra el material entrante y saliente del almacén.
- Controla el estado y la cantidad existente de las herramientas.
- Realiza reporte de inventarios.
- Elabora comprobante de salida y entrada de materiales



- Realiza solicitudes de compra de material y herramientas.

## **Trabajador**

Las funciones de los trabajadores son las siguientes:

- Realizar el avance en los socavones de cada sección existente en la cooperativa.
- Trasladar la carga hacia la maquinaria de molido.
- Preparar el material requerido para la explotación de oro y wólfram.
- Abrir caminos hacia nuevos lugares de perforación.

### **3.3.2 Fase de elaboración**

En esta fase se analiza el dominio del problema de manera que se facilite su comprensión, estableciendo el contexto del sistema, aproximándonos al modelo del diseño, con una arquitectura de base sólida, logrando definir los objetivos del alcance del proyecto. El primer flujo de trabajo consiste en la captura de requisitos en el cual se define las principales funciones del sistema, como ser: caso de uso, descripción detallada de los casos de uso y finalmente el primer prototipo de interfaz.

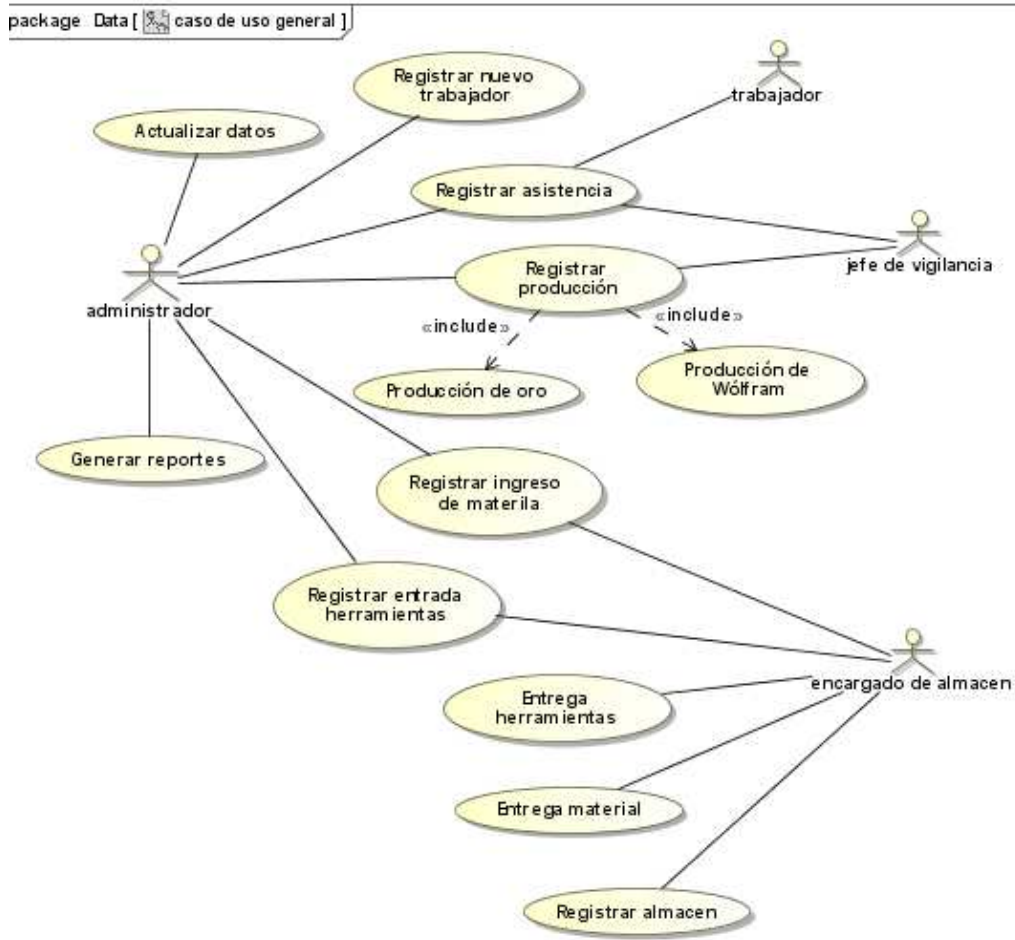
#### **3.3.2.1 Modelado de caso de uso**

Los casos de uso nos ayudan a capturar los requisitos adecuados. Cada usuario necesita que haga algunas funciones y los casos de uso representan los modos de utilizar el sistema.

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

### 3.3.2.1.1 Diagrama de caso de uso general

A continuación, se muestra el diseño de caso de uso general.

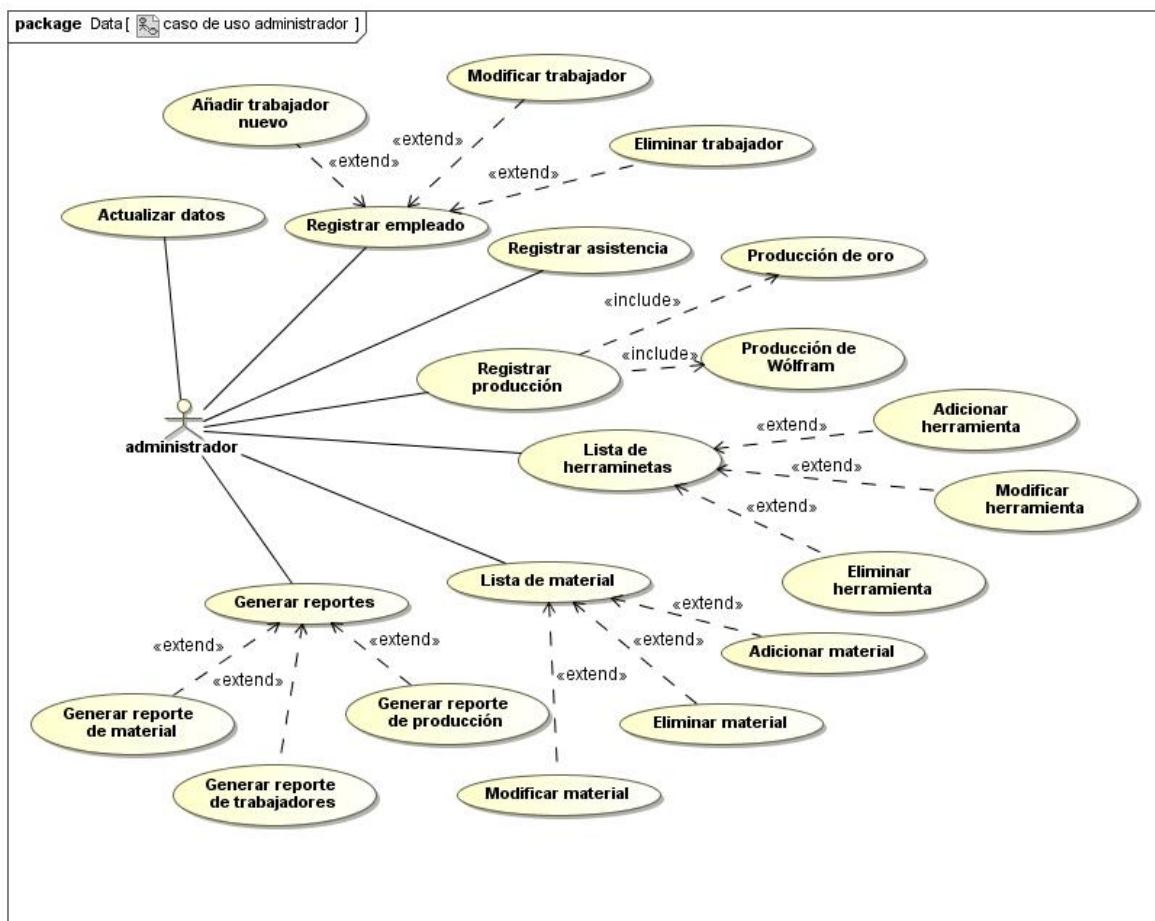


**Figura N° 17** Diagrama caso de uso general

Fuente: (Elaboración propia)

### 3.3.2.1.2 Caso de uso: Administrador

A continuación, se muestra el diseño de Caso de Uso Expandido del Administrador.



**Figura N° 18** *Caso de uso Administrador*

Fuente: (Elaboración Propia)

En la figura N° 18 el administrador del sistema de información puede realizar lo siguiente:

- Registrar datos de: Registrar producción, Materiales, Herramientas, con el fin de realizar un buen control.
- Obtener información sobre los trabajadores activos.
- Generar reportes sobre trabajadores, producción, materiales y herramientas.

**Tabla 7**

*Especificación del Caso de Uso de Registrar, Modificar y Eliminar Trabajador*

<b>CASO DE USO: REGISTRAR, MODIFICAR Y ELIMINARA TRABAJADOR</b>		
<b>Nombre</b>	Administración de trabajador	
<b>Actor</b>	Administrador	
<b>Descripción</b>	El administrador puede añadir, modificar y eliminar trabajadores	
	<b>Evento actor</b>	<b>Evento sistema</b>
<b>Flujo Principal</b>	Ingresar al sistema	Verificar si la sesión esta activa
	Ingresar usuario y contraseña	Inicia sesión y despliega el panel principal con los módulos correspondientes
	Del menú de navegación elegir trabajador y del sub menú elegir Lista de trabajador	Despliega la lista de empleados más las acciones de registrar, modificar o eliminar trabajador.
	Elige una de las acciones: registrar, modificar o eliminar trabajador.	Dependiendo de las opción seleccionada despliega un formulario para realizar dicha acción, ya sea registrar, modificar o eliminar.
	Introduce, modifica o elimina un trabajador.	Valida los datos del formulario y registra en la base de datos, mostrando un mensaje de notificación y retorna a la lista de trabajadores.

Fuente: (Elaboración propia)

**Tabla 8***Especificación del Caso de Uso de Registrar, Modificar y Eliminar Material*

<b>CASO DE USO: REGISTRAR, MODIFICAR Y ELIMINARA MATERIAL</b>		
<b>Nombre</b>	Administración de material	
<b>Actor</b>	Administrador	
<b>Descripción</b>	El administrador puede añadir, modificar y eliminar material	
	<b>Evento actor</b>	<b>Evento sistema</b>
<b>Flujo Principal</b>	Ingresar al sistema	Verificar si la sesión esta activa
	Ingresar usuario y contraseña	Inicia sesión y despliega el panel principal con los módulos correspondientes
	Del menú de navegación elegir Almacenes y del sub menú elegir Lista de material	Despliega la lista de materiales más las acciones de registrar, modificar o eliminar material.
	Elige una de las acciones: registrar, modificar o eliminar material.	Dependiendo de la opción seleccionada despliega un formulario para realizar dicha acción, ya sea registrar, modificar o eliminar.
	Introduce, modifica o elimina un material.	Valida los datos del formulario y registra en la base de datos, mostrando un mensaje de notificación y retorna a la lista de materiales.

Fuente: (Elaboración Propia).

**Tabla 9**

*Especificación del Caso de Uso de Registrar, Modificar y Eliminar Producción.*

<b>CASO DE USO: REGISTRAR, MODIFICAR Y ELIMINARA PRODUCCIÓN</b>		
<b>Nombre</b>	Administración de producción	
<b>Actor</b>	Administrador	
<b>Descripción</b>	El administrador puede añadir, modificar y eliminar producción	
	<b>Evento actor</b>	<b>Evento sistema</b>
<b>Flujo Principal</b>	Ingresar al sistema	Verificar si la sesión esta activa
	Ingresar usuario y contraseña	Inicia sesión y despliega el panel principal con los módulos correspondientes
	Del menú de navegación elegir Trabajador y del sub menú elegir Lista de producción	Despliega la lista de producción más las acciones de registrar, modificar o eliminar producción.
	Elige una de las acciones: registrar, modificar o eliminar producción.	Dependiendo de la opción seleccionada despliega un formulario para realizar dicha acción, ya sea registrar, modificar o eliminar.
	Introduce, modifica o elimina una producción.	Valida los datos del formulario y registra en la base de datos, mostrando un mensaje de notificación y retorna a la lista de producción.

Fuente: (Elaboración Propia).

**Tabla 10**

*Especificación del Caso de Uso de Registrar, Modificar y Eliminar herramienta*

<b>CASO DE USO: REGISTRAR, MODIFICAR Y ELIMINARA HERRAMIENTA</b>		
<b>Nombre</b>	Administración de herramientas	
<b>Actor</b>	Administrador	
<b>Descripción</b>	El administrador puede añadir, modificar y eliminar herramientas	
	<b>Evento actor</b>	<b>Evento sistema</b>
<b>Flujo Principal</b>	Ingresar al sistema	Verificar si la sesión esta activa
	Ingresar usuario y contraseña	Inicia sesión y despliega el panel principal con los módulos correspondientes
	Del menú de navegación elegir Almacenes y del sub menú elegir Lista de herramientas.	Despliega la lista de herramienta más las acciones de registrar, modificar o eliminar herramienta.
	Elige una de las acciones: registrar, modificar o eliminar herramienta.	Dependiendo de la opción seleccionada despliega un formulario para realizar dicha acción, ya sea registrar, modificar o eliminar.
	Introduce, modifica o elimina una herramienta.	Valida los datos del formulario y registra en la base de datos, mostrando un mensaje de notificación y retorna a la lista de herramienta.

Fuente: (Elaboración propia).

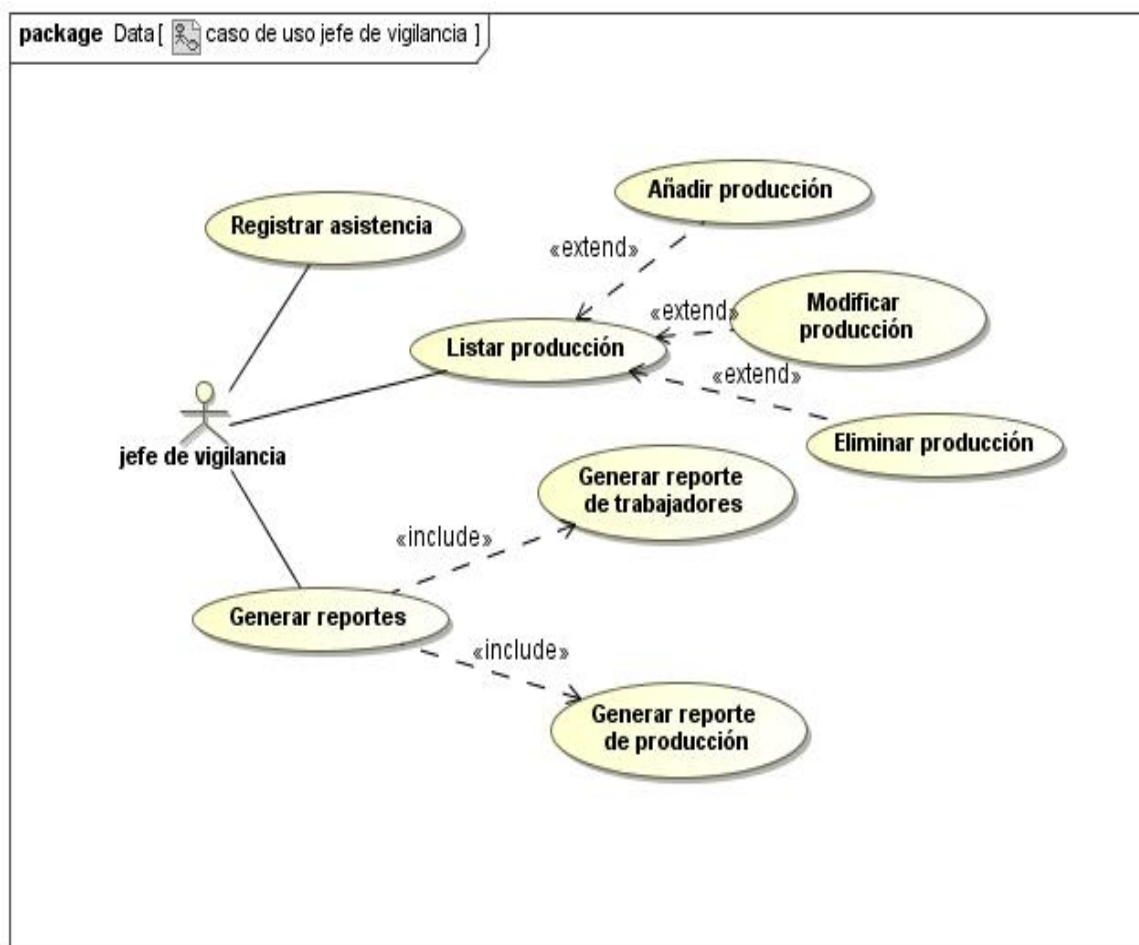
**Tabla 11**  
*Especificación del Caso de Uso de Generar Reportes.*

<b>CASO DE USO: GENERAR REPORTE</b>		
<b>Nombre</b>	Reportes	
<b>Actor</b>	Administrador	
<b>Descripción</b>	El administrador puede generar reporte de los trabajadores, herramientas, producción y materiales.	
	<b>Evento actor</b>	<b>Evento sistema</b>
<b>Flujo Principal</b>	Ingresar al sistema	Verificar si la sesión esta activa
	Ingresar usuario y contraseña	Inicia sesión y despliega el panel principal con los módulos correspondientes
	Del menú de navegación elegir Reportes y del sub menú elegir reporte empleado, reporte material, reporte herramientas o reporte producción.	Despliega formulario de reportes según seleccionado, con la acción de Imprimir Reporte.
	Elige la acción de Imprimir Reporte	Genera el Reporte en formato Pdf.

Fuente: (Elaboración propia).



### 3.3.2.1.3 Caso de uso: jefe de vigilancia



**Figura N° 19** Caso de uso jefe de vigilancia

Fuente: (Elaboración propia).

En la figura N° 19 el jefe de vigilancia puede realizar acciones como:

- Registrar asistencia del trabajador.
- Puede registrar modificar o eliminar producción.
- Generar reporte de la producción y del trabajador.

