

# UNIVERSIDAD PÚBLICA DE EL ALTO

## CARRERA INGENIERÍA DE SISTEMAS



### PROYECTO DE GRADO

**“SISTEMA PARA LA GESTIÓN DE CLIENTES MEDIANTE EL USO DE  
MICROSERVICIOS, A TRAVÉS DEL MODELO DE NEGOCIO CRM”**

**CASO: ELECTRONIC, PRINTER & SERVICES S.R.L.**

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

**MENCIÓN: INFORMÁTICA Y COMUNICACIONES**

**Postulante: Univ. Ricardo Franco Sanchez Quispe**

**Tutor Metodológico: M. Sc. Lic. Ing. Enrique Flores Baltazar**

**Tutor revisor: Lic. Santos Chillo Espinoza**

**Tutor Especialista: Lic. Maria Magdalena Aguilar Guanto**

**EL ALTO – BOLIVIA**

**2023**

## **DECLARACIÓN JURADA DE AUTENTICIDAD Y RESPONSABILIDAD**

Yo **Ricardo Franco Sanchez Quispe** estudiante con **C.I. 9204244 LP.** mediante la presente **declaro** de manera pública que la propuesta del **PROYECTO DE GRADO** titulado “**SISTEMA PARA LA GESTIÓN DE CLIENTES MEDIANTE EL USO DE MICROSERVICIOS, A TRAVÉS DEL MODELO DE NEGOCIO CRM**” es original, siendo resultado de mi trabajo personal y no constituye una copia o replica de trabajos similares elaborados.

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

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

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

El Alto, diciembre de 2023.

-----  
**Ricardo Franco Sanchez Quispe**  
**C.I. 9204244 LP**  
**e-mail: rsanchezhell69@gmail.com**

Dedicatoria:

*“A mis queridos padres, quienes han sido mi mayor inspiración y apoyo incondicional a lo largo de toda mi vida. Gracias por su amor, paciencia y sacrificios para brindarme las herramientas necesarias para alcanzar mis metas. Sin su guía y aliento, el presente trabajo no hubiera sido posible. Siempre estaré agradecido y orgulloso de tenerlos como mis padres”*

## Agradecimientos

"Quiero expresar mi más sincero agradecimiento a mis tutores, M. Sc.Ing. Enrique Flores Baltazar, Ing. Santos Chillo y también a la Lic. María Magdalena Aguilar Guanto por su paciencia, dedicación y orientación durante todo el proceso de desarrollo de mi proyecto. Gracias por compartir sus conocimientos y experiencias conmigo.

También quiero agradecer a Electronic, Printer & Services por darme la oportunidad de realizar mi proyecto en sus instalaciones y por brindarme todas las facilidades necesarias para llevar a cabo el presente trabajo.

Gracias a todos ellos, este proyecto ha sido posible y ha sido una experiencia enriquecedora para mi crecimiento profesional y personal."

## **Resumen**

La informática se ha vuelto esencial en la vida empresarial debido a la cantidad de información que manejan dichas empresas actualmente.

La idea de automatizar las actividades cotidianas en las organizaciones es cada vez más necesaria, la empresa E.P.S. S.R.L se dedica a dar soporte técnico con este proyecto se busca mejorar la eficiencia y confiabilidad en el manejo de la información y reducir la probabilidad de errores en la empresa, actualmente la empresa maneja sus actividades de forma manual, lo cual genera demoras, falta de control en los procesos y una alta probabilidad de errores. Para mejorar y agilizar este proceso se propuso implementar un sistema CRM (Customer Relationship Management) que permita centralizar la información de los clientes, proveedores y actividades, y así pasar de un enfoque de producto a un enfoque al cliente. El objetivo de la investigación es analizar la relación entre el sistema CRM y la automatización de la información en las empresas, y buscar herramientas alternativas para potenciar el uso y la adopción del CRM en la empresa E.P.S. S.R.L

**Palabras clave:** CRM, E.P.S., procesos, información

## **Abstract**

Computer science has become essential in business life due to the amount of information that these companies currently handle.

The idea of automating daily activities in organizations is increasingly necessary, the company E.P.S. S.R.L is dedicated to providing technical support with this project, which seeks to improve efficiency and reliability in information management and reduce the probability of errors in the company, currently the company manages its activities manually, which generates delays, lack of control in the processes and a high probability of errors. To improve and expedite this process, a CRM (Customer Relationship Management) system will be implemented to centralize customer, supplier and activity information, and thus move from a product focus to a customer focus. The objective of the research is to analyze the relationship between the CRM system and the automation of information in companies, and to look for alternative tools to promote the use and adoption of CRM in the company E.P.S. S.