**Tabla 12**

*Especificación del Caso de Uso de Registrar, Modificar y Eliminar Producción.*

<b>CASO DE USO: REGISTRAR, MODIFICAR Y ELIMINARA PRODUCCIÓN</b>		
<b>Nombre</b>	Registro de producción	
<b>Actor</b>	Jefe de vigilancia	
<b>Descripción</b>	El jefe de vigilancia puede añadir, modificar y eliminar producción	
	<b>Evento actor</b>	<b>Evento sistema</b>
<b>Flujo Principal</b>	Ingresar al sistema	Verificar si la sesión esta activa
	Ingresar usuario y contraseña	Inicia sesión y despliega el panel principal con los módulos correspondientes
	Del menú de navegación elegir Trabajador y del sub menú elegir Lista de producción	Despliega la lista de producción más las acciones de registrar, modificar o eliminar producción.
	Elige una de las acciones: registrar, modificar o eliminar producción.	Dependiendo de la opción seleccionada despliega un formulario para realizar dicha acción, ya sea registrar, modificar o eliminar.
	Introduce, modifica o elimina una producción.	Valida los datos del formulario y registra en la base de datos, mostrando un mensaje de notificación y retorna a la lista de producción.

**Tabla 13**

*Especificación del Caso de Uso de Registrar asistencia.*

<b>CASO DE USO: REGISTRAR ASISTENCIA</b>		
<b>Nombre</b>	Asistencia	
<b>Actor</b>	Jefe de vigilancia	
<b>Descripción</b>	El jefe de vigilancia puede registrar la asistencia del trabajador.	
	<b>Evento actor</b>	<b>Evento sistema</b>
<b>Flujo Principal</b>	Ingresar al sistema	Verificar si la sesión esta activa
	Ingresar usuario y contraseña	Inicia sesión y despliega el panel principal con el módulo de registro de asistencia
	Ingresar el número de carnet del trabajador en la casilla correspondiente	Valida el número de carnet introducido y registra en la base de datos, mostrando un mensaje de notificación.

Fuente: (Elaboración propia)

**Tabla 14**

*Especificación del Caso de Uso de Generar Reportes.*

<b>CASO DE USO: GENERAR REPORTE</b>	
<b>Nombre</b>	Reportes
<b>Actor</b>	Jefe de vigilancia
<b>Descripción</b>	El jefe de vigilancia puede generar reporte de los trabajadores y de producción.

	<b>Evento actor</b>	<b>Evento sistema</b>
<b>Flujo Principal</b>	Ingresar al sistema	Verificar si la sesión esta activa
	Ingresar usuario y contraseña	Inicia sesión y despliega el panel principal con los módulos correspondientes
	Del menú de navegación elegir Reportes y del sub menú elegir reporte empleado, reporte material, reporte herramientas o reporte producción.	Despliega formulario de reportes según seleccionado, con la acción de Imprimir Reporte.
	Elige la acción de Imprimir Reporte	Genera el Reporte en formato Pdf.

Fuente: (Elaboración propia).

#### 3.3.2.1.4 Caso de uso: Trabajador



**Figura N° 20** Caso de uso trabajador

Fuente: (Elaboración propia)

En la figura N° 20 los trabajadores pueden realizar las siguientes acciones:

- Registrar su asistencia
- Realizar pedidos de herramientas.

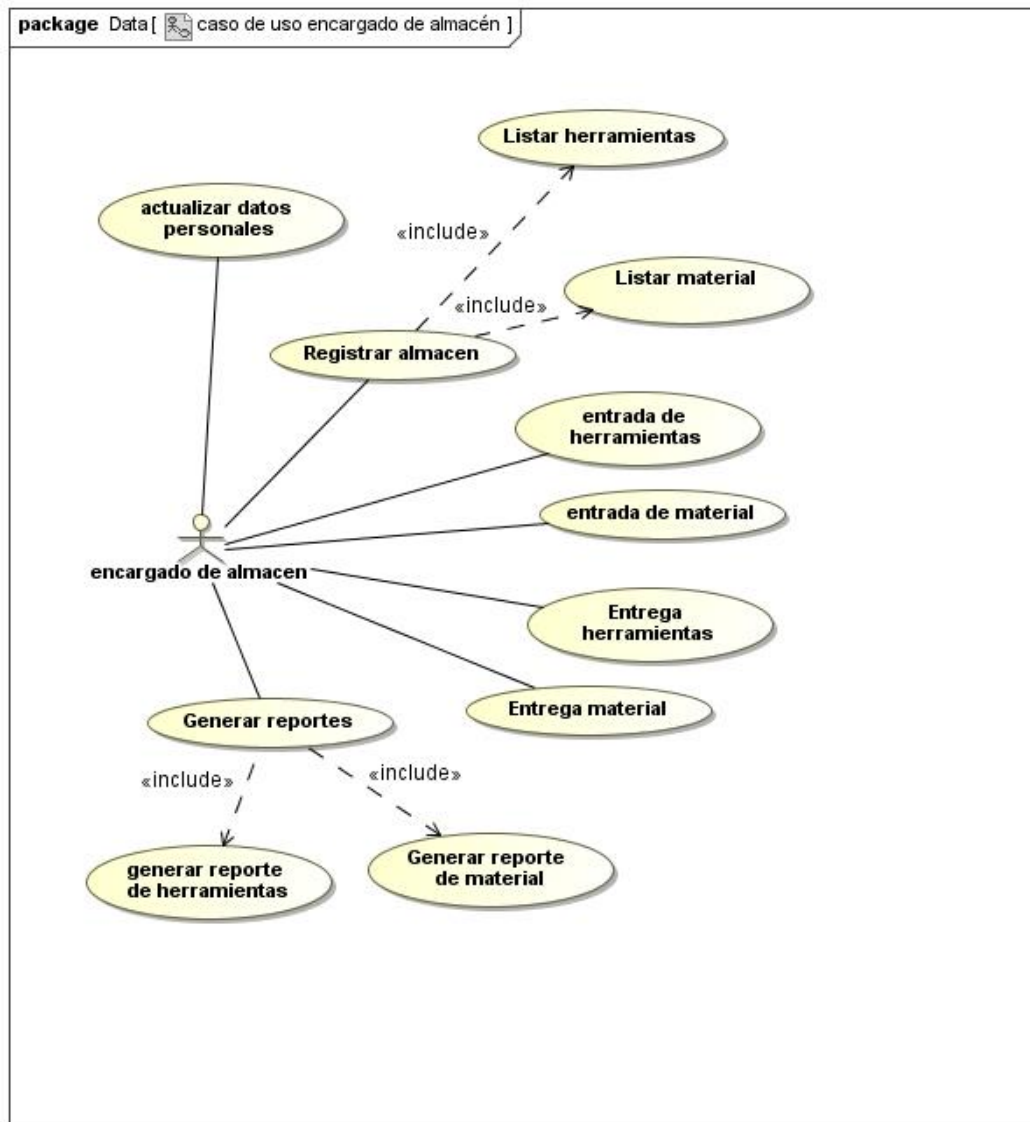
**Tabla 15**

*Especificación del Caso de Uso de Registrar asistencia del trabajador*

<b>CASO DE USO: REGISTRAR ASISTENCIA</b>		
<b>Nombre</b>	Asistencia	
<b>Actor</b>	Trabajador	
<b>Descripción</b>	El trabajador puede registrar su asistencia.	
	<b>Evento actor</b>	<b>Evento sistema</b>
<b>Flujo Principal</b>	Ingresar al sistema	Verificar si la sesión esta activa
	Solicita iniciar sección al jefe de vigilancia	Inicia sesión y despliega el panel principal con el módulo de registro de asistencia
	Ingresar el número de carnet del trabajador en la casilla correspondiente	Valida el número de carnet introducido y registra en la base de datos, mostrando un mensaje de notificación.

Fuente: (Elaboración propia).

### 3.3.2.1.5 Caso de uso: encargado de almacén



**Figura N° 21** Caso de uso encargado de almacén

Fuente: (Elaboración propia)

En la figura N° 21 el encargado de almacén puede realizar actividades como:

Registrar entradas y salidas de materiales como también entradas y salidas de herramientas y también generar reportes de los mismos.

**Tabla 16***Especificación del Caso de Uso de movimiento de herramientas*

<b>CASO DE USO: REGISTRAR ENTRADA Y SALIDA DE HERRAMIENTA</b>		
<b>Nombre</b>	Administración de almacén	
<b>Actor</b>	Encargado de almacén	
<b>Descripción</b>	El encargado de almacén puede registrar los movimientos de las herramientas en el almacén.	
	<b>Evento actor</b>	<b>Evento sistema</b>
<b>Flujo Principal</b>	Ingresar al sistema	Verificar si la sesión esta activa
	Ingresar usuario y contraseña	Inicia sesión y despliega el panel principal con los módulos correspondientes
	Del menú de navegación elegir Almacenes y del sub menú elegir Lista de salida o entrada de herramientas.	Despliega la lista de herramienta más las acciones de entrada o salida de herramientas.
	Elige una de las acciones: entrada o salida de herramientas	Dependiendo de la opción seleccionada despliega un formulario para realizar dicha acción, ya sea entrada o salida .
	Introduce, los datos para las acciones de salida o entrada de herramienta.	Valida los datos del formulario y registra en la base de datos, mostrando un mensaje de notificación y retorna a la lista de herramienta.

Fuente: (Elaboración propia)

**Tabla 17***Especificación del Caso de Uso de movimiento de material*

<b>CASO DE USO: REGISTRAR ENTRADA Y SALIDA DE MATERIALES</b>		
<b>Nombre</b>	Administración de almacén	
<b>Actor</b>	Encargado de almacén	
<b>Descripción</b>	El encargado de almacén puede registrar los movimientos de los materiales en el almacén.	
	<b>Evento actor</b>	<b>Evento sistema</b>
<b>Flujo Principal</b>	Ingresar al sistema	Verificar si la sesión esta activa
	Ingresar usuario y contraseña	Inicia sesión y despliega el panel principal con los módulos correspondientes
	Del menú de navegación elegir Almacenes y del sub menú elegir Lista de salida o entrada de materiales.	Despliega la lista de materiales más las acciones de entrada o salida de materiales.
	Elige una de las acciones: entrada o salida de materiales.	Dependiendo de la opción seleccionada despliega un formulario para realizar dicha acción, ya sea entrada o salida .
	Introduce, los datos para las acciones de salida o entrada de material.	Valida los datos del formulario y registra en la base de datos, mostrando un mensaje de notificación y retorna a la lista de material.

Fuente: (Elaboración propia)



### 3.4 Desarrollo del modelo

#### 3.4.1 Modelo de contenido

El diagrama de contenido tiene el propósito de mostrar las relaciones entre las entidades y las estructuras de los datos que se encuentran alojados en el sistema.

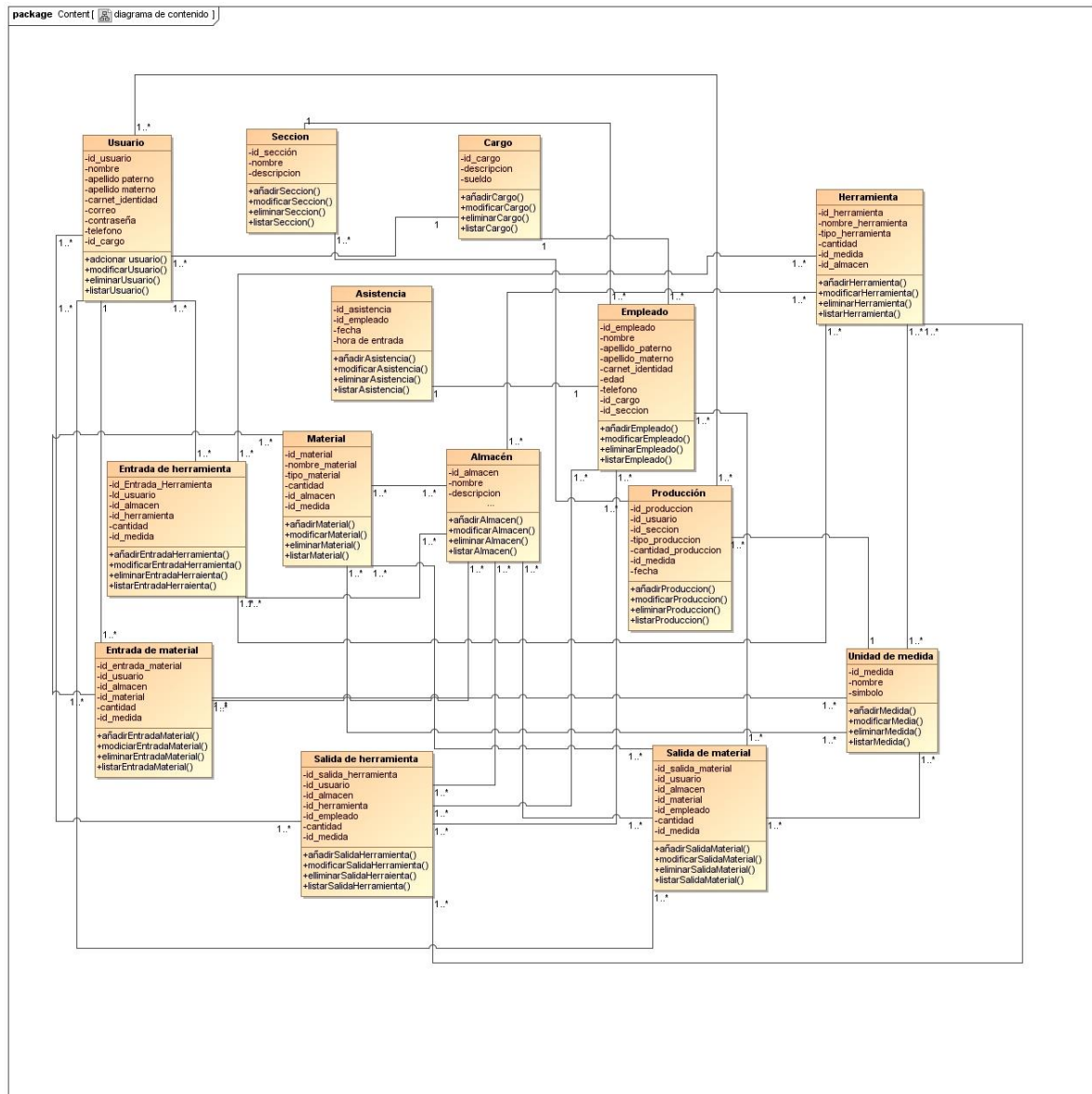
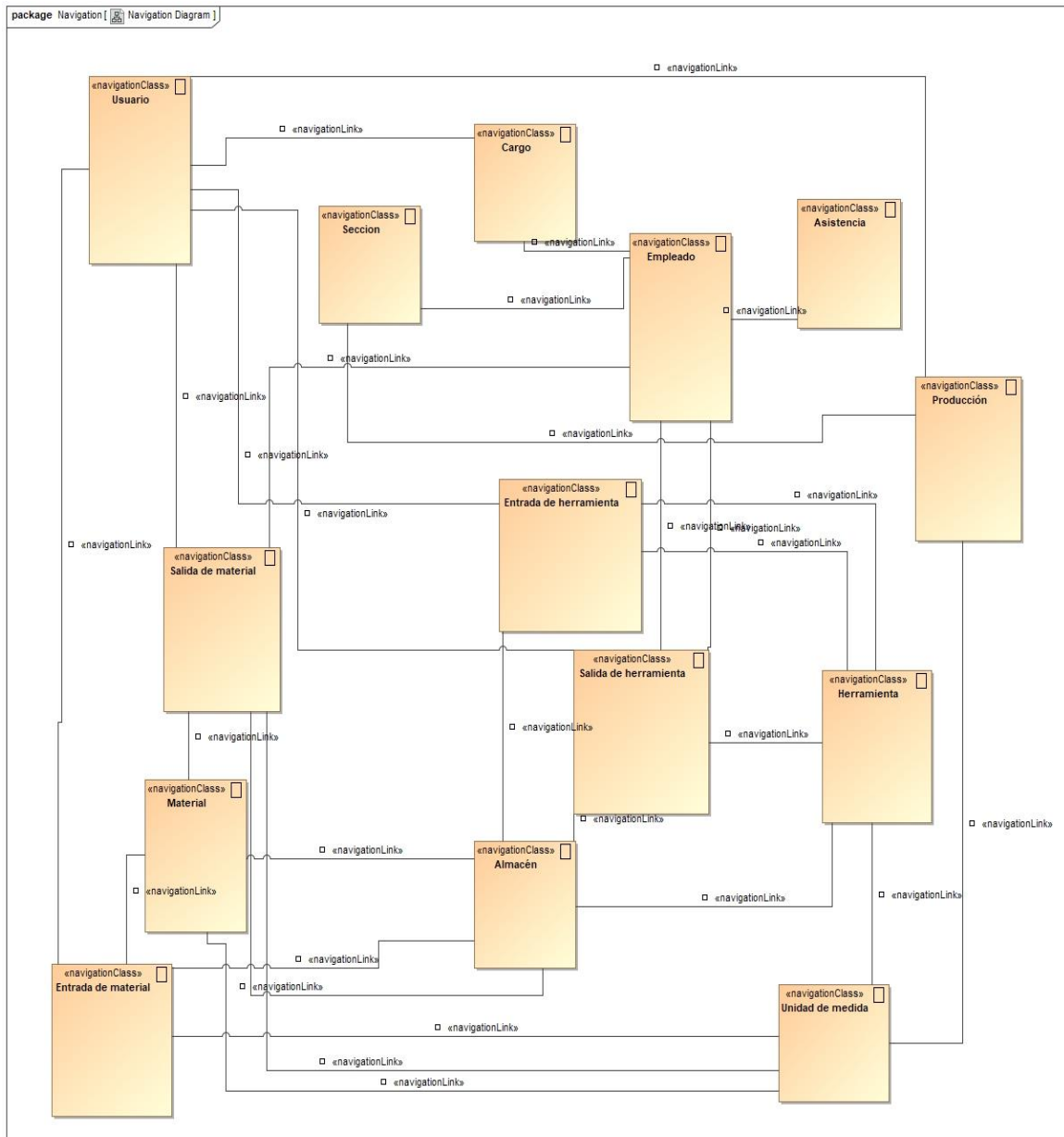


Figura N° 22 Modelo de contenido del sistema

Fuente: (Elaboración propia)

### 3.4.2 Modelo de navegación

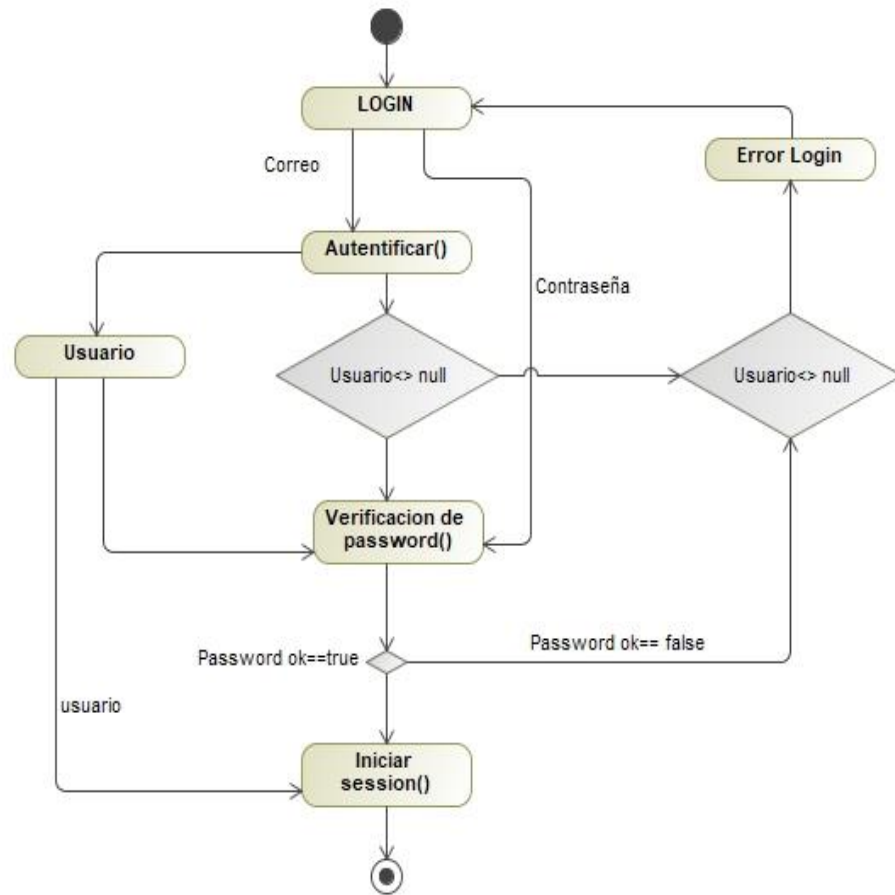
Tiene como objetivo la representación de nodos y enlaces de la estructura de hipertexto, y el diseño de las rutas de navegación mediante diagramas de clases UML.