**Keywords:** CRM, E.P.S., processes, information

## Contenido

1 MARCO PRELIMINAR .....	1
1.1 INTRODUCCIÓN.....	1
1.2 ANTECEDENTES DEL PROYECTO.....	2
1.2.1 Antecedentes institucionales .....	2
1.2.2 Antecedentes de trabajos afines .....	3
1.3 PLANTEAMIENTO DEL PROBLEMA .....	5
1.3.1 Problema Principal .....	6
1.3.2 Problemas Secundarios.....	6
1.3.3 Formulación del problema .....	6
1.4 OBJETIVOS .....	7
1.4.1 Objetivo General.....	7
1.4.2 Objetivos Específicos .....	7
1.5 JUSTIFICACIÓN DEL TEMA.....	8
1.5.1 Justificación Técnica.....	8
1.5.2 Justificación Económica .....	8
1.5.3 Justificación Social .....	9
1.6 METODOLOGÍA.....	9
1.6.1 Scrum .....	9
1.6.2 ISO 27001:2022 .....	9
1.6.3 ISO 25000 .....	9
1.6.4 Arquitectura MVC .....	10
1.6.5 Arquitectura de micro servicios.....	11
1.6.6 Métodos de recolección de datos .....	11
1.6.7 Estimación de costos.....	12
1.6.8 Pruebas de software.....	12
1.6.9 Pruebas de estrés .....	13

1.7 HERRAMIENTAS .....	13
1.7.1 Lenguaje de programación .....	13
1.7.2 Framework.....	14
1.7.3 Jasper Soft .....	14
1.7.4 Lenguaje de diseño .....	14
1.7.5 Scene Builder .....	15
1.7.6 Editor de código.....	15
1.7.7 Base de datos.....	16
1.8 LÍMITES Y ALCANCES .....	16
1.8.1 Limites .....	16
1.8.2 Alcances.....	16
1.9 APORTES .....	17
2 MARCO TEÓRICO .....	18
2.1 CONCEPTOS BÁSICOS .....	18
2.1.1 Dato .....	18
2.1.2 Información.....	18
2.1.3 Sistema.....	18
2.1.4 Seguimiento y control .....	19
2.1.5 Gestión de clientes .....	19
2.1.6 CRM .....	20
2.1.7 Microservicios.....	20
2.1.8 Gestión de clientes .....	21
2.1.9 Modelo de negocio .....	21
2.2 Metodologías .....	22
2.2.1 Scrum .....	22
2.2.3 Arquitectura de microservicios.....	24



2.2.3 Ciclo de vida de los microservicios .....	25
2.2.4 Arquitectura MVC .....	26
2.2.5 Customer relationship management “CRM” .....	28
2.2.6 Tipos de CRM.....	32
2.2.7 UML.....	33
2.3 HERRAMIENTAS .....	34
2.3.1 Lenguaje de programación Java .....	34
2.3.2 Framework Java Fx .....	35
2.3.3 Lenguaje de diseño gráfico CSS .....	35
2.3.4 Entorno de desarrollo IntelliJ IDEA.....	36
2.3.5 Oracle Database.....	36
2.4 CALIDAD DEL SOFTWARE .....	37
2.4.1 ISO 2500 .....	38
2.5 PRUEBAS DE SOFTWARE .....	38
2.5.1 Métricas .....	39
2.5.2 Calidad interna y externa.....	39
2.5.3 Calidad en uso.....	39
2.5.4 Prueba de Caja blanca .....	40
2.5.5 Prueba de caja negra .....	40
2.5.6 Pruebas de estrés .....	40
2.6 COSTOS DEL SOFTWARE .....	41
2.6.1 COCOMO II .....	41
2.7 SEGURIDAD .....	45
2.7.1 ISO 27000:2022 .....	46
2.7.2 Autenticar al usuario .....	46
2.7.3 Roles de los usuarios .....	47

2.7.4	Encriptación de contraseñas .....	47
2.7.5	Backup.....	48
2.8	MANTENIMIENTO DEL SOFTWARE .....	48
2.8.1	Pruebas unitarias.....	49
2.8.2	Pruebas de interfaz de usuario (UI).....	49
2.8.3	Pruebas de Integración en JavaFX .....	50
3.	MARCO APLICATIVO .....	48
3.1	INTRODUCCIÓN.....	48
3.2	APLICACIÓN DE LA METODOLOGÍA SCRUM.....	48
3.3	EQUIPO SCRUM .....	48
3.4	APLICANDO EL MODELO CRM.....	49
3.4.1	Descripción y características del sistema.....	50
3.5	PRIMERA ITERACIÓN.....	51
3.6.1	Etapa de planificación.....	51
3.6.1.1	Análisis de requerimientos.....	51
3.6.2	Etapa de implementación .....	56
3.7	SEGUNDA ITERACIÓN .....	58
3.7.1	Etapa de planificación.....	58
3.6.2	Etapa de ejecución .....	63
3.7.3	Etapa de prueba y evaluación .....	66
3.8	TERCERA ITERACIÓN .....	67
3.8.1	Etapa de planificación.....	67
3.8.2	Etapa de ejecución .....	70
3.8.3	Etapa de pruebas y evaluaciones.....	76
3.9	CUARTA ITERACIÓN .....	78
3.9.1	Etapa de planificación.....	78
3.9.2	Etapa de ejecución .....	81