**Figura N° 23** Modelo de navegación

Fuente: (Elaboración propia)

En la Figura N° 24, muestra el flujo de proceso que sigue en la autenticación de usuario.



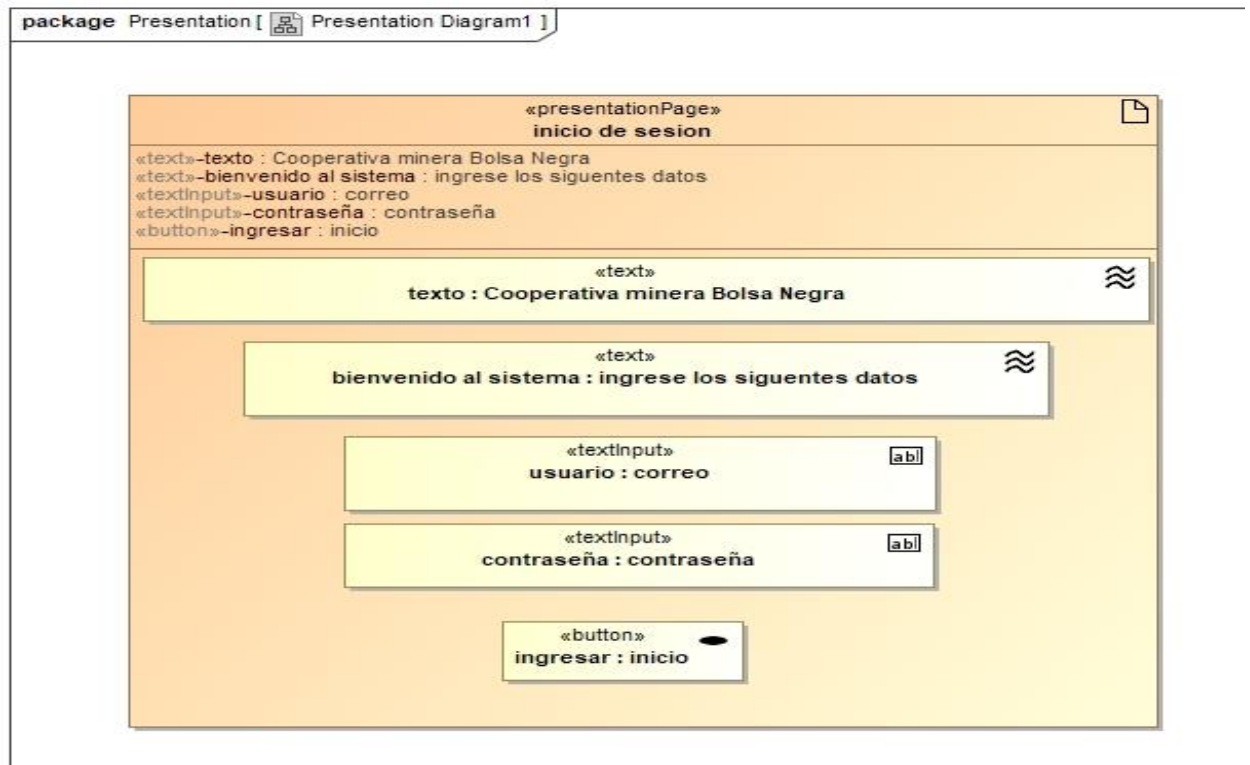
**Figura N° 24** Modelo de Flujo de Proceso de Autenticar Usuario del Sistema.

Fuente:(Elaboración Propia)

### 3.4.3 Modelo de presentación

A continuación, se mostrarán los modelos de presentación según UWE propone para la construcción del sistema en forma de bosquejos.

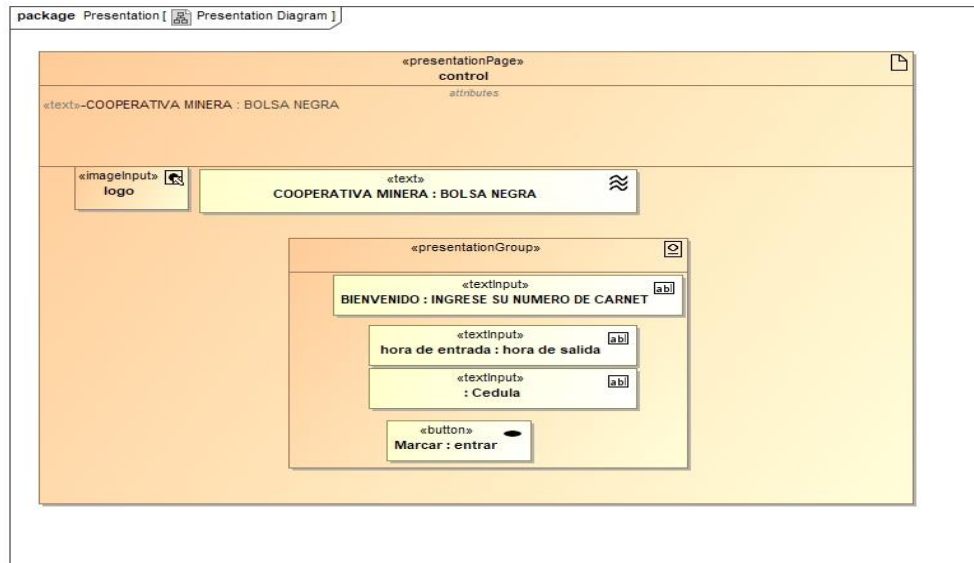
En la Figura N° 25, se muestra el diseño que se usara para la autenticación del Usuario, realizando la verificación del Sistema con los datos del Usuario y Contraseña. Que contendrá las medidas de seguridad de autenticación para el sistema.



**Figura N° 25** Pantalla de autenticación del sistema

**Fuente:** (Elaboración Propia)

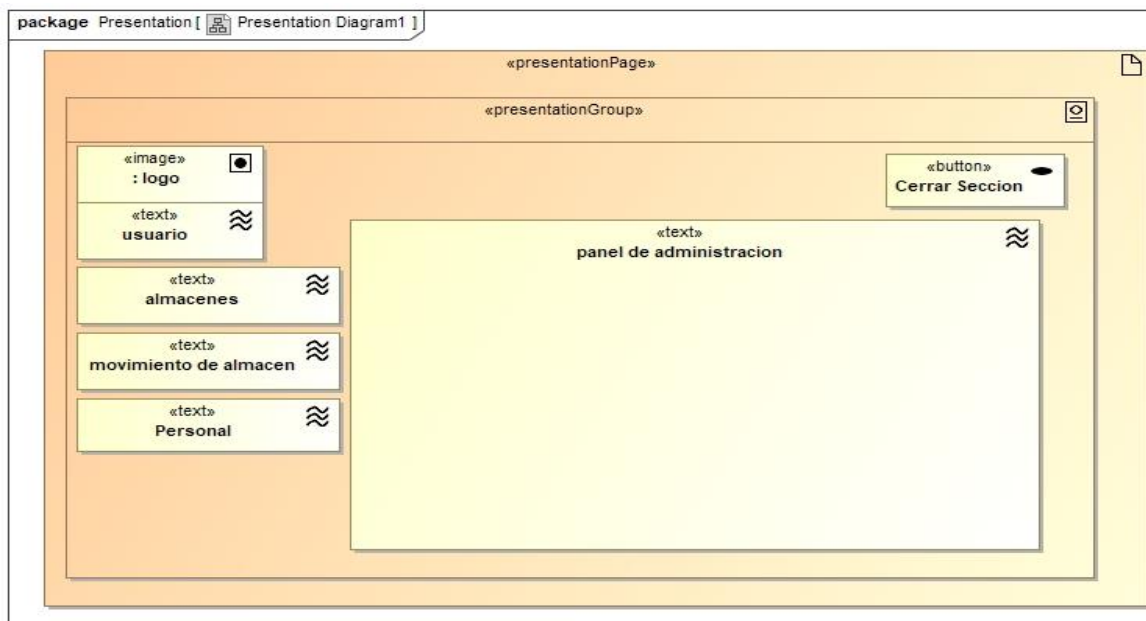
En la Figura N° 26, se muestra el diseño que se usara para el control de asistencia, donde cada trabajador podrá seleccionar marcar su hora de entrada o su hora de salida para lo cual usaran su número de carnet.



**Figura N° 26** Diagrama de presentación para Control

**Fuente:** (Elaboración propia)

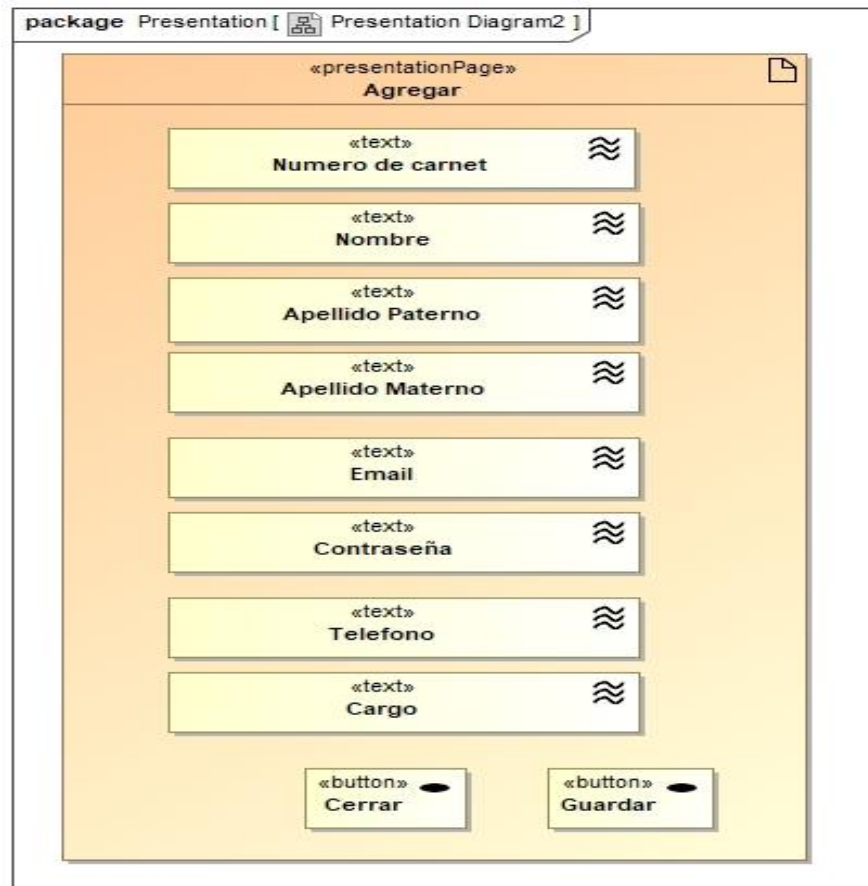
En la Figura N° 27, se muestra el diseño para el Usuario Administrador, mostrando el menú de navegación con la que cuenta el sistema.



**Figura N° 27** Modelo de Presentación de Modulo de Administración del sistema

**Fuente:** (Elaboración propia)

En la Figura N° 27, se muestra el diseño para el registro que se usara para agregar al personal como también la parte de almacenes.



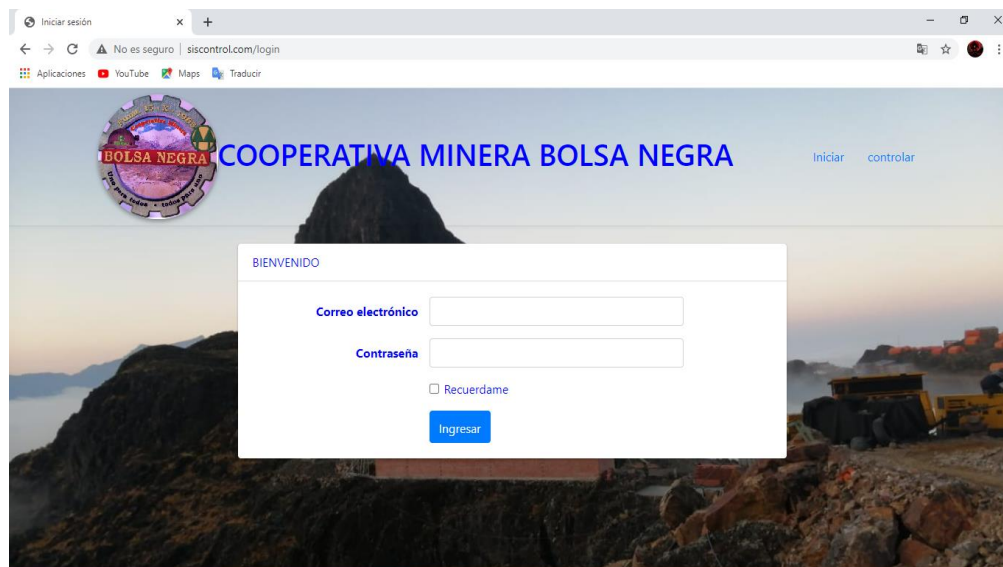
**Figura N° 28** *Modelo de Presentación de Registro*

**Fuente:** (Elaboración propia)

### 3.5 Fase de construcción

#### 3.5.1 Modelo de implementación

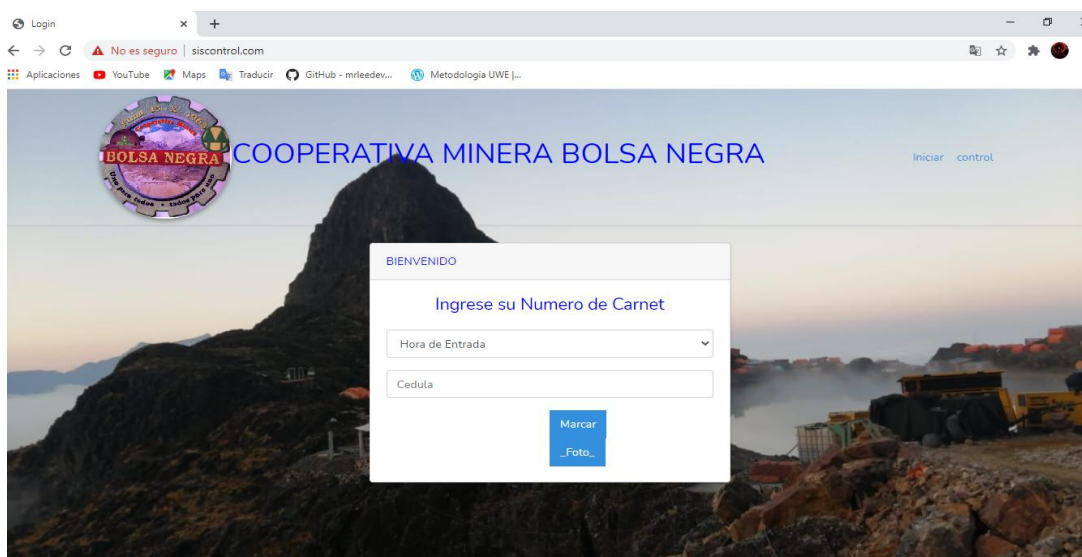
En esta fase de implementación consiste en mostrar el desarrollo de la presentación de las interfaces del sistema y sus elementos construidos a partir del diseño del modelo de presentación UWE. Para el sistema tenemos la siguiente secuencia de pantallas básicas.



**Figura N° 29** *Pantalla de Autenticación de usuario*

Fuente: (Elaboración propia)

La Figura N° 29, presenta la pantalla de autenticación al sistema. En la que el usuario del sistema tiene que autenticarse ingresando su correo y contraseña. Los datos ingresados serán validados y verificados con la base de datos del sistema.



**Figura N° 30** *Pantalla de control de trabajador*

Fuente: (Elaboración Propia)

En la Figura N° 30, se muestra la pantalla de control de asistencia del trabajador. En la que el trabajador podrá ingresar su número de carnet y así marcar su hora de entrada o la hora de salida. Los datos de su asistencia serán ingresados y guardados en la base de datos del sistema.



**Figura N° 31** Pantalla de menú

**Fuente:** (Elaboración Propia)

Una vez autenticado el usuario, ingresa a la pantalla principal donde se encuentra el menú de navegación, también se puede ver la cantidad de empleados con la que cuenta la cooperativa y cuántos de ellos están activos, como también el número de secciones que tiene la cooperativa minera, en un costado se puede ver la barra de navegación para cada acción que desee realizar el usuario autenticado, tal como se observa en la Figura N° 31.



#	Nombre	Seccion	Descripcion	acciones
1	almacen Derecha entrada	Derecha entrada	almacen de nuevo avance	[Edit] [Delete]
2	almacen Piso A	Piso A	almacen de nuevo avance	[Edit] [Delete]
3	almacen Recorte Rincon	Recorte Rincon	almacen de nuevo avance	[Edit] [Delete]
4	almacen Piso entrada	Piso Entrada	almacen de nuevo avance	[Edit] [Delete]
5	almacen Izquierda entrada	Izquierda Entrada	almacen de nuevo avance	[Edit] [Delete]

**Figura N° 32** *Pantalla de listado de almacén*

**Fuente:** (Elaboración Propia)

En la Figura N° 32, del menú de navegación, al seleccionar la opción de Almacenes se despliega las sub-opciones de: Almacén, Herramientas, Materiales y unidad de medida, de las cuales seleccionamos la opción de Almacén en la que el usuario puede ver una lista de los almacenes ya registrados, también podrá realizar modificaciones, eliminar un registro o adicionar un nuevo almacén.

**Agregar Almacen**

Nombre:

Seccion:

Descripcion:

**Figura N° 33** *Pantalla de registro de almacén*

**Fuente:** (Elaboración Propia)

En la figura N° 33, se muestra el modelo de registro de datos que se utilizara para agregar un nuevo dato.



#	Almacen	Nombre	Tipo	Cantidad	Medida	Estado	Acciones
1	almacen Derecha entrada	Pala	manual	31	unidades	nuevo	[Editar] [Eliminar]
2	almacen Derecha entrada	Carretilla	manual	7	unidades	nuevo	[Editar] [Eliminar]
3	almacen Derecha entrada	Carros	manual	0	unidades	nuevo	[Editar] [Eliminar]
4	almacen Derecha entrada	Combo	manual	10	unidades	nuevo	[Editar] [Eliminar]
5	almacen Derecha entrada	Picota	manual	7	unidades	nuevo	[Editar] [Eliminar]

**Figura N° 34** Pantalla de listado de herramientas

**Fuente:** (Elaboración Propia)

En la Figura N° 34, se muestra la pantalla de las herramientas que existen en los almacenes de cada sección, en esta parte se puede realizar búsquedas, registro de nuevas herramientas, modificación y eliminar un registro.



#	Almacen	Nombre	Tipo	Cantidad	Medida	Estado	Acciones
1	almacen Derecha entrada	dinamita	explosivo	377	unidades	usable	[Editar] [Eliminar]
2	almacen Derecha entrada	gasolina	combustible	74	litros	usable	[Editar] [Eliminar]
3	almacen Derecha entrada	fulminante	explosivo	200	unidades	usable	[Editar] [Eliminar]
4	almacen Derecha entrada	Mecha(guia)	explosivo	70	metros	usable	[Editar] [Eliminar]
5	almacen Derecha entrada	Nitrate de amonio	quimico	20	kilogramos	usable	[Editar] [Eliminar]

**Figura N° 35** Pantalla de listado de materiales

**Fuente:** (Elaboración Propia)

En la Figura N° 35, se muestra la pantalla del listado de materiales existentes en los almacenes de cada sección, en esta parte se puede realizar búsquedas, registro de nuevos materiales, modificación y eliminar un registro.



#	Usuario	Almacen	Material	Cantidad	Medida	Estado	Fecha	Acciones
1	marin duran	almacen Derecha entrada	gasolina	20	litros		2020-10-29	[Eliminar]
2	marin duran	almacen Derecha entrada	dinamita	33	unidades		2020-10-29	[Eliminar]
3	marin duran	almacen Piso A	gasolina	33	litros		2020-10-29	[Eliminar]
4	marin duran	almacen Derecha entrada	dinamita	20	unidades		2020-11-04	[Eliminar]
5	marin duran	almacen Derecha entrada	dinamita	200	unidades		2020-11-07	[Eliminar]

**Figura N° 36** pantalla de listado de material entrante

**Fuente:** (Elaboración Propia)

En la Figura N° 36, se muestra la pantalla del listado de entrada de materiales a los almacenes de cada sección, en esta parte se puede realizar búsquedas, registro de nuevos ingresos o eliminar un registro.



#	Usuario	Almacen	Material	Empleado	Cantidad	Medida	Fecha	Acciones
1	marin duran	almacen Derecha entrada	gasolina	martin churata	6	litros	2020-10-29	[Eliminar]
2	marin duran	almacen Piso A	dinamita	Macos Apaza	20	unidades	2020-10-29	[Eliminar]
3	julio quispe	almacen Piso A	dinamita	Pedro Mamani	30	unidades	2020-11-04	[Eliminar]
4	marin duran	almacen Derecha entrada	dinamita	Julio Quiroz	33	unidades	2020-11-07	[Eliminar]
5	marin duran	almacen Derecha entrada	gasolina	martin churata	80	litros	2020-11-18	[Eliminar]

**Figura N° 37** Pantalla de listado de salida de material

**Fuente:** (Elaboración Propia)

En la Figura N° 37, se muestra la pantalla del listado de materiales entregados a los trabajadores en cada sección, en esta parte se puede realizar búsquedas, registro de nuevas entregas o eliminar un registro de una entrega.



#	Usuario	Almacen	Herramienta	Cantidad	Medida	Estado	Fecha	Acciones
1	marin duran	almacen Derecha entrada	pala	2	unidades	reparable	2020-11-12	
2	marin duran	almacen Derecha entrada	Carros	4	unidades	nuevo	2020-11-12	
3	marin duran	almacen Derecha entrada	pala	1	unidades	reparable	2020-11-18	
4	marin duran	almacen Derecha entrada	Pala	20	unidades	nuevo	2020-11-18	
5	marin duran	almacen Derecha entrada	Combo	4	unidades	nuevo	2020-11-21	

**Figura N° 38** Pantalla de listado de entrada de herramientas

**Fuente:** (Elaboración Propia)

En la Figura N° 38, se muestra la pantalla del listado de entrada de herramientas a los almacenes de cada sección, en esta parte se puede realizar búsquedas, registro de nuevos ingresos o eliminar un registro.



#	Usuario	Almacen	Herramienta	Empleado	Cantidad	Medida	Fecha	Acciones
1	marin duran	almacen Derecha entrada	Pala	Manuel Fernandez	2	unidades	2020-10-29	
2	marin duran	almacen Piso A	carretilla	Pedro Mamani	1	unidades	2020-10-29	
3	marin duran	almacen Derecha entrada	Carros	Julio Quiroz	4	unidades	2020-11-12	
4	marin duran	almacen Derecha entrada	Carros	Julio Quiroz	3	unidades	2020-11-12	
5	marin duran	almacen Derecha entrada	Pala	Felipe Mamani	6	unidades	2020-11-22	

**Figura N° 39** Pantalla de listado de salida de herramientas

**Fuente:** (Elaboración Propia)

En la Figura N° 39, se muestra la pantalla del listado de herramientas entregados a los trabajadores en cada sección, en esta parte se puede realizar búsquedas, registro de nuevas entregas o eliminar un registro de una entrega.

#	Cedula	Nombre	Apellidos	Correo	Telefono	Cargo	Acciones
1	6976558	jorge	fernandez quiroz	prueba@hotmail.com	2566236	Delegado	[Edit] [Delete]
2	401233	julio	quispe quispe	jhum@gmail.com	2566236	Delegado	[Edit] [Delete]
3	1234567	jorge	chura quispe	cooperativa@gmail.com	79645214	Delegado	[Edit] [Delete]
4	1234567	marin	durán Quispe	cooperativa1@gmail.com	75259656	Delegado	[Edit] [Delete]

**Figura N° 40** Pantalla de listado de usuarios

**Fuente:** (Elaboración Propia)

En la Figura N° 40, se muestra la pantalla del listado de usuarios que existen para poder acceder al sistema, en esta parte se puede realizar búsquedas, registro de nuevos usuarios, modificación y eliminar un registro.

#	Cedula	Nombre	Apellidos	Edad	Genero	Telefono	Cargo	Seccion	Acciones
1	6976558	marin	durán quispe	25	masculino	75259656	Delegado	Derecha entrada	[Edit] [Delete]
2	2345234	martin	churata mamani	30	masculino	79645214	Perforista	Derecha entrada	[Edit] [Delete]
3	4312234	Manuel	Fernandez Quille	23	masculino	72134245	Molinero	Derecha entrada	[Edit] [Delete]
4	6542341	Roger	Fernandez Quille	25	masculino	64356235	Jefe de mantenimiento	Derecha entrada	[Edit] [Delete]
5	5423456	Felipe	Mamani Mamani	40	masculino	79645214	Carrero	Derecha entrada	[Edit] [Delete]

**Figura N° 41** Pantalla de listado de trabajadores

**Fuente:** (Elaboración Propia)

En la Figura N° 41, se muestra la pantalla del listado de empleados con los que cuenta la cooperativa, en esta parte se puede realizar búsquedas, registro de nuevos usuarios, modificación y eliminar un registro.



#	Cedula	Nombre	Fecha	Hora de entrada	foto de entrada	Hora de salida	foto de salida	Acciones
1	6976558	durán marín	2020-11-22	10:15:00		11:48:00		
2	2345612	Apaza Macos	2020-11-21	17:41:00		17:45:00		
3	6976558	durán marín	2020-11-19	14:52:00		14:54:00		
4	2345612	Apaza Macos	2020-11-18	16:51:00		16:54:00		
5	4312234	Fernandez Manuel	2020-11-18	15:05:00		16:55:00		

**Figura N° 42** Pantalla de listado de asistencia

**Fuente:** (Elaboración Propia)

En la Figura N° 42, se muestra la pantalla del listado de asistencias de los empleados con los que cuenta la cooperativa, en esta parte se puede realizar búsquedas, registro de asistencias, modificación y eliminar un registro.



#	Usuario	Sección	Trabajador	Inicio de Permiso	Fin de Permiso	Acciones
1	marín durán	Derecha entrada	martín churata	2020-11-11	2020-11-12	
2	marín durán	Derecha entrada	Felipe Mamani	2020-11-19	2020-11-20	
3	marín durán	Derecha entrada	Manuel Fernandez	2020-11-23	2020-11-23	

**Figura N° 43** Pantalla de listado de permiso

**Fuente:** (Elaboración Propia)

En la Figura N° 43, se muestra la pantalla del listado de permisos concedidos a los empleados de la cooperativa minera.

#	Usuario	Seccion	Tipo	Cantidad	Medida	Fecha	Acciones
1	marin duran	Derecha entrada	oro	1.5	kilogramos	2020-10-28	[Edit] [Delete]
2	marin duran	Derecha entrada	oro	1	kilogramos	2020-09-30	[Edit] [Delete]
3	jorge chura	Recorte Rincon	wolfram	6	kilogramos	2020-10-28	[Edit] [Delete]
4	julio quispe	Izquierda Entrada	wolfram	13	kilogramos	2020-10-28	[Edit] [Delete]
5	marin duran	Derecha entrada	oro	12	kilogramos	2020-11-07	[Edit] [Delete]

**Figura N° 44** Pantalla de listado de producción

**Fuente:** (Elaboración Propia)

En la Figura N° 44, se muestra la pantalla del listado de la producción tanto de oro como la de wólfram, en esta parte se puede realizar búsquedas, registro de nuevas producciones, modificación y eliminar un registro.

#	Descripción	Sueldo	acciones
1	Delegado	5000	[Edit] [Delete]
2	Carrero	2000	[Edit] [Delete]
3	Perforista	3500	[Edit] [Delete]
4	Molinero	2500	[Edit] [Delete]
5	Jefe de mantenimiento	2000	[Edit] [Delete]

**Figura N° 45** Pantalla de listado de cargos

**Fuente:** (Elaboración Propia)

En la Figura N° 45, se muestra la pantalla del listado de los cargos con los que cuenta cada sección de la cooperativa, en esta parte se puede realizar búsquedas, registro de nuevas producciones, modificación y eliminar un registro.

#	Nombre	Descripción	Acciones
1	Derecha entrada	avance antiguo	[Edit] [Delete]
2	Piso A	avance antiguo	[Edit] [Delete]
3	Recorte Rincon	avance nuevo llimani	[Edit] [Delete]
4	Piso Entrada	avance antiguo llimani	[Edit] [Delete]
5	Izquierda Entrada	avance antiguo	[Edit] [Delete]

**Figura N° 46** Pantalla de listado de secciones

**Fuente:** (Elaboración Propia)

En la Figura N° 46, se muestra la pantalla del listado de las secciones con los que cuenta la cooperativa, en esta parte se puede realizar búsquedas, registro de nuevas secciones, modificación y eliminar un registro.

#	Cedula	Nombre	Apellidos	Edad	Genero	Telefono	Cargo	Seccion
1	6976558	marin	duran quispe	25	masculino	75259656	Delegado	Derecha entrada
2	2345234	martin	churata mamani	30	masculino	79645214	Perforista	Derecha entrada
3	4312234	Manuel	Fernandez Quille	23	masculino	72134245	Molinerio	Derecha entrada
4	6542341	Roger	Fernandez Quille	25	masculino	64356235	Jefe de mantenimiento	Derecha entrada
5	5423456	Felipe	Mamani Mamani	40	masculino	79645214	Carrero	Derecha entrada

**Figura N° 47** Pantalla de reportes

**Fuente:** (Elaboración Propia)



En la Figura N° 45, se muestra la pantalla de modelo de reporte, en las cuales se podrá ver la información que se desea imprimir o tenerlo en formato Pdf.

### 3.5.2 Base de datos

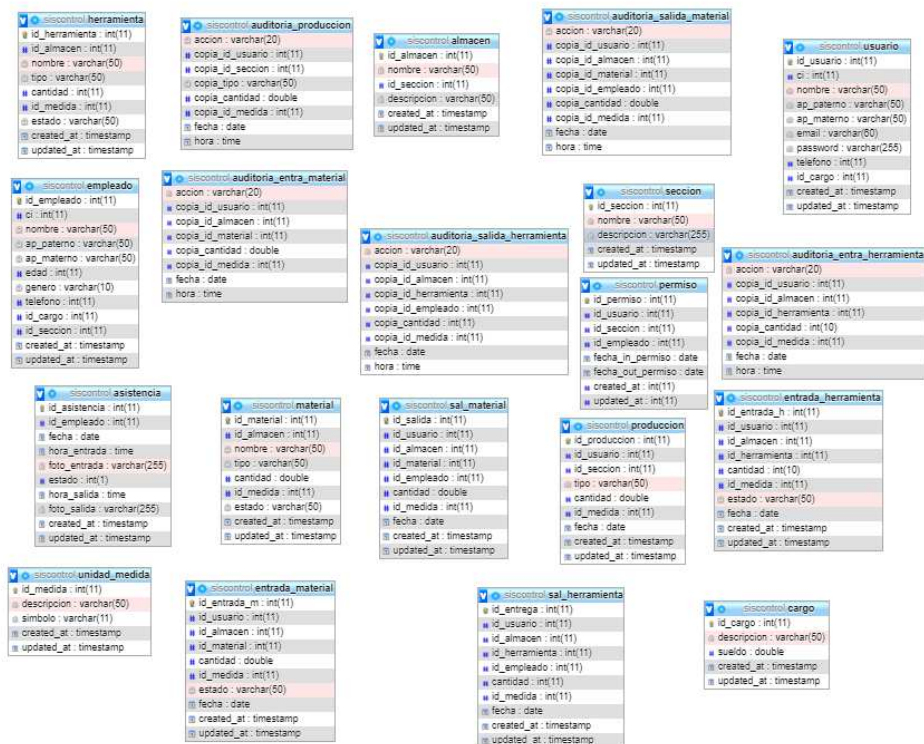


Figura N° 48 Base de datos del sistema

Fuente: (Elaboración Propia)

## 3.6 Fase de transición

### 3.6.1 Prueba de la caja negra

#### Inicio de sesión

Descripción del caso: El sistema enviara al usuario al panel de control una vez ingresado que el usuario ingrese sus datos correctamente caso contrario mostrara un error si no se introduce los datos correctamente.

Técnica de pruebas de caja negra: Requerimiento funcional.

Caso 1 Datos de entrada: Ingresar correo y contraseña correcta.

Resultado esperado(Salida): El sistema autentica y envía al usuario a la página principal.

Caso 2 Datos de entrada: Ingresar correo o contraseña incorrecto.

Resultado esperado(Salida): El sistema manda un mensaje de error, ya sea de que el correo no es válido o la contraseña es errónea.

### **Control de Asistencia**

Descripción del caso: El sistema recibe como dato el número de carnet del empleado y registra en la base de datos su hora de entrada y como también su hora de salida.

Técnica de pruebas de caja negra: Requerimiento funcional.

Caso 1 Datos de entrada: Ingreso de numero de carnet del empleado registrado con la opción registrar entrada.

Resultado(Salida): El sistema guarda los datos del empleado con la hora de entrada.

Caso 2 Datos de entrada: Ingreso de numero de carnet erróneo o no registrado con la opción de registrar entrada.

Resultado(salida): El sistema manda un mensaje de error “No se encuentra registrado o digito mal”.

Caso 3 Datos de entrada: Ingreso de numero de carnet del empleado registrado con la opción registrar salida.

Resultado(Salida): El sistema guarda la hora de salida del usuario.

Caso 4 Datos de entrada: Ingreso de numero de carnet del empleado registrado con la opción registrar salida sin haber registrado su entrada.

Resultado(Salida): El sistema manda un mensaje de error “No puedes marcar salida sin marcar entrada”.

Caso 5 Datos de entrada: Ingreso de numero de carnet del empleado registrado con la opción registrar entrada habiendo anteriormente registrado su entrada.

Resultado(Salida): El sistema manda un mensaje de error “No puedes marcar hoy”.

### **Registro de usuario**

Descripción del caso: El sistema agrega a un nuevo usuario, modifica y también elimina a un usuario.

Técnica de pruebas de caja negra: Requerimiento funcional.

Caso 1 Datos de entrada: Ingreso de datos del usuario en el formulario de registro.

Resultado(Salida): Guarda los datos en la base de datos.

Caso 2 Datos de entrada: Se despliega el formulario de modificación de usuario y se modifica un dato.