3.9.3 Etapa de prueba y evaluación .....	83
3.10 IMPLEMENTACIÓN .....	84
3.11 USO DE MICROSERVICIOS .....	97
3.11.1 Aplicando microservicios .....	97
4 CALIDAD COSTO Y SEGURIDAD.....	96
4.1 Introducción .....	96
4.2 Aplicación de la Métrica de Calidad ISO 25000.....	96
4.2.1 Funcionalidad .....	96
4.2.2 Usabilidad.....	102
4.2.3 Confiabilidad.....	103
4.2.4 Mantenibilidad .....	104
4.2.5 Evaluación global .....	105
4.3 ESTIMACIÓN DEL COSTO.....	106
4.3.1 COCOMO II .....	106
4.3.2 Costo de elaboración.....	108
4.3.3 Costo total del proyecto.....	109
4.4 Seguridad .....	109
4.4.1 ISO 27000:2022 .....	109
4.4.2 Seguridad de base de datos.....	110
4.4.3 Seguridad lógica .....	111
4.4.4 Seguridad física.....	111
4.4.5 Backup de la base de datos .....	112
4.5 Pruebas del sistema .....	112
4.5.1 Prueba unitaria .....	112
4.5.2 Prueba de integración.....	113
4.5.3 Prueba interfaz de usuario UI .....	114
4.5.4 Prueba de caja de blanca .....	115

4.5.5 Prueba de caja negra ..... 117

4.5.6 Prueba de estrés ..... 119

5 Conclusiones y recomendaciones ..... 113

5.1 Conclusiones ..... 113

5.2 Recomendaciones ..... 114

Bibliografía..... 116

## Índice de tablas

<b>Tabla 2.1</b> Aplicaciones en COCOMO II.....	42
<b>Tabla 2.2</b> Puntos objeto en COCOMO II.....	43
<b>Tabla 3.1</b> Equipo scrum .....	49
<b>Tabla 3.2</b> backlog usuarios .....	52
<b>Tabla 3.3</b> backlog servicios.....	52
<b>Tabla 3.4</b> backlog equipos .....	53
<b>Tabla 3.5</b> backlog técnicos.....	53
<b>Tabla 3.6</b> backlog clientes.....	54
<b>Tabla 3.7</b> backlog seguimiento.....	54
<b>Tabla 3.8</b> backlog reportes.....	55
<b>Tabla 3.9</b> Sprint backlog de usuarios .....	59
<b>Tabla 3.10</b> Grupo de usuario administrador.....	59
<b>Tabla 3.11</b> Inicio de sesión usuario.....	60
<b>Tabla 3.12</b> Creación de usuarios .....	60
<b>Tabla 3.13</b> Administración de usuarios .....	61
<b>Tabla 3.14</b> Asignación de roles de usuario .....	61
<b>Tabla 3.15</b> Bloqueo de usuarios.....	61
<b>Tabla 3.16</b> Cambio de contraseñas del usuario.....	62
<b>Tabla 3.17</b> Desbloqueo de usuarios.....	62
<b>Tabla 3.18</b> Reseteo de contraseñas .....	62
<b>Tabla 3.19</b> Prueba de modulo usuarios .....	67
<b>Tabla 3.20</b> Sprint backlog de técnicos .....	67
<b>Tabla 3.21</b> Sprint backlog de clientes .....	68
<b>Tabla 3.22</b> Crud de servicios.....	69
<b>Tabla 3.23</b> Crud de equipos.....	69
<b>Tabla 3.24</b> Crud de clientes .....	69
<b>Tabla 3.25</b> Crud de seguimiento .....	70
<b>Tabla 3.26</b> Prueba registro de datos modulo taller.....	76
<b>Tabla 3.27</b> Prueba edición de datos modulo taller .....	77