Resultado(Salida): Guarda los datos modificados.

Caso 3 Datos de entrada: Se acciona el botón de eliminar.

Resultado(Salida): despliega formulario de confirmación para eliminar y elimina los datos del usuario deseado.

## **Registro de Empleado**

Descripción del caso: El sistema agrega a un nuevo empleado, modifica y también elimina a un empleado.

Técnica de pruebas de caja negra: Requerimiento funcional.

Caso 1 Datos de entrada: Ingreso de datos del empleado en el formulario de registro.

Resultado(Salida): Guarda los datos en la base de datos.

Caso 2 Datos de entrada: Se despliega el formulario de modificación de empleado y se modifica uno o más datos.

Resultado(Salida): Guarda los datos modificados.

Caso 3 Datos de entrada: Se acciona el botón de eliminar.

Resultado(Salida): despliega formulario de confirmación para eliminar y elimina los datos del empleado deseado.

## **Registro de secciones**

Descripción del caso: El sistema agrega a una nueva sección, modifica y también elimina una sección.

Técnica de pruebas de caja negra: Requerimiento funcional.

Caso 1 Datos de entrada: Ingreso de datos de la sección en el formulario de registro.

Resultado(Salida): Guarda los datos en la base de datos.

Caso 2 Datos de entrada: Se despliega el formulario de modificar sección y se modifica uno o más datos.

Resultado(Salida): Guarda los datos modificados con éxito.

Caso 3 Datos de entrada: Se acciona el botón de eliminar.

Resultado(Salida): despliega formulario de confirmación para eliminar y elimina los datos de la sección deseada.

### **Ingreso de material**

Descripción del caso: El sistema puede agregar entradas de materiales a los almacenes de cada sección de la cooperativa y también elimina una entrada de material.

Técnica de pruebas de caja negra: Requerimiento funcional.

Caso 1 Datos de entrada: Se ingresa datos de la entrada de material en el formulario de registro.

Resultado(Salida): Guarda los datos en la base de datos con éxito además incrementa la cantidad de materiales existente en almacén.

Caso 2 Datos de entrada: Se acciona el botón de eliminar.

Resultado(Salida): despliega formulario de confirmación para eliminar y elimina los datos de la entrada de materiales deseado.

### **Salida de material**

Descripción del caso: El sistema puede agregar salida de materiales de los almacenes de cada sección de la cooperativa y también elimina una salida de material.

Técnica de pruebas de caja negra: Requerimiento funcional.

Caso 1 Datos de entrada: Se ingresa datos de la entrada de material en el formulario de registro.

Resultado(Salida): Guarda los datos en la base de datos con éxito además resta la cantidad de materiales existente en almacén.

Caso 2 Datos de entrada: Se acciona el botón de eliminar.

Resultado(Salida): despliega formulario de confirmación para eliminar y elimina los datos de la salida de materiales deseado.

### **Ingreso de herramientas**

Descripción del caso: El sistema puede agregar entradas de herramientas a los almacenes de cada sección de la cooperativa y también elimina una entrada de herramienta según el id de la herramienta.

Técnica de pruebas de caja negra: Requerimiento funcional.

Caso 1 Datos de entrada: Se ingresa datos de la entrada de la herramienta en el formulario de registro.

Resultado(Salida): Guarda los datos en la base de datos con éxito además incrementa la cantidad de herramientas existente en almacén.

Caso 2 Datos de entrada: Se acciona el botón de eliminar.

Resultado(Salida): despliega formulario de confirmación para eliminar y elimina los datos de la entrada de materiales deseado.

### **Salida de herramienta**

Descripción del caso: El sistema puede agregar información sobre la salida de herramientas de los almacenes de cada sección de la cooperativa y también elimina una de la salida de herramientas.

Técnica de pruebas de caja negra: Requerimiento funcional.

Caso 1 Datos de entrada: Se ingresa datos de la entrega de herramienta en el formulario de registro.

Resultado(Salida): Guarda los datos en la base de datos con éxito además resta la cantidad de herramientas existente en almacén.

Caso 2 Datos de entrada: Se acciona el botón de eliminar.

Resultado(Salida): despliega formulario de confirmación para eliminar y elimina los datos de la entrega de herramientas deseado.

### **Registro de Producción**

Descripción del caso: El sistema puede agregar Producciones diarias de cada sección de la cooperativa, modifica y también elimina los datos de un registro de producción.

Técnica de pruebas de caja negra: Requerimiento funcional.

Caso 1 Datos de entrada: Se ingresa datos de la entrada de la herramienta en el formulario de registro.

Resultado(Salida): Guarda los datos en la base de datos con éxito además incrementa la cantidad de herramientas existente en almacén.

Caso 2 Datos de entrada: Se despliega el formulario para modificar la producción y se modifica un dato o más.

Resultado(Salida): Guarda los datos modificados con éxito.

Caso 3 Datos de entrada: Se acciona el botón de eliminar.

Resultado(Salida): despliega formulario de confirmación para eliminar y elimina los datos del registro deseado.

### **Reportes**

Descripción del caso: El sistema puede generar reportes en formato Pdf.

Técnica de pruebas de caja negra: Requerimiento funcional.

Caso 1 Datos de entrada: Se acciona el botón de imprimir según los datos que dese obtener en el reporte.

Resultado(Salida): Descarga un archivo en formato Pdf con los datos seleccionados.



**CAPITULO IV**  
**CALIDAD Y SEGURIDAD**

## CAILIDAD Y SEGURIDAD

### 4.1 Métricas de calidad

Se hará la medición de calidad de software mediante la métrica ISO 9126, que establece cualquier componente de la calidad de software puede ser descrito en términos de una o más de seis características básicas las cuales son: funcionalidad, confiabilidad, usabilidad, mantenibilidad y portabilidad.

#### 4.1.1 Usabilidad

La usabilidad consiste de un conjunto de atributos que permite evaluar el esfuerzo necesario que deberá invertir el usuario para utilizar el sistema, es decir realizar una serie de preguntas que permiten ver cuán sencillo, fácil de aprender y manejar es para los usuarios. Esta comprensión por parte de los usuarios con relación al sistema evalúa los siguientes pasos:

- Entendimiento
- Aprendizaje
- Operabilidad
- Atracción
- Conformidad de uso

En la siguiente tabla se observa estos criterios en niveles de porcentajes a los que llego el sistema en cuanto a su comprensibilidad, para el usuario, y posteriormente se da el porcentaje

final de usabilidad del sistema. Se realizó una encuesta al usuario final sobre el manejo, la comprensión y la facilidad de aprender el sistema para medir la usabilidad según la siguiente tabla.

**Tabla 18**  
*Encuesta de usabilidad del sistema*

Preguntas	Respuestas		Porcentaje
	SI(1-10)	NO(1-10)	
¿El acceso al sistema es complicado?	0	10	100%
¿Las respuestas del sistema son comprensibles?	8	2	80%
¿Tiene alguna dificultad en realizar los procesos del sistema?	9	1	90%
¿La interfaz del sistema es amigable y entendible a su parecer?	8	2	80%
¿Son comprensibles y satisfactorios los datos que se muestran en el sistema?	9	1	90%
¿El sistema ayuda de manera eficiente en los procesos que se realizaban anteriormente?	10	0	100%
¿El sistema satisface los requerimientos de la empresa?	8	2	80%
¿El sistema es de fácil uso bajo su criterio?	10	0	100%
<b>Promedio total</b>			<b>90%</b>

**Fuente:** (Elaboración propia)

De acuerdo a los datos obtenidos en la tabla N° 18, se concluye que el sistema tiene una usabilidad del 90%. Esto indica que el usuario del tiene una conformidad hacia el sistema y que este podrá ayudar de alguna manera a solucionar los problemas de la cooperativa planteados anteriormente.

#### **4.1.2 Funcionalidad**

La funcionalidad del software se mide según la complejidad del mismo. La evaluación se valora según un conjunto de características y capacidades del programa. La funcionalidad no se puede medir directamente, por esta razón corresponde derivar medidas directas como es el punto función que cuantifica el tamaño y la complejidad del sistema en términos de las funciones del usuario. Se determina las cinco características del dominio de la información, teniendo en cuenta su cantidad y se define de la siguiente forma:

- Numero de entrada de usuario.
- Numero de salida de usuario.
- Numero de petición de usuario.
- Numero de archivos.
- Numero de interfaces externas.

Para realizar el cálculo de la cuenta total con los factores de ponderación se debe realizar la siguiente Tabla.

**Tabla 19**  
*Factor de ponderación*

Parámetros de medición	Cuenta	Factor	Total
Entrada de usuario	48	4	192
Salida de usuario	20	5	100
Petición de usuario	21	4	84
Numero de archivos	20	10	200
Numero de interfaces	2	7	14
<b>Cuenta total</b>			590

Fuente: (Elaboración en base a criterios: R. Pressman, 2007)

La Tabla N° 19, muestra la Cuenta Total que se obtiene de la sumatoria de los parámetros de medición.

**Tabla 20**  
*Valores de ajuste de complejidad*

		Valoración de factores						
Nro.	Factor de complejidad	0	1	2	3	4	5	Valor obtenido
		No influye	Incremental	Moderado	Medio	Significativo	Esencial	
1	¿Requiere el Sistema copias de Seguridad y de recuperación fiables?						X	5
2	¿Se requiere comunicación de datos?					X		4
3	¿Existen funciones de procesamiento distribuido?					X		4

4	¿Es crítico el rendimiento?				X			3
5	¿Será ejecutado el sistema en un entorno operativo existente y fuertemente utilizado?						X	5
6	¿Requiere el sistema entrada de datos interactiva?					X		4
7	¿Requiere la entrada de datos interactiva que las transacciones de entrada se lleven a cabo sobre múltiples pantallas u operaciones?				X			3
8	¿Se actualizan los archivos maestros de forma interactiva?					X		4
9	¿Son complejas las entradas, las salidas, los archivos o las peticiones?				X			3
10	¿Es complejo el procesamiento interno?				X			3
11	¿Se ha diseñado el código para ser reutilizado?						X	5
12	¿Están incluidas en el diseño la conversión y la instalación?				X			3
13	¿Se ha diseñado el sistema para soportar múltiples instalaciones en diferentes organizaciones?				X			3

14	¿Se ha diseñado la aplicación para facilitar los cambios y para ser fácilmente utilizada por el usuario?							X	5
<b>TOTAL</b>									54

Fuente: (Elaboración en base de criterios de: S. Pressman, 2007)

Para calcular los puntos de función se utiliza la siguiente ecuación.

$$\text{Punto de Función} = \text{Cuenta Total} * [0.65 + 0.01 * \sum Fi]$$

Dónde: Punto Función: Medida de funcionalidad entregada a la aplicación como valor de normalización. Cuenta Total: Es el nivel de complejidad del sistema respecto al usuario.

Cuenta Total: Es el nivel de complejidad del sistema respecto al usuario.

$[0.65 + 0.01 * \sum Fi]$ : Ajuste de la complejidad según el dominio de la información de software, en donde  $\sum Fi$  son los valores de complejidad.

**0,01**: Factor de conversión, es decir, un error del 1%.

**0,65**: Valor mínimo del ajuste de la complejidad.

Se tiene:

$$\text{Punto de Función} = 590 * [0,65 + 0,01 * 54]$$

$$\text{Punto de Función} = 702.1$$

Para hallar el punto función ideal al 100% de los factores sería 70:

$$PF=590 * (0,65 + 0,01 * 70)$$

$$\text{PF Ideal} = 796.5$$

Calculando del % de funcionalidad real:

$$PF \text{ Real} = (702.1 / 796.5) * 100 = 88.2$$

Por tanto:

Entonces la funcionalidad del sistema es un 88% esto quiere decir que el sistema tiene un 88% que funciona sin riesgo de fallo con operatividad constante y 12% de colapso de sistema.

### 4.1.3 Confiabilidad

La confiabilidad permite evaluar la relación entre el nivel de funcionalidad y la cantidad de recursos usados, es decir, representa el tiempo que el software está disponible para su uso, la misma se calcula utilizando la privacidad de que un sistema presente fallas:

- Comportamiento con respecto al tiempo: Atributos de software relativos a los tiempos de respuesta y de procesamiento de los datos.
- Comportamiento con respecto a Recursos: Atributos software relativo a la cantidad de recursos usados y la duración de su uso en la realización de funciones.

La función a continuación muestra el nivel de confiabilidad del sistema:

$$F(t) = (\text{Funcionalidad}) * e^{-\lambda t}$$

Se ve el trabajo hasta que se observa un fallo en un instante t, la función es la siguiente:

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

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



Dónde:

Funcionalidad = 0.88

$\lambda = 0.14$  (1 error cada 7 ejecuciones)

Tomemos un tiempo  $t$  de 12 meses

Ahora hallando la confiabilidad del sistema:

$$F(12) = (0.88) * e^{-0.14*12}$$

$$F(12) = 0.164$$

Remplazando en la fórmula de no hallar una falla se tiene:

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

$$P(T > t) = 1 - 0.164$$

$$P(T > t) = 0.84$$

Con este resultado podemos decir que la probabilidad que el sistema no presente fallas es de 0.84 y que presente fallas es del 0.16.

Con este resultado se concluye que la aplicación tiene un grado de confiabilidad del 84% durante los próximos 12 meses.

#### **4.1.4 Mantenibilidad**

La Mantenibilidad se refiere a los atributos que permiten medir el esfuerzo necesario para realizar modificaciones al software, ya sea por la corrección de errores o por el incremento

de funcionalidad. Para hallar mantenibilidad del sistema se utiliza el índice de madurez de software (IMS), que proporciona una indicación de la estabilidad de un producto de software.

Para calcular el índice hacen falta una serie de medidas anteriores:

$M_t$  = número de módulos en la versión actual.

$F_m$  = número de módulos en la versión actual que han sido modificados.

$F_a$  = número de módulos en la versión actual que han sido añadidos.

$F_e$  = número de módulos de la versión anterior que se han eliminado en la versión actual.

A partir de estas, el IMS se calcula de la siguiente forma:

$$IMS = \frac{[M_t - (F_a + F_m + F_e)]}{M_t}$$

$$IMS = \frac{[15 - (1 + 0 + 0)]}{15}$$

$$IMS = 0.93$$

### **Mantenibilidad 93%**

La interpretación a este resultado establece un 93%, lo que indica que no requiere de mantenimiento inmediatamente.

#### **4.1.5 Portabilidad**

Un sistema es portable, si el costo de transportar y adaptar la aplicación aun nuevo entorno es menor al costo de rediseñar el sistema para dicho ambiente. La relación que permite determinar el grado de portabilidad es la siguiente:

- Adaptabilidad. Capacidad del producto que le permite ser adaptado de forma efectiva y eficiente a diferentes entornos determinados de hardware, software, operacionales o de uso.
- Capacidad para ser instalado. Facilidad con la que el producto se puede instalar y/o desinstalar de forma exitosa en un determinado entorno.
- Capacidad para ser reemplazado. Capacidad del producto para ser utilizado en lugar de otro producto software determinado con el mismo propósito y en el mismo entorno.

Para el cálculo de la portabilidad se tomó en cuenta la siguiente tabla que contiene las características que se mencionaron anteriormente.

**Tabla 21**  
*Portabilidad*

Factor de portabilidad	Valor %
Puede ser trasferido de un entorno a otro	80
Se puede adaptar a otros ambientes con facilidad (Instituciones similares)	100
Es capaz de reemplazar a una aplicación similar	80
TOTAL	86.7

Fuente: (Elaboración propia)

**Potabilidad 87 %**

Viendo la tabla la portabilidad del sistema será del 87% mientras se cumpla con los requisitos necesarios a cumplir.

## **4.2 Seguridad**

La seguridad es la capacidad de protección de la información y los datos de manera que las personas no autorizadas no puedan leerlas y/o modificarlas.

### **4.2.1 Seguridad de software**

Laravel es el framework de Php más popular hoy en día, pues aporta grandes ventajas ante sus competidores más cercanos, pues una de sus características más relevantes es su nivel de seguridad que le da (por defecto) a las aplicaciones construidas bajo este framework.

#### **Protección CSFR**

Laravel facilita la protección del sistema contra ataques de falsificación de solicitudes entre sitios (CSRF – Cross-site Request Forgery).

Laravel genera automáticamente un ‘token’ CSRF para cada sesión activa de usuario, administrada por la aplicación. Este token se utiliza para verificar que el usuario autenticado es el que realiza las solicitudes a la aplicación.

#### **HTTP Middleware**

HTTP Middleware provee un conveniente mecanismo para filtrar peticiones HTTP que ingresan a la aplicación. Por ejemplo, Laravel incluye un middleware que verifica que el usuario de la aplicación está autenticado. Si el usuario no está autenticado, el middleware redirigirá al usuario a la pantalla de inicio de sesión. Por otro lado, si el usuario está autenticado, el middleware permitirá que la petición continúe en la aplicación.

Se implementa un middleware para restringir el acceso por peticiones HTTP a las rutas que solo el personal administrativo puede acceder, así filtrar el acceso al área administrativa por parte de otros usuarios del sistema.

### **Encriptación**

Laravel cuenta con una llave (key) por cada aplicación, la misma es utilizada para la encriptación y las sesiones manejadas por la aplicación. Dicho key es un string aleatorio compuesto de 32 o 64 caracteres.

### **Hashing**

Hash de Laravel proporciona un hashing seguro a través de la función Bcrypt para almacenar las contraseñas de los usuarios. Esto a través del controlador LoginController que se incluye con las aplicaciones de Laravel.

Se implementa la función Bcrypt para realizar el hash de las contraseñas de los usuarios y el inicio de sesión, por su uso conveniente en la aplicación.

**CAPITULO V**  
**ANALISIS DE COSTOS**

**CAPITULO V****ANALISIS DE COSTOS**

Para determinar el Costo Total del proyecto se tomará en cuenta el costo de software, costo de implementación de la aplicación y elaboración del proyecto.

**5.1 Costo del Software desarrollado**

Para determinar el costo de la aplicación se usa el modelo COCOMO II orientado en los puntos de función. Para calcular el esfuerzo, necesitamos hallar la variable KLDC (Kilo-líneas de código).

En este proyecto se implementa 16857 líneas de código en el lenguaje PHP. Aplicando conversiones tenemos:

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

$$KLDC = \frac{16857}{1000}$$

$$KLDC = 16.857$$

Así pues, en nuestro caso el tipo semi - acoplado será el más apropiado ya que el número de líneas de código es menor a 50 KLDC, y además el proyecto tiene una complejidad media.

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

**Tabla 22**  
*Modelo de cocomo*

<b>MODO</b>	<b>a</b>	<b>b</b>	<b>c</b>	<b>d</b>
<b>ORGÁNICO</b>	3.2	1.05	2.5	0.38
<b>SEMI-ACOPLADO</b>	3.0	1.12	2.5	0.35
<b>EMPOTRADO</b>	2.8	1.20	2.5	0.32

Fuente: (S. Pressman, 2007)

Para hallar los valores de FAE, se utilizará la Tabla N° 23.

**Tabla 23**  
*Conductores de costo*

<b>Atributos</b>	<b>Valor</b>					
	Muy Bajo	Bajo	Nominal	Alto	Muy Alto	Extra Alto
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 del producto	0.70	0.85	1.00	1.15	1.30	1.65
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	-
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	1.17	1.00	0.95	-	-
Técnicas actualizadas de programación	1.24	1.10	1.00	0.91	0.82	-
Utilización de herramientas software	1.24	1.10	1.00	0.91	0.83	-
Restricciones de tiempo de desarrollo	1.24	1.08	1.00	1.04	1.10	-
<b>TOTAL</b>					0.61	

**Fuente:** (Calculo de costos de producción de software, Castillo, 2017)

Por tanto, nuestro factor de ajuste será el resultado de la multiplicación de los valores evaluados en la Tabla N° 23.

Por lo cual, el factor de ajuste será:  $FAE = 0.61$

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

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

$$E = 2.4 * (16.857)^{1.05} * 0.61$$

$$E = 28.42 \text{ (Personas/Mes)}$$

Calculo tiempo de desarrollo:

$$T = c \text{ Esfuerzo}^d = 2.5 * (28.42)^{0.35}$$

$$T = 8.07 \text{ [meses]}$$

Personal promedio:

$$P = E/T \quad P=28.42 /8.07$$

$$P= 3.53 \text{ personas}$$

Según este resultado será necesario un equipo de 3 personas trabajando alrededor de 8 meses.

Al tener los datos definidos, se realiza el supuesto estimación de costo

$$\text{La hora por trabajar} = 2.5\$$$

$$\text{día} = 2.5 * 8 = 20\$$$

$$\text{Semana} = 20 * 5 = 100\$$$

$$\text{Al mes } 100 * 4 \text{ semanas} = 400 * 3 \text{ personas} = 1200 \$/\text{mes.}$$

$$\text{Calculamos } 1200 \$ * 8 \text{ meses} = 9600 \$, \text{ el costo del producto elaborado en 8 meses.}$$

**CAPITULO VI**

**CONCLUSIONES**

**Y**

**RECOMENDACIONES**

## CAPITULO VI

### CONCLUSIONES Y RECOMENDACIONES

En este capítulo se dará a conocer las conclusiones y recomendaciones pertinentes del proyecto de grado.

#### 6.1 Conclusiones

Una vez finalizado el desarrollo del “SISTEMA DE INFORMACIÓN WEB PARA EL CONTROL DE PERSONAL RESPECTO A LA PRODUCCIÓN Y ALMACENES CASO: COOPERATIVA MINERA BOLSA NEGRA” para la cooperativa minera Bolsa Negra, se ha logrado alcanzar el objetivo principal planteado, bajo los requerimientos de la cooperativa. Tomando en cuenta los objetivos planteados se llega a las siguientes conclusiones:

- Se ha logrado diseñar una interfaz amigable y de fácil manipulación para el usuario.
- El sistema agiliza los tiempos de acceso a la información.
- Se logró automatizar el control de asistencia del personal.
- Se logró mejorar el control del personal de la cooperativa.
- Se tiene un mejor control de entradas y salidas en herramientas y materiales.
- Se tiene un registro con los datos necesarios sobre la disponibilidad en almacenes.
- Se tiene un registro con los datos necesarios de la producción de oro o wólfram por secciones con la que cuenta la cooperativa.
- Se logró disminuir los tiempos en la generación de reportes.

De esta forma, se alcanzó el objetivo general de lograr optimizar los procesos de control de asistencia, control de inventarios en almacenes y del registro de la producción de manera que la información sea de fácil acceso, y no exista los problemas anteriores de pérdida de información.

## **6.2 Recomendaciones**

En base a las observaciones realizadas durante la prueba del sistema se elaboraron las siguientes recomendaciones.

- Se recomienda al usuario cambiar continuamente su contraseña o clave de seguridad del sistema y proteger el acceso de personas ajenas y malintencionadas al sistema.
- Mantener un control acerca del equipo que hace de servidor físico en el ambiente.
- Para resguardar la información, el administrador del sistema debe realizar copias de seguridad de la base de datos.
- Para posteriores trabajos se recomienda utilizar herramientas similares para la construcción de software.

## BIBLIOGRAFIA

Ludwig von Bertalanffy (1968). Teoría General de Sistemas.

Tamayo Alzate, Alonso (1999). Teoría General de Sistemas.

Roger Pressman (2013) Ingeniería del Software. Un enfoque práctico, Séptima Edición.

José M. Sánchez (2015) Pruebas de Software. Fundamentos y técnicas

Zea M, Honores J, Rivas W (2015) Fundamentos de base de datos

A. Gómez, María del C.López (2010) - Cocomo - Un Modelo De Estimación De  
Proyectos De Software

Larman, C. (1999) UML y patrones, introducción al análisis y diseño orientado a  
objetos.

Rumbaugh, Jacobson, Booch (2007) El lenguaje unificado de modelado. Manual de  
referencia.

Anyer J. Martinez (2011) Lenguaje modelado unificado (Objeto informática).

Daniel Minguez S (2012) Metodologías para el Desarrollo de Aplicaciones Web: UWE

Michael Arias Chaves (2006) La ingeniería de requerimientos y su importancia en el  
desarrollo de proyectos de software

Lizka Johany H (2003) Ingeniería de Requerimientos

## WEBGRAFIA

José A. Senso (2016). Los sistemas de información basados en la web  
<https://blogs.ugr.es/tecweb/los-sistemas-informacion-basados-la-web/>

Uriel Hernández (2015) MVC (Model, View, Controller) Explicado.  
<https://codigofacilito.com/articulos/mvc-model-view-controller-explicado>

Robert McCool. (1995) Servidor HTTP Apache  
[https://httpd.apache.org/ABOUT\\_APACHE.html](https://httpd.apache.org/ABOUT_APACHE.html)

Carlos Villagómez (2017) Ciclo de vida del Software <https://es.ccm.net/contents/223-ciclo-de-vida-del-software>

Reynier (2017) Almacén <https://www.ecured.cu/Almac%C3%A9n>

Miriam Quiroa (2007) Producción  
<https://economipedia.com/definiciones/produccion.html>

Kimaldi (2017) Control personal  
[https://www.kimaldi.com/blog/control\\_de\\_acceso\\_y\\_presencia/control\\_de\\_personal/](https://www.kimaldi.com/blog/control_de_acceso_y_presencia/control_de_personal/)

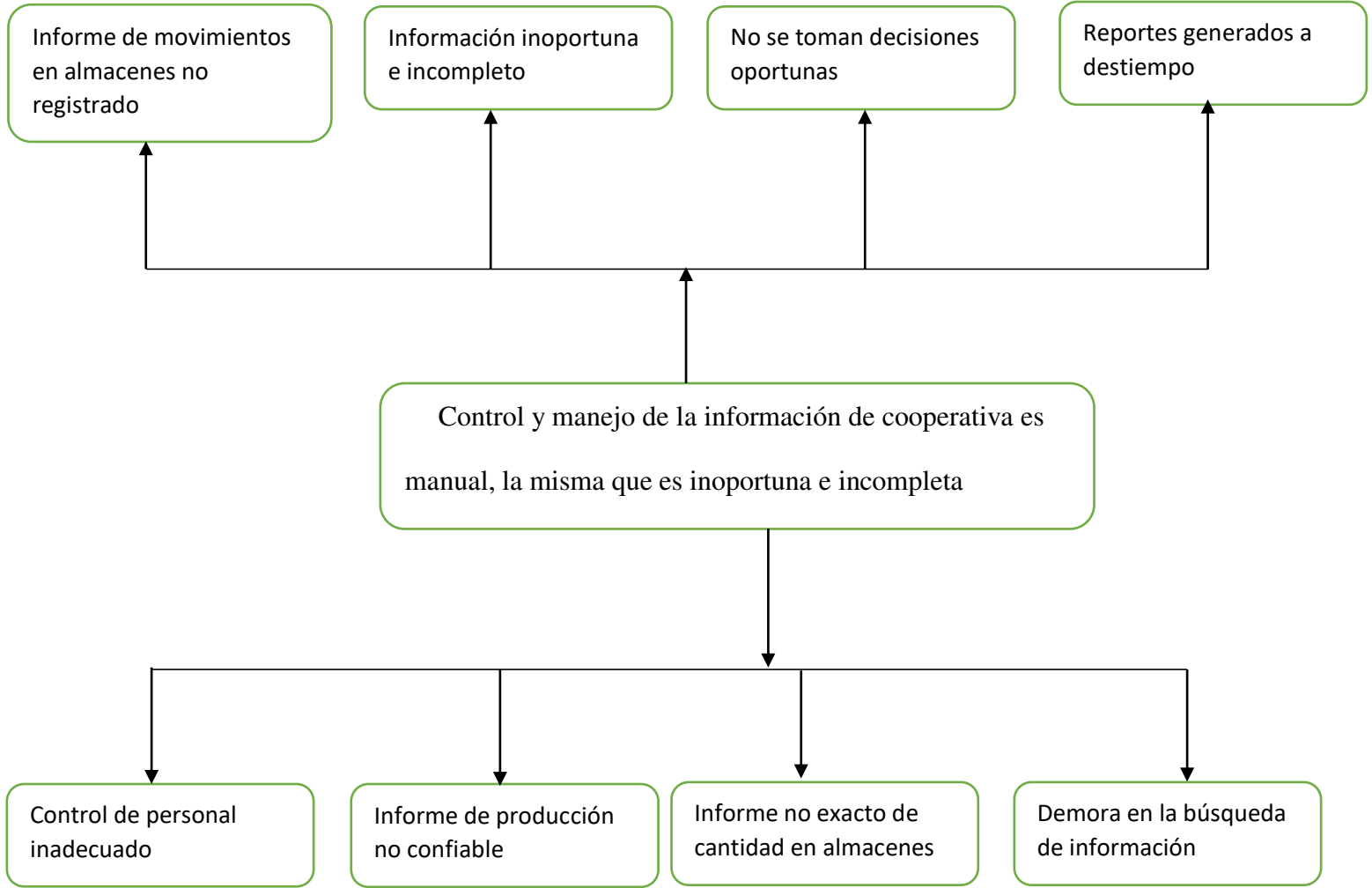
# ANEXOS



## **ANEXOS “A”**

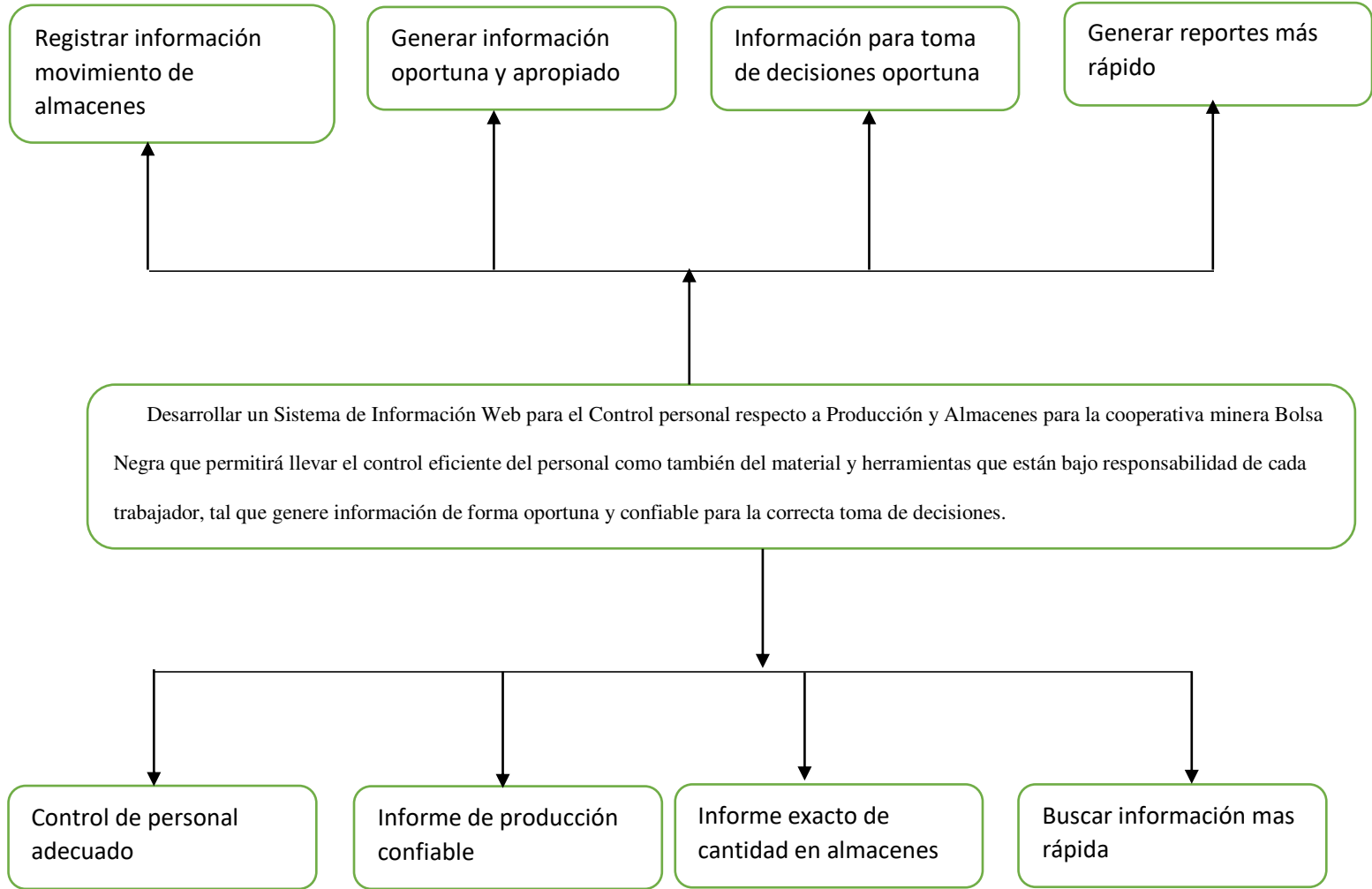
### **ARBOL DE PROBLEMAS**

Árbol de problemas



## **ANEXOS “B”**

### **ARBOL DE PROBLEMAS**



# **ANEXOS “C”**

## **MANUAL DE USUARIO**



**Cooperativa Minera**

**Bolsa Negra**

# **MANUAL DE USUARIO**

**SISTEMA DE INFORMACION WEB PARA EL CONTROL DE PERSONAL**

**RESPECTO A LA PRODUCCION Y ALMACENES**

**CASO: COOPERATIVA MINERA BOLSA NEGRA**

**MARIN DURAN QUISPE**

**LA PAZ- BOLIVIA**

## CONTENIDO

1. Introducción.....	4
2. Ingreso Al Sistema.....	4
3. Control de asistencia.....	4
4. Inicio de sesión .....	5
5. Menú almacén.....	6
6. Menú movimiento de almacén.....	9
7. Menú Personal.....	14
8. Menú Reporte.....	19

## 1 Introducción

En el presente manual se explicarán lo necesario para poder utilizar el Sistema de Información Web para el Control de Personal Respecto a la Producción y Almacenes.

## 2 Ingreso al sistema

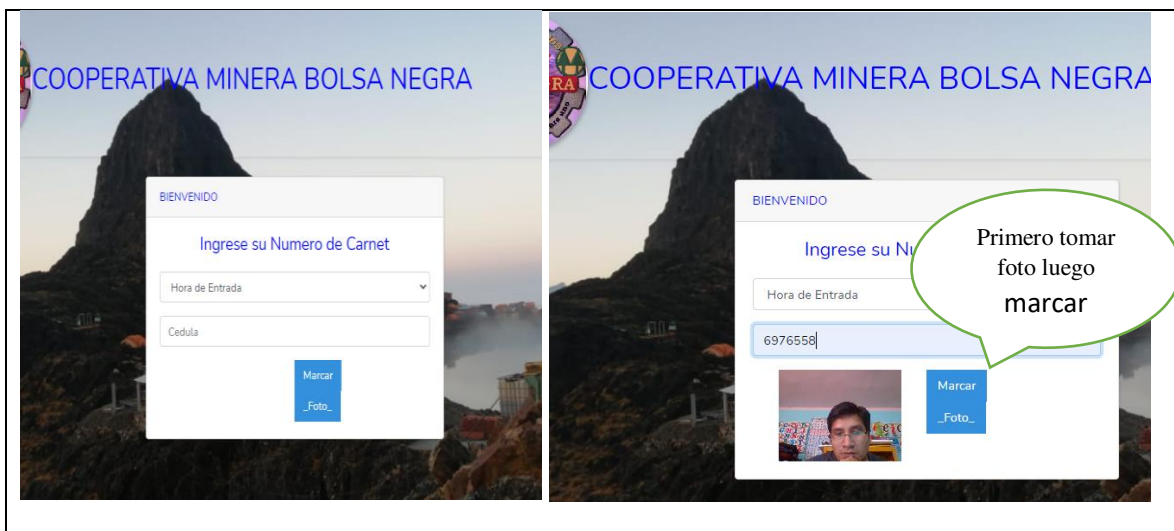
Para poder utilizar el sistema de control de inventarios, se debe acceder al sistema a través de estos navegadores de Internet:

Mozilla Firefox. 

Google Chrome. 