<b>Tabla 3.28</b> Prueba edición de datos modulo taller .....	77
<b>Tabla 3.29</b> Sprint backlog de reportes .....	78
<b>Tabla 3.30</b> Selección de cliente para reporte .....	79
<b>Tabla 3.31</b> Selección de fecha para reporte.....	79
<b>Tabla 3.32</b> Informe de servicios técnico preventivo y correctivo .....	80
<b>Tabla 3.33</b> Informe de servicios técnico preventivo y correctivo equipo .....	80
<b>Tabla 3.34</b> Informe técnico.....	80
<b>Tabla 3.35</b> Prueba de reportes.....	84
<b>Tabla 3. 36</b> Página de dashboard .....	96
<b>Tabla 4.1</b> Número de entradas de Usuario .....	97
<b>Tabla 4.2</b> Número de salidas de Usuario .....	97
<b>Tabla 4.3</b> Número de entradas de Usuario .....	98
<b>Tabla 4.4</b> Complejidad funcional de cada modulo.....	98
<b>Tabla 4.5</b> Ponderación de Valores .....	99
<b>Tabla 4.6</b> Preguntas de complejidad.....	99
<b>Tabla 4.7</b> Ajuste de preguntas .....	102
<b>Tabla 4.8</b> Tabla evaluación global.....	106
<b>Tabla 4.9</b> Cantidad de línea de código.....	107
<b>Tabla 4.10</b> Costo beneficio del proyecto .....	108
<b>Tabla 4.11</b> Costo total del proyecto.....	109
<b>Tabla 4.12</b> Prueba de autenticación.....	117
<b>Tabla 4.13</b> Prueba de registro de servicios.....	117
<b>Tabla 4.14</b> Prueba de alta de servicios .....	118

## Índice de figuras

<b>Figura 1.1</b> Organigrama Electronic Printer & Services .....	3
<b>Figura 1.2</b> Arquitectura MVC .....	11
<b>Figura 2.1</b> Procesos de Scrum .....	22
<b>Figura 2.2</b> Elementos del sprint .....	23
<b>Figura 2.3</b> Arquitectura de los microservicios .....	24
<b>Figura 2.4</b> Modelo UML .....	33
<b>Figura 2.5</b> Ejemplo de diagramas UML .....	34
<b>Figura 2.6</b> ISO 2500 .....	38
<b>Figura 2.7</b> Evaluación Interna, externa y Calidad de Uso ISO/IEC 2500.....	39
<b>Figura 2.8</b> Autenticación de un usuario .....	46
<b>Figura 2.9</b> Esquema de roles de un sistema.....	47
<b>Figura 3.1</b> Caso de uso primera iteración.....	56
<b>Figura 3.2</b> Diagrama de clases primera iteración .....	56
<b>Figura 3.3</b> Modelo entidad relación primera iteración .....	57
<b>Figura 3.4</b> Código modulo primera iteración.....	58
<b>Figura 3.5</b> Casos de uso usuarios .....	63
<b>Figura 3.6</b> Casos de uso administración de usuarios .....	63
<b>Figura 3.7</b> Diagrama de clases modulo usuarios.....	64
<b>Figura 3.8</b> Modelo entidad relación módulo de usuarios .....	65
<b>Figura 3.9</b> Código modulo usuarios .....	66
<b>Figura 3.10</b> Casos de uso entrada de usuarios al módulo taller.....	70
<b>Figura 3.11</b> Casos de uso tareas de los usuarios.....	71
<b>Figura 3.12</b> Caso de uso registro de equipos .....	71
<b>Figura 3.13</b> Caso de uso registro de técnicos.....	72
<b>Figura 3.14</b> Caso de uso registro de clientes.....	72
<b>Figura 3.15</b> Caso de uso servicios.....	73
<b>Figura 3.16</b> Caso de uso seguimiento .....	73
<b>Figura 3.17</b> Diagrama de clases segunda iteracion.....	74
<b>Figura 3.18</b> Modelo entidad relación módulo de taller .....	75
<b>Figura 3.19</b> Código modulo taller.....	76

<b>Figura 3.20</b> Caso de uso cuarta iteracion .....	81
<b>Figura 3.21</b> Caso de uso realización de reportes .....	81
<b>Figura 3.22</b> Construcción de la estructura de reporte.....	82
<b>Figura 3.23</b> Código resultado de módulo de reportes.....	82
<b>Figura 3.24</b> Generación de reporte en pdf.....	83
<b>Figura 3.25</b> Generación de reporte en Excel.....	83
<b>Figura 3.26</b> Página de login.....	85
<b>Figura 3.27</b> Página de acceso denegado .....	85
<b>Figura 3.28</b> Página de nuevos usuarios .....	86
<b>Figura 3.29</b> administración de roles de usuario .....	86
<b>Figura 3.30</b> Administración de usuarios.....	87
<b>Figura 3.31</b> asignación de roles de usuario .....	87
<b>Figura 3.32</b> bloquear cuenta de usuario .....	88
<b>Figura 3.33</b> Cambio de password de usuario .....	88
<b>Figura 3.34</b> desbloquear usuarios .....	89
<b>Figura 3.35</b> reseteo de password de usuario.....	89
<b>Figura 3.36</b> roles de usuario .....	90
<b>Figura 3.37</b> Página de reportes .....	90
<b>Figura 3.38</b> Selección de cliente para reporte .....	91
<b>Figura 3.39</b> Selección de fecha para reporte.....	91
<b>Figura 3.40</b> Selección de informe de mantenimiento preventivo y correctivo .....	92
<b>Figura 3.41</b> Selección de informe de mantenimiento por equipo.....	92
<b>Figura 3.42</b> Selección de informe técnico.....	93
<b>Figura 3.43</b> Página de menú principal .....	93
<b>Figura 3.44</b> Página de clientes .....	94
<b>Figura 3.45</b> Página de técnicos .....	94
<b>Figura 3.46</b> Página de equipos.....	95
<b>Figura 3.47</b> Página de servicios.....	95
<b>Figura 3.48</b> Página de seguimiento .....	96
<b>Figura 3.49</b> Microservicios módulo de usuarios.....	97
<b>Figura 3.50</b> Microservicios marca y cliente .....	98



<b>Figura 3.51</b> Microservicios equipos y técnicos.....	98
<b>Figura 3.52</b> Microservicios servicios .....	99
<b>Figura 4.1</b> Prueba unitaria .....	112
<b>Figura 4.2</b> Resultado obtenido prueba unitaria .....	113
<b>Figura 4.3</b> Prueba de integración.....	113
<b>Figura 4.4</b> Resultado obtenido prueba integración .....	114
<b>Figura 4.5</b> Prueba interfaz de usuario UI .....	114
<b>Figura 4.6</b> Resultado prueba interfaz de usuario UI .....	115
<b>Figura 4.7</b> Flujograma prueba de caja blanca.....	115
<b>Figura 4.8</b> Prueba de estrés .....	119
<b>Figura 4.9</b> Resultados prueba de estrés.....	120

## ***Listado de abreviaciones***

<b>E.P.S.</b>	Electronic Printer & Services
<b>S.R.L.</b>	Sociedad de responsabilidad limitada
<b>MVC</b>	Modelo vista controlador
<b>CRM</b>	Gestión de Relación con los Clientes (Customer Relationship Management)
<b>DB</b>	Base de datos (Data base)
<b>UI</b>	Interfaz de Usuario (User Interface)
<b>SQL</b>	Lenguaje de Consulta Estructurado (Structured Query Language)
<b>JavaFX</b>	Java Effects
<b>MS</b>	Microservicios
<b>Oracle DB</b>	Oracle Database
<b>CSS</b>	Hojas de estilo en cascada (Cascading Style Sheets)
<b>IT</b>	Tecnología de la Información (Information Technology)
<b>IDE</b>	Entorno de Desarrollo Integrado (IntelliJ IDEA)
<b>GUI</b>	Interfaz Gráfica de Usuario (Graphical User Interface)
<b>XML</b>	Lenguaje de Marcado Extensible (Extensible Markup Language)
<b>JSP</b>	Páginas de Servidor Java (JavaServer Pages)
<b>JAR:</b>	Archivo de Java (Java Archive)
<b>JR</b>	Jasper Reports
<b>API de JR</b>	Interfaz de Programación de Aplicaciones de Jasper Reports



# CAPÍTULO I MARCO PRELIMINAR

# 1. MARCO PRELIMINAR

## 1.1 INTRODUCCIÓN

Hoy en día, la informática se ha convertido en una parte esencial en la vida de una empresa la razón principal implica la cantidad de información que actualmente se maneja, hace que el tratamiento automático de la información sea realmente útil y necesario.

En la actualidad los sistemas de información están basados en computadoras que son objetos de gran consideración en la toma de decisiones oportunas, confiables y efectivas en cuanto a técnicas de planificación, programación y administración con el fin de garantizar su éxito, limitar el riesgo, reducir costos y aumentar las ganancias.

Debido a esta razón, nace la idea de automatizar las actividades cotidianas en las organizaciones; cabe mencionar el avance de las telecomunicaciones y el progreso en las ciencias informáticas que obliga a entrar al moderno mundo de la tecnología.

La empresa E.P.S S.R.L (Electronic Printer & Services) se dedica a dar soporte técnico, para este trabajo cuenta con varios trabajadores los cuales deben realizar sus reportes de trabajo, además, de controlar lo equipos que llegan a su oficina. Este proceso se realiza de forma manual lo cual conlleva a una demora en las actividades, baja confiabilidad en el manejo de la información y alta probabilidad de errores en la empresa.