U otros navegadores compatibles con el sistema de información web, para ingresar al sistema se debe digitar la dirección web <https://siscontrol.com/>

## 3 Control de Asistencia

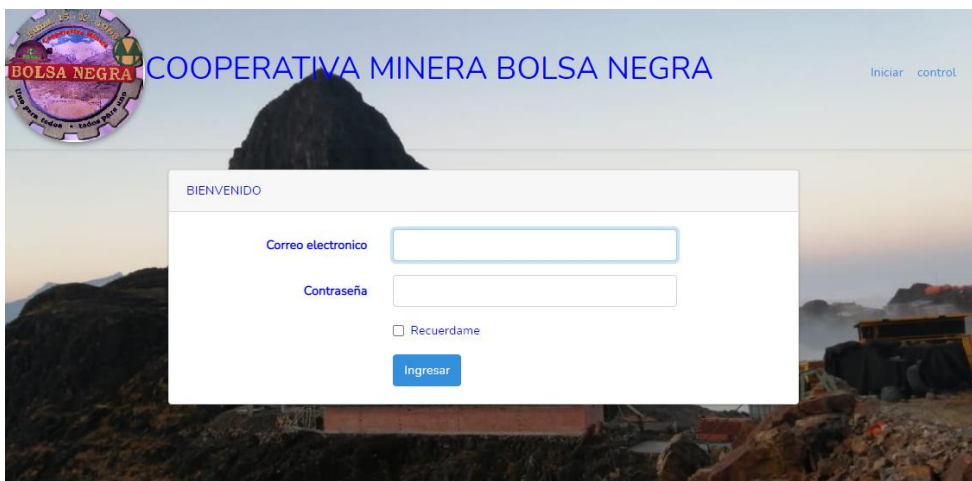


**Figura 1** Pantalla control de asistencia



La primera pantalla que se mostrara una vez ingresado la dirección web será la de control de asistencia, donde podrá el trabajador seleccionar si desea marcar como entrada o salida, una vez seleccionado la salida o entrada el trabador deberá ingresar su número de carnet de identidad, luego se activara una cámara de foto, el trabajador deberá primero tomar la foto luego presionar el botón de marcar como se ve en la figura 1.

#### 4 Inicio de sesión



*Figura 2 Pantalla de inicio de sesión*

Para ir a la pantalla de inicio de sesión deberá presionar en el botón iniciar que se encuentra al de del botón control, luego en la pantalla de inicio de sesión deberá ingresar el usuario su correo y su contraseña como se ve en la figura 2.

Una vez que el usuario ingresa sus datos correctamente para iniciar sesión, ingresa a la pantalla principal de control, donde se muestra la barra de navegación la cual servirá para seleccionar los distintos módulos, también se muestra la cantidad de trabajadores, trabajadores activos y numero de secciones como podrá ver en la figura 3.

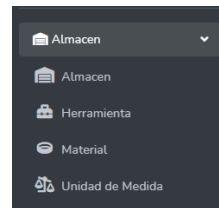


**Figura 3** Pantalla de Control

## 5 Menú almacén

En el menú almacén se encuentra los sub-menús de:

Almacén, Herramienta, Material y Unidad de medida







Para realizar un registro u otra acción de material debe dirigirse a Almacén luego a Material.

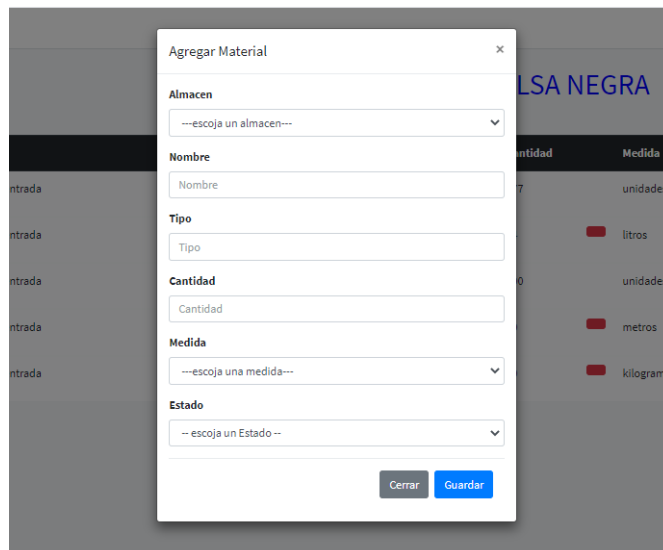
#	Almacen	Nombre	Tipo	Cantidad	Medida	Estado	Acciones
1	almacen Derecha entrada	dinamita	explosivo	377	unidades	usable	[Edit] [Delete]
2	almacen Derecha entrada	gasolina	combustible	74	litros	usable	[Edit] [Delete]
3	almacen Derecha entrada	fulminante	explosivo	200	unidades	usable	[Edit] [Delete]
4	almacen Derecha entrada	Mecha(guia)	explosivo	70	metros	usable	[Edit] [Delete]
5	almacen Derecha entrada	Nitrato de amonio	quimico	20	kilogramos	usable	[Edit] [Delete]

**Figura 4** Pantalla listado Material

En la figura 4 podrá realizar las siguientes acciones:

- **Agregar**  : muestra una ventana en la cual podrá registrar un nuevo material.
- **Modificar**  : Muestra una ventana donde puede modificar los datos de un material ya registrado.
- **Eliminar**  : puede eliminar el registro de material seleccionado.
- **Buscar**  : Permite realizar búsquedas de material en base a los registros existentes.

En la figura 5 se muestra el formulario para registrar un nuevo material.







**Figura 5** *Formulario de registro de nuevo material*

Para realizar un registro u otra acción de herramienta debe dirigirse a Almacén luego a Herramienta.

#	Almacen	Nombre	Tipo	Cantidad	Medida	Estado	Acciones
1	almacen Derecha entrada	Pala	manual	31	unidades	nuevo	[Modificar] [Eliminar]
2	almacen Derecha entrada	Carretilla	manual	7	unidades	nuevo	[Modificar] [Eliminar]
3	almacen Derecha entrada	Carros	manual	0	unidades	nuevo	[Modificar] [Eliminar]
4	almacen Derecha entrada	Combo	manual	10	unidades	nuevo	[Modificar] [Eliminar]
5	almacen Derecha entrada	Picota	manual	7	unidades	nuevo	[Modificar] [Eliminar]

**Figura 6** Pantalla lista de herramientas

En la figura 6 podrá realizar las siguientes acciones:

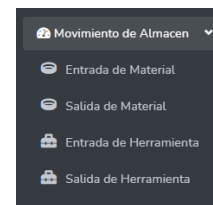
- **Agregar**  : muestra una ventana en la cual podrá registrar una nueva herramienta.
- **Modificar**  : Muestra una ventana donde puede modificar los datos de una herramienta ya registrado.
- **Eliminar**  : puede eliminar el registro de Herramienta seleccionado.
- **Buscar**  : Permite realizar búsquedas de material en base a los registros existentes.

En la figura 7 se muestra el formulario para registrar un nuevo material.

**Figura 7** Formulario de registro de nueva herramienta

## 6 Menú Movimiento de almacén

En el menú de movimiento de almacén se encuentra los sub-menús de: Entrada de material, Salida de Material, Entrada de Herramienta, Salida de Herramienta.






Para realizar un registro u otra acción de Entrada de Material debe dirigirse a Moviente de Almacén luego a Entrada Material.

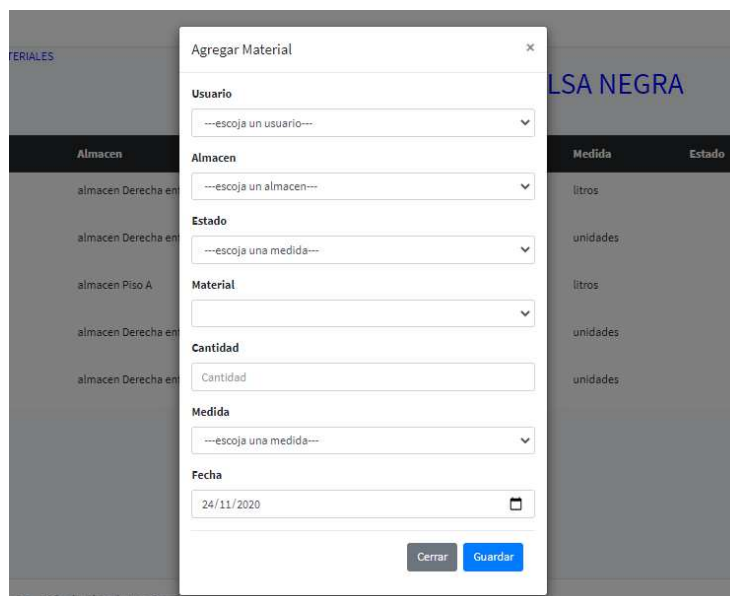
#	Usuario	Almacen	Material	Cantidad	Medida	Estado	Fecha	Acciones
1	marín duran	almacen Derecha entrada	gasolina	20	litros		2020-10-29	
2	marín duran	almacen Derecha entrada	dinamita	33	unidades		2020-10-29	
3	marín duran	almacen Piso A	gasolina	33	litros		2020-10-29	
4	marín duran	almacen Derecha entrada	dinamita	20	unidades		2020-11-04	
5	marín duran	almacen Derecha entrada	dinamita	200	unidades		2020-11-07	

**Figura 8** Pantalla lista de entrada de material

En la figura 8 podrá realizar las siguientes acciones:

- **Agregar**  : muestra una ventana en la cual podrá registrar una nueva entrada de material.
- **Eliminar**  : puede eliminar el registro de una entrada de material seleccionado.
- **Buscar**  : Permite realizar búsquedas de entrada de materiales en base a los registros existentes.

En la figura 9 se muestra el formulario para registrar una nueva entrada de material.





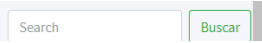
**Figura 9** *Formulario de registro de entrada material*

Para realizar un registro u otra acción de salida de material debe dirigirse a Movimiento de almacén, luego a Salida de material.

#	Usuario	Almacen	Material	Empleado	Cantidad	Medida	Fecha	Acciones
1	marin duran	almacen Derecha entrada	gasolina	martin churata	6	litros	2020-10-29	
2	marin duran	almacen Piso A	dinamita	Macos Apaza	20	unidades	2020-10-29	
3	julio quispe	almacen Piso A	dinamita	Pedro Mamani	30	unidades	2020-11-04	
4	marin duran	almacen Derecha entrada	dinamita	Julio Quiroz	33	unidades	2020-11-07	
5	marin duran	almacen Derecha entrada	gasolina	martin churata	80	litros	2020-11-18	

**Figura 10** Pantalla de lista de salida de materiales

En la figura 10 podrá realizar las siguientes acciones:

- **Agregar**  : muestra una ventana en la cual podrá registrar una nueva salida de material.
- **Eliminar**  : puede eliminar el registro de una salida de material seleccionado.
- **Buscar**  : Permite realizar búsquedas de salida de materiales en base a los registros existentes.

En la figura 11 se muestra el formulario para registrar una nueva salida de material.




**Figura 11** Formulario de registro de nueva salida de material

Para realizar un registro u otra acción de entrada de herramienta debe dirigirse a Movimiento de almacén, luego a Salida de Herramienta.

#	Usuario	Almacen	Herramienta	Cantidad	Medida	Estado	Fecha	Acciones
1	marín duran	almacen Derecha entrada	pala	2	unidades	reparable	2020-11-12	
2	marín duran	almacen Derecha entrada	Carros	4	unidades	nuevo	2020-11-12	
3	marín duran	almacen Derecha entrada	pala	1	unidades	reparable	2020-11-18	
4	marín duran	almacen Derecha entrada	Pala	20	unidades	nuevo	2020-11-18	
5	marín duran	almacen Derecha entrada	Combo	4	unidades	nuevo	2020-11-21	

*Figura 12 Pantalla de listado de entrada de herramienta*

En la figura 12 podrá realizar las siguientes acciones:

- **Agregar**  : muestra una ventana en la cual podrá registrar una nueva entrada de herramienta.
- **Eliminar**  : puede eliminar el registro de una entrada de herramienta seleccionado.
- **Buscar**   : Permite realizar búsquedas de entrada de herramientas en base a los registros existentes.

En la figura 13 se muestra el formulario para registrar una nueva entrada de herramienta.




**Figura 13** *Formulario de nueva entrada de herramienta*



Para realizar un registro u otra acción de salida de material debe dirigirse a Movimiento de almacén, luego a Salida de herramienta.

#	Usuario	Almacen	Herramienta	Empleado	Cantidad	Medida	Fecha	Acciones
1	marín duran	almacen Derecha entrada	Pala	Manuel Fernandez	2	unidades	2020-10-29	
2	marín duran	almacen Piso A	carretilla	Pedro Mamani	1	unidades	2020-10-29	
3	marín duran	almacen Derecha entrada	Carros	Julio Quiroz	4	unidades	2020-11-12	
4	marín duran	almacen Derecha entrada	Carros	Julio Quiroz	3	unidades	2020-11-12	
5	marín duran	almacen Derecha entrada	Pala	Felipe Mamani	6	unidades	2020-11-22	

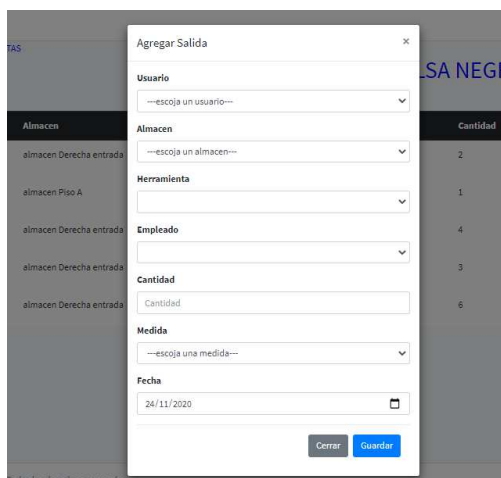
**Figura 14** *Pantalla de listado salida de herramientas*

En la figura 14 podrá realizar las siguientes acciones:

- **Agregar**  : muestra una ventana en la cual podrá registrar una nueva salida de herramientas.

- **Eliminar** : puede eliminar el registro de una salida de herramientas seleccionado.
- **Buscar** : Permite realizar búsquedas de salida de herramientas en base a los registros existentes.

En la figura 15 se muestra el formulario para registrar una nueva salida de herramienta.



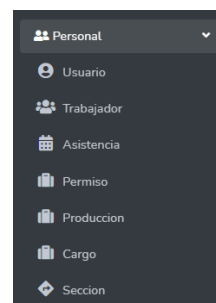
**Figura 15** Formulario de registro de nueva salida de herramientas

## 7 Menú personal

En el menú Personal se encuentra los sub-menús de:

Usuario, Trabajador, Asistencia, Permiso, Producción

Cargo, Sección.



Para realizar un registro u otra acción de Trabajador debe dirigirse a Personal, a Trabajador.

COOPERATIVA MINERA BOLSA NEGRA

LISTA DE EMPLEADO

Agregar

Search Buscar

#	Cedula	Nombre	Apellidos	Edad	Genero	Telefono	Cargo	Seccion	Acciones
1	6976558	marin	durán quispe	25	masculino	75259656	Delegado	Derecha entrada	
2	2345234	martin	churata mamani	30	masculino	79645214	Perforista	Derecha entrada	
3	4312234	Manuel	Fernandez Quille	23	masculino	72134245	Molinero	Derecha entrada	
4	6542341	Roger	Fernandez Quille	25	masculino	64356235	Jefe de mantenimiento	Derecha entrada	
5	5423456	Felipe	Mamani Mamani	40	masculino	79645214	Carrero	Derecha entrada	

< 1 2 >

**Figura 16** Pantalla de listado de trabajador

En la figura 16 podrá realizar las siguientes acciones:

- **Agregar** : Muestra una ventana en la cual podrá registrar un nuevo trabajador.
- **Modificar** : Muestra una ventana donde puede modificar los datos de un trabajador ya registrado.
- **Eliminar** : puede eliminar el registro de Trabajador seleccionado.
- **Buscar** : Permite realizar búsquedas de Empleados en base a los registros existentes.

En la figura 17 se muestra el formulario para registrar un nuevo empleado.


**Figura 17** Formulario de registro de Trabajador






Para ver el listado de Asistencia de trabajador debe dirigirse a Personal, luego a Asistencia.

#	Cedula	Nombre	Fecha	Hora de entrada	foto de entrada	Hora de salida	foto de salida	Acciones
1	6976558	durán marín	2020-11-22	10:15:00		11:48:00		
2	2345612	Apaza Macos	2020-11-21	17:41:00		17:45:00		
3	6976558	durán marín	2020-11-19	14:52:00		14:54:00		
4	2345612	Apaza Macos	2020-11-18	16:51:00		16:54:00		
5	4312234	Fernandez Manuel	2020-11-18	15:05:00		16:55:00		

**Figura 18** Pantalla de listado de asistencia

En la figura 18 podrá realizar las siguientes acciones:

- **Agregar**  : Muestra una ventana en la cual podrá registrar una nueva asistencia.

- **Modificar**  : Muestra una ventana donde puede modificar los datos de una asistencia de trabajador ya registrado.
- **Eliminar**  : puede eliminar el registro de Asistencia de Trabajador seleccionado.
- **Buscar**      : Permite realizar búsquedas de asistencia de trabajadores en base a los registros existentes.





COOPERATIVA MINERA BOLSA NEGRA


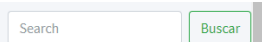

#	Usuario	Seccion	Trabajador	Inicio de Permiso	Fin de Permiso	Acciones
1	marin duran	Derecha entrada	martin churata	2020-11-11	2020-11-12	 
2	marin duran	Derecha entrada	Felipe Mamani	2020-11-19	2020-11-20	 
3	marin duran	Derecha entrada	Manuel Fernandez	2020-11-23	2020-11-23	 

Copyright © 2020 MarinDQ . Todos los derechos reservados. Cooperativa Minera Bolsa Negra

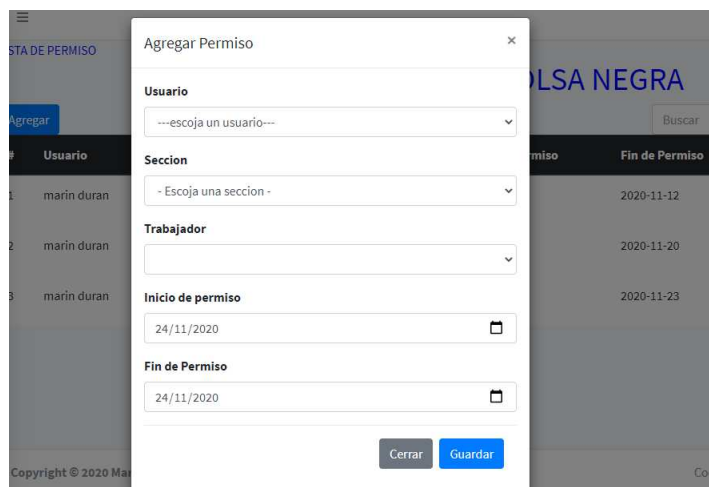
**Figura 19** Pantalla de listado de permisos

En la figura 19 podrá realizar las siguientes acciones:

- **Agregar**  : Muestra una ventana en la cual podrá registrar un nuevo permiso.
- **Modificar**  : Muestra una ventana donde puede modificar los datos de un permiso de trabajador ya registrado.