El presente proyecto pretende mejorar y agilizar todo el proceso de trabajo de los técnicos de la empresa con la implementación de un sistema CRM (Customer Relationship Management– Administración de la Relación con los Clientes), mediante el uso de una base de datos Oracle en su versión gratuita, así mismo se eligió el lenguaje de programación Java, todo este proceso se desarrollará con una metodología de desarrollo ágil la cual sería Scrum, métrica de calidad ISO 2500 y para la seguridad del mismo se implementara la ISO 2700:2022.

## **1.2 ANTECEDENTES DEL PROYECTO**

### **1.2.1 Antecedentes institucionales**

La empresa E.P.S S.R.L. es una institución que tiene por objetivo dar soporte técnico de calidad a empresas que requieran de estos servicios.

➤ **Misión**

La misión de E.P.S S.R.L. es proporcionar soluciones tecnológicas confiables y efectivas a nuestros clientes, Nos esforzamos por ser líderes en el campo del soporte técnico electrónico, brindando servicios personalizados y de alta calidad que satisfagan las necesidades únicas de cada cliente. Además, nos comprometemos a mantenernos actualizados con las últimas tendencias y avances tecnológicos para poder ofrecer soluciones innovadoras y efectivas.

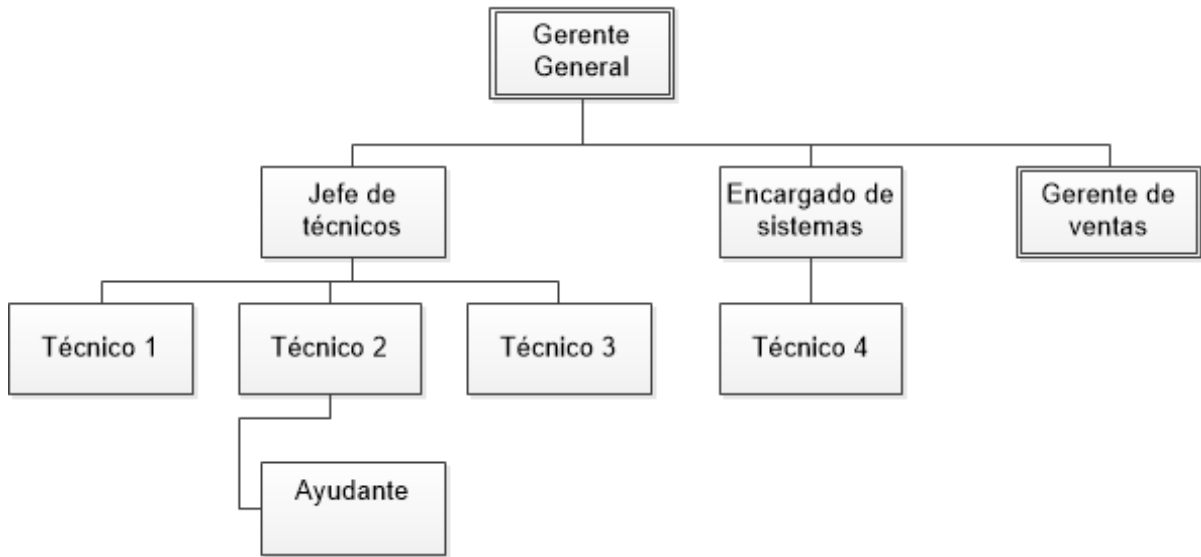
➤ **Visión**

Ser una institución reconocida como una de las mejores en el campo en brindar servicio técnico de calidad.

➤ **Objetivo**

Nuestro objetivo es convertirnos en el referente en el sector del soporte técnico, destacando por la calidad de nuestros servicios y la satisfacción de nuestros clientes. Buscamos establecer relaciones duraderas con nuestros clientes, ofreciendo soluciones rápidas y efectivas que resuelvan sus problemas tecnológicos. Además, nos esforzamos por estar a la vanguardia de la tecnología, proporcionando soluciones innovadoras y adaptadas a las necesidades cambiantes del mercado.

**Figura 1.1**  
Organigrama Electronic Printer & Services



Nota (Organigrama de trabajadores de la empresa Electronic Printer & Services)

## 1.2.2 Antecedentes de trabajos afines

### 1.2.2.1 Antecedentes Internacionales

- (López-Salinas, 2013) “Diseño de un modelo de implementación de software libre en la nube, orientado a la parte administrativa de las relaciones con el cliente (CRM) para SUPRALIVE S.A. del cantón Samborondón”, presentado en la Universidad de Guayaquil Ecuador. El uso de un modelo de implementación CRM alojado en la nube permite gestionar de manera eficiente la información relacionada con el cliente, detectando nuevas oportunidades de venta y minimizando costos ante la adquisición de un centro de cómputo. Mediante el presente se pretende diseñar un plan en el cual se implementarán acciones que se puedan aplicar a Top Seg S.A. y en donde se logre un mejoramiento tanto en los índices de retención como en los de fidelización de sus clientes mediante la aplicación de la estrategia de Gestión de Relaciones con el Cliente (CRM).
- (Heredia, 2015), “Aplicación del Customer Relationship Management – CRM para el mejoramiento de sistemas de fidelización y retención de clientes en el sector de

seguros, em el distrito metropolitano de Quito, caso: TOP SEG compañía de seguros”, Presentado en la Universidad de las fuerzas Armadas de Ecuador.

- (Rojas, 2017), “Implementación de un sistema CRM para la mejora en la gestión de atención al cliente para una empresa del sector servicios”, presentado en la Universidad Mayor de San Marcos Lima-Perú. El presente trabajo tiene por objetivo mejorar la gestión de atención al cliente a través de la implementación de un sistema CRM, el CRM que se estudia en este proyecto ha sido desarrollado a medida, considerando la experiencia de los programadores, los usuarios y especialistas en marketing.
- (Toalongo, 2013). “Propuesta de implementación de una estrategia basada en CRM para la empresa AGROTA CIA. LTDA.”, desarrollado den la Universidad Politécnica Salesiana Cuenca – Ecuador. En el presente trabajo se busca conocer los fundamentos para la implementación de una estrategia basada en Gestión de relaciones con los Clientes, examina a profundidad cuales son las características que establecen una relación exitosa con los clientes.

#### **1.2.2.2 Antecedentes Nacionales**

- (Montevilla, 2009) “Sistema de información para la gestión de la relación con los clientes”, presentado en la Universidad Mayor de San Andrés. Mediante el presente se pretende desarrollar los medios tecnológicos de software que permitan a la fuerza de ventas de la empresa realizar una gestión adecuada de la relación con sus clientes.
- (Bardales y Galarza, 2018), “Implementación de un ERP para la automatización del proceso logístico en una empresa de servicios técnicos”, presentado en la universidad San Ignacio De Loyola para la empresa Servicenter S.A.C. Esta tesis busca automatizar los procesos logísticos de una empresa de servicios técnicos

tales como compra, venta de repuestos, servicio al cliente, almacenamiento y gestión de inventario; todo ello a través de la implementación de un ERP

### **1.2.2.3 Antecedentes Locales**

- (Jhonny Vargas Ramos, 2020) “sistema de información para la gestión académica en un entorno web para la unidad de kardex académico” caso: carrera contaduría pública, universidad pública de el alto”. Desarrollar un Sistema de Información para la Gestión Académica en un entorno Web para la Unidad de Kardex Académico Carrera de Contaduría Pública de la Universidad Pública de El Alto, con la finalidad de que la información sea generada de manera oportuna, efectiva y confiable para el uso y catalogación de la misma,
- (Erick Roger Gutierrez Zapata, 2020) “sistema de gestión y digitalización bibliotecaria caso: carrera de odontología U.P.E.A.”. Desarrollar e implementar un Sistema de Gestión y Digitalización Bibliotecaria como herramienta para administrar y controlar la información de material bibliográfico de forma segura y oportuna, dando una calidad de servicio y satisfacción a los usuarios permitiendo la correcta toma de decisiones, ,
- (Juan Carlos Choque Mamani, 2020)” sistema web de gestión de inventarios para almacenes caso: caja nacional de salud regional la paz – sección de equipos electrónicos”, El objetivo principal de administración Regional La Paz es brindar de forma oportuna eficaz y eficiente la presentación de servicios de salud de calidad y calidez a toda su población protegida, optimizando el uso de recursos e insumos buscando ampliar el nivel de cobertura,

### **1.3 PLANTEAMIENTO DEL PROBLEMA**

Según el análisis preliminar la empresa E.P.S. S.R.L. administra de forma escrita los equipos, reportes de trabajo y demás actividades, esto a causa de que no se cuenta con un modo de seguimiento adecuado lo cual provoca una pérdida de información, desempeño lento en las actividades diarias de los técnicos y de falta de control en los procesos diarios por parte de los directivos.



### **1.3.1 Problema Principal**

El problema principal abordado en el presente proyecto es la falta de eficiencia y automatización en la gestión de la información de los clientes y las actividades realizadas por los técnicos en la empresa EPS S.R.L. Actualmente los técnicos llevan a cabo la gestión de datos de forma escrita o en archivos Excel, lo que provoca una probabilidad de que cometan errores a la hora de administrar dichos datos, demoras y falta de control en los procesos. Por lo tanto, se requiere la implementación de un sistema CRM para mejorar la eficiencia en la gestión de los datos y las actividades garantizando la toma de decisiones confiables.