- **Eliminar** : puede eliminar el registro de permiso de Trabajador seleccionado.
- **Buscar**  : Permite realizar búsquedas de permisos de trabajadores en base a los registros existentes.

En la figura 17 se muestra el formulario para registrar un nuevo permiso.



**Figura 20** *Formulario de registro de un nuevo permiso*




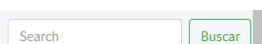
Para realizar un registro u otra acción de Producción de las secciones debe dirigirse a Personal, luego a Producción.



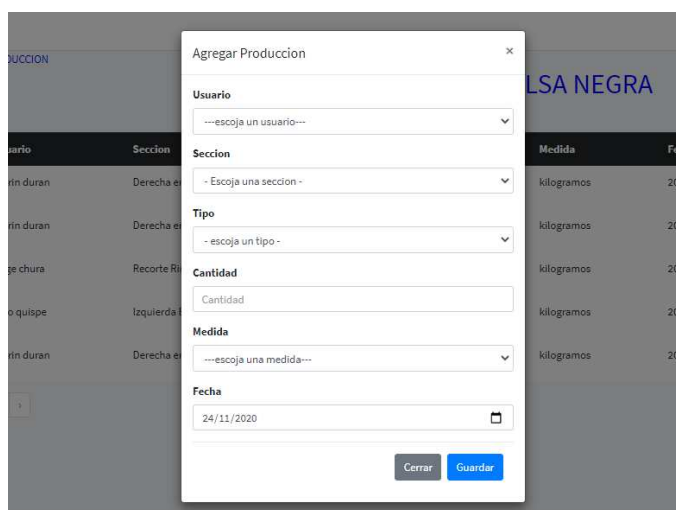
#	Usuario	Seccion	Tipo	Cantidad	Medida	Fecha	Acciones
1	marin duran	Derecha entrada	oro	1.5	kilogramos	2020-10-28	 
2	marin duran	Derecha entrada	oro	1	kilogramos	2020-09-30	 
3	jorge chura	Recorte Rincon	wolfam	6	kilogramos	2020-10-28	 
4	julio quispe	Izquierda Entrada	wolfam	13	kilogramos	2020-10-28	 
5	marin duran	Derecha entrada	oro	12	kilogramos	2020-11-07	 

**Figura 21** *Pantalla de Listado de Producción*

En la figura 21 podrá realizar las siguientes acciones:

- **Agregar**  : Muestra una ventana en la cual podrá registrar una nueva producción.
- **Modificar**  : Muestra una ventana donde puede modificar los datos de una producción ya registrado.
- **Eliminar**  : puede eliminar el registro de una producción seleccionado.
- **Buscar**  : Permite realizar búsquedas de Producción en base a los registros existentes.

En la figura 22 se muestra el formulario para registrar una nueva producción.

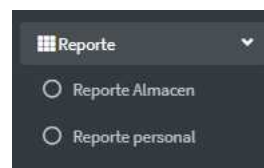


**Figura 22** *Formulario de ingreso de una nueva producción*

## 8 Menú de reportes

En el menú Reporte se encuentra los sub-menús de:

Reporte Almacén, Reporte Personal



Si se desea sacar reporte de almacén se debe seleccionar la opción Reporte Almacén del menú Reportes para sacar el tipo de reporte de almacén deseado como se ve en la figura 23.



Figura 23 pantalla de listado de reportes almacén

Una vez seleccionado el botón imprimir según lo deseado, se descargará en PDF el reporte la cual se seleccionó como se ve en la figura 24.

#	Almacen	Nombre	Tipo	Cantidad	Medida	Estado
1	almacen Piso A	pala	manual	0	unidades	inservible
2	almacen Piso A	carretilla	manual	2	unidades	reparable
3	almacen Piso A	pala	manual	0	unidades	reparable
4	almacen Piso A	Carros	manual	4	unidades	nuevo
5	almacen Piso A	carretilla	manual	5	unidades	nuevo
6	almacen Piso A	pala	manual	12	unidades	nuevo
7	almacen Derecha entrada	Barreno	manual	10	unidades	nuevo

Figura 24 Vista de reporte en pdf.

Si se desea sacar reporte de Personal se debe seleccionar la opción Reporte Personal del menú Reportes, para sacar el tipo de reporte de personal deseado como se ve en la figura 25.



COOPERATIVA MINERA BOLSA NEGRA

Nombre	Accion
Informe de listas de empleados	imprimir
Informe de Asistencia de empleados	imprimir
Informe de Permiso de empleados	imprimir
Informe de Produccion	imprimir

**Figura 25** Pantalla de listado de reporte de personal

Una vez seleccionado el botón imprimir según lo deseado, se descargará en PDF el reporte la cual se seleccionó como se ve en la figura 26.

Lista de Empleados

#	Cedula	Nombre	Apellidos	Edad	Genero	Telefono	Cargo	Seccion
1	6976558	marin	durán quispe	25	masculino	75259656	Delegado	Derecha entrada
2	2345234	martin	churata mamani	30	masculino	79645214	Perforista	Derecha entrada
3	4312234	Manuel	Fernandez Quille	23	masculino	72134245	Molinerero	Derecha entrada
4	6542341	Roger	Fernandez Quille	25	masculino	64356235	Jefe de mantenimiento	Derecha entrada
5	5423456	Felipe	Mamani Mamani	40	masculino	79645214	Carrero	Derecha entrada

**Figura 26** Vista del pdf de reporte de personal

**ANEXOS “D”**

**MANUAL TECNICO**



**Cooperativa Minera**

**Bolsa Negra**

# **MANUAL TECNICO**

**SISTEMA DE INFORMACION WEB PARA EL CONTROL DE PERSONAL**

**RESPECTO A LA PRODUCCION Y ALMACENES**

**CASO: COOPERATIVA MINERA BOLSA NEGRA**

**MARIN DURAN QUISPE**

**LA PAZ- BOLIVIA**

## **1 Requerimientos Técnicos.**

### **Requerimientos Mínimos De Hardware**

La computadora que contendrá el sistema tiene que tener las siguientes características mínimas para un buen funcionamiento.

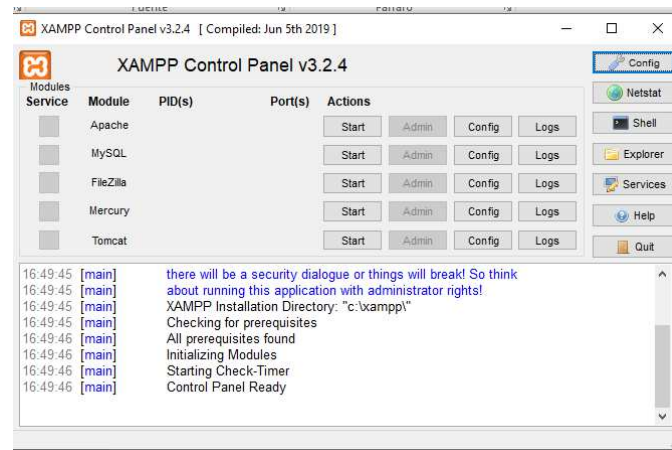
- Procesador ryzen 5 3200
- Placa madre b450m
- Tarjeta ram de 8gigas
- Disco duro de 1 tera
- Fuente de energía

### **Requisitos de software**

- Privilegios de administrador
- Sistema operativo Windows 7 o superior

## **2 instalación de web server**

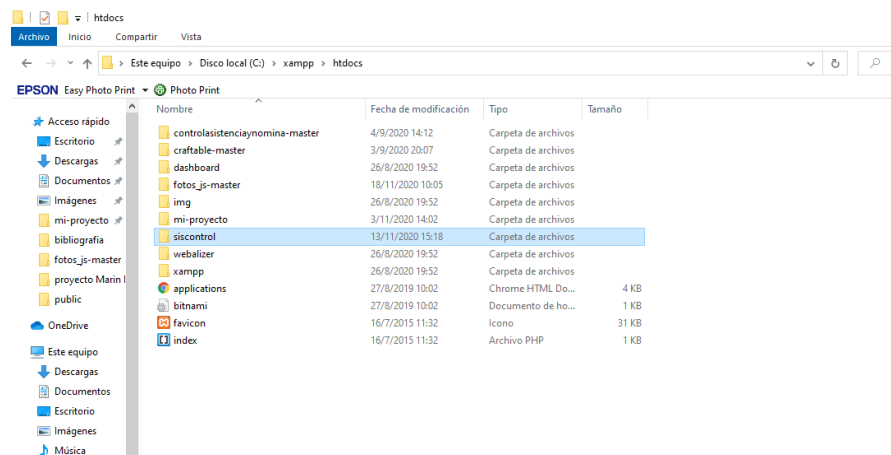
Descargar al PC el fichero de instalación de xampp el cuál se encuentra en la página:  
<https://www.apachefriends.org/es/download.html> una vez descargado, dar clic sobre él para iniciar la instalación. La instalación se debe hacer con una cuenta de Administrador o con derechos de administrador.



**Figura 27** Pantalla de administración de xampp

### 3. Configuración Del Aplicativo

Se coloca la carpeta llamada siscontrol en la ruta donde se encuentra instalado el aplicativo de xampp normalmente es: C:\xampp\htdocs



**Figura 28** Carpeta de xampp

#### 3.1 Crear Local Host

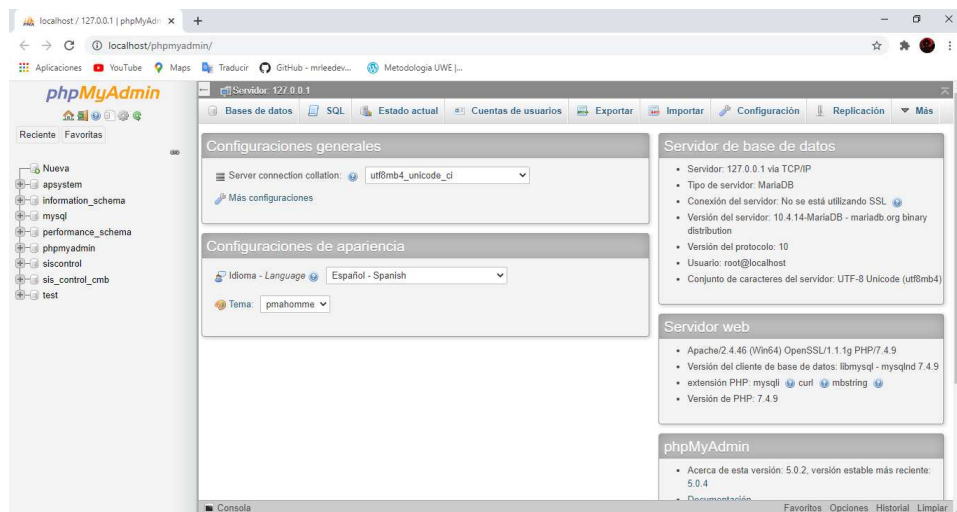
- Accederemos a la carpeta: **C:\Windows\System32\drivers\etc**
- Copiaremos el fichero **hosts** al **escritorio** y lo editaremos.

- Añadiremos la siguiente entrada al final del fichero: **127.0.0.1 siscontrol.com**
- Grabamos el fichero y lo pegaremos de nuevo en la carpeta: **C:\Windows\System32\drivers\etc** sobrescribiendo el actual.

A partir de este momento cuando queramos acceder al siscontrol.com nuestro equipo se conectará a nuestro servidor localhost.

#### 4 Crear la Base de Datos

Accede a tu servidor local **escribiendo en la barra de navegación** de tu navegador: **localhost**. A continuación, haz clic en **phpMyAdmin**. Una vez en el administrador de bases de datos, podrá importar la base de datos del sistema que se encuentra en un script en la misma carpeta del sistema, siscontrol\database y en ahí se encuentra un script con la base de datos con el nombre siscontrol.



**Figura 29** pantalla de phpMyAdmin

# DOCUMENTACION

El Alto, noviembre de 2020

Señor  
**Ing. David Carlos Mamani Quispe**  
**DIRECTOR**  
**CARRERA INGENIERÍA DE SISTEMAS – UPEA**  
Presente.-

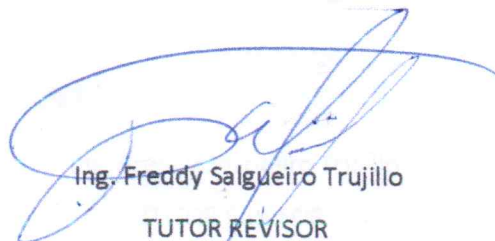
**REF.: AVAL DE CONFORMIDAD**

Distinguido ingeniero:

Tengo a bien dirigirme a su persona para comunicarle mi conformidad del trabajo final del Proyecto de Grado, titulado **“SISTEMA DE INFORMACIÓN WEB PARA EL CONTROL DE PERSONAL RESPECTO A LA PRODUCCIÓN Y ALMACENES CASO: COOPERATIVA MINERA BOLSA NEGRA”**, elaborado por el universitario: **MARIN DURAN QUISPE**, con cedula de identidad: **6976558 L.P.** y registro universitario: **10036320**, para su defensa pública y evaluación correspondiente a la materia de Taller de Licenciatura II, de acuerdo al reglamento vigente de la Carrera de Ingeniería de Sistemas de la Universidad Pública de El Alto.

Sin otro particular, me despido de usted.

Atentamente.



Ing. Freddy Salgueiro Trujillo  
TUTOR REVISOR



El Alto, noviembre de 2020

Señor  
**Ing. David Carlos Mamani Quispe**  
**DIRECTOR**  
**CARRERA INGENIERÍA DE SISTEMAS – UPEA**  
Presente. -

**REF.: AVAL DE CONFORMIDAD**

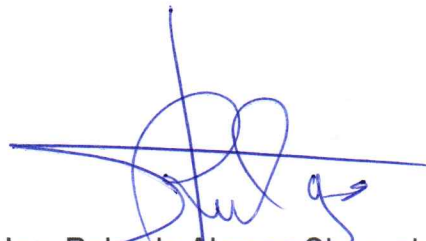
Distinguido ingeniero:

Tengo a bien dirigirme a su persona para comunicarle mi conformidad del trabajo final del Proyecto de Grado, titulado **“SISTEMA DE INFORMACIÓN WEB PARA EL CONTROL DE PERSONAL RESPECTO A LA PRODUCCIÓN Y ALMACENES**

**CASO: COOPERATIVA MINERA BOLSA NEGRA”**, elaborado por el universitario: **MARIN DURAN QUISPE**, con cedula de identidad: **6976558 L.P.** y registro universitario: **10036320**, para su defensa pública y evaluación correspondiente a la materia de Taller de Licenciatura II, de acuerdo al reglamento vigente de la Carrera de Ingeniería de Sistemas de la Universidad Pública de El Alto.

Sin otro particular, me despido de usted.

Atentamente.



Ing. ~~Rolando Alarcon Choquehuaca~~  
TUTOR ESPECIALISTA

El Alto, noviembre de 2020

Señor  
Ing. David Carlos Mamani Quispe  
DIRECTOR  
CARRERA INGENIERÍA DE SISTEMAS – UPEA  
Presente. -

**REF.: AVAL DE CONFORMIDAD**

Distinguido ingeniero:

Tengo a bien dirigirme a su persona para comunicarle mi conformidad del trabajo final del Proyecto de Grado, titulado **“SISTEMA DE INFORMACIÓN WEB PARA EL CONTROL DE PERSONAL RESPECTO A LA PRODUCCIÓN Y ALMACENES CASO: COOPERATIVA MINERA BOLSA NEGRA”**, elaborado por el universitario: **MARIN DURAN QUISPE**, con cedula de identidad: **6976558 L.P.** y registro universitario: **10036320**, para su defensa pública y evaluación correspondiente a la materia de Taller de Licenciatura II, de acuerdo al reglamento vigente de la Carrera de Ingeniería de Sistemas de la Universidad Pública de El Alto.

Sin otro particular, me despido de usted.

Atentamente.

  
Ing. Marisol Arguedas Balladares  
TUTOR METODOLÓGICO



# Cooperativa Minera "Bolsa Negra" R.L.

Fundada en 15-X-1965 - Aprobado según Resolución No. 00469 - Registro Dirección Nacional Cooperativas No. 446  
Bolsa Negra - La Paz - Bolivia



Of. La Paz: Pasaje Inca No. 100  
Teléfono; 2-462955

CMBNRL- CITE No. \_\_\_\_\_

La Paz, de Noviembre de 2020

Señor:  
Ing. David Carlos Mamani Quispe  
DIRECTOR DE CARRERA INGENIERIA DE SISTEMAS  
Presente.-

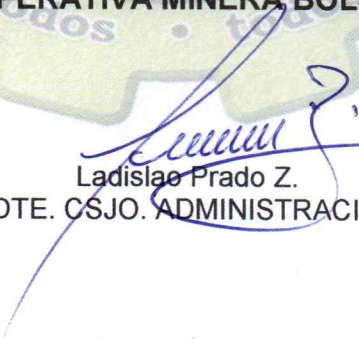
**Ref.: AVAL DE CONFORMIDAD.-**

Mediante la presente quisiera hacer de su conocimiento la conformidad con el trabajo que ha realizado para la cooperativa Minera Bolsa Negra y tengo bien comunicarle mi conformidad del proyecto de grado SISTEMA DE INFORMACION WEB PARA EL CONTROL DE PERSONAL RESPECTO A LA PRODUCCION Y ALMACENES CASO: COOPERATIVA MINERA BOLSA NEGRA, propuesto por el estudiante MARIN DURAN QUISPE, con cedula de identidad 6976558 para su defensa pública de la carrera de ingeniería de sistemas de la Universidad Pública de El Alto..

Sin otro particular mediato, le saludamos muy atentamente.

Atentamente.

**P' COOPERATIVA MINERA BOLSA NEGRA RL.**

  
Ladislao Prado Z.  
PDTE. CSJO. ADMINISTRACION