### **1.3.2 Problemas Secundarios**

- No se cuenta con un registro digitalizado de los trabajos realizados por la empresa lo cual provoca una demora en el trabajo.
- La información de los clientes es insuficiente y en muchos casos inexistente lo que hace difícil la atención al cliente
- No se cuenta con herramientas tecnológicas que permitan gestionar de forma eficiente el trabajo demorando el tiempo empleado en este.
- No existe un método de búsqueda de información generando trabajo extra al buscar dicha información.
- El registro de la información es lento y cansador para los trabajadores.
- No existe un servicio de atención al cliente lo cual provoca quejas y descontento por parte de ellos.
- La lentitud en el trabajo realizado de forma manual por los empleados genera la pérdida de clientes.
- El registro de trabajo solo puede ser realizado de manera manual provocando una demora en la entrega de los mismos.

### **1.3.3 Formulación del problema**

Dado todos estos problemas en la empresa E.P.S S.R.L. se llegó a la siguiente pregunta:

¿Cómo implementar un sistema CRM basado en microservicios para mejorar la eficiencia y el control en la gestión de datos de clientes y trabajos realizados por los técnicos en la empresa E.P.S. S.R.L. considerando la falta de seguimiento adecuado?

## **1.4 OBJETIVOS**

### **1.4.1 Objetivo General**

Desarrollar e implementar un “Sistema para la gestión de clientes mediante el uso de microservicios, a través del modelo de negocio CRM”, Para mejorar el control y el seguimiento de los datos de los cliente y trabajos realizados por el personal de la empresa ELECTRONIC, PRINTER & SERVICES S.R.L.

### **1.4.2 Objetivos Específicos**

- Realizar un análisis exhaustivo de los requerimientos identificando de manera precisa y detallada las necesidades y funcionalidades clave que deben ser abordadas en el diseño e implementación del sistema.
- Aplicar el modelo de negocio CRM para optimizar la gestión del trabajo de la empresa E.P.S S.R.L., permitiendo una mayor eficiencia en la ejecución del trabajo realizado.
- Diseñar una base de datos que permita ingresar y almacenar de manera fácil la información de cada trabajo realizado por los técnicos de la empresa además de los datos de clientes, equipos entre otros, ahorrando tiempo a los empleados
- Implementar el uso de microservicios permitiendo una mayor concurrencia en los datos cargados al sistema
- Permitir el registro de trabajos de forma digital, se desarrollará un módulo que permita a los técnicos registrar sus trabajos de manera sencilla, evitando la necesidad de hacerlo de manera manual agilizando la entrega de los informes correspondientes.
- Optimizar los procesos de trabajo de los técnicos mediante la automatización y digitalización del seguimiento de los equipos. Se busca eliminar el uso de los

procesos manuales, implementando módulos que agilicen y optimicen las actividades diarias de los técnicos.

- Implementar un método de búsqueda eficiente dentro del sistema, desarrollando un sistema de búsqueda que permita acceder rápidamente a la información requerida, reduciendo el tiempo y esfuerzo dedicados a la búsqueda manual.
- Establecer modulo para generar informes ejecutivos al cliente, se diseñarán reportes que permitan gestionar las consultas y solicitudes de los clientes de manera oportuna y efectiva, mejorando su experiencia de atención con a la empresa.

## **1.5 JUSTIFICACIÓN DEL TEMA**

De acuerdo al problema planteado el presente proyecto se vio conveniente diseñar e implementar un sistema CRM en la empresa EPS S.R.L para optimizar la gestión de clientes además de facilitar el trabajo a los empleados dándoles las opciones de realizarlo desde cualquier lugar en el que se encuentren y así mejorar la imagen de la institución brindando un servicio de calidad, más eficiente e inmediato a los clientes.

### **1.5.1 Justificación Técnica**

Para la implementación de un sistema CRM no será necesario tener equipos costosos, simplemente se necesitarán las computadoras de la empresa la cual cada empleado posee desde la cual tendrán acceso al sistema donde podrán trabajar de forma fácil, rápida y eficiente.

### **1.5.2 Justificación Económica**

La optimización de las tareas y la facilidad de realizarlas desde el sistema mejoraran el servicio dado por la empresa y minimizara los gastos en papelería e impresiones, además para el uso del presente sistema no se necesitara un presupuesto ya que la empresa E.P.S S.R.L. cuenta con los equipos necesarios.

La instalación de las herramientas necesarias para el funcionamiento de dicho sistema tampoco será un gasto en la empresa ya que se están usando en sus versiones “Free” y “open source”.

### **1.5.3 Justificación Social**

El sistema realizara consultas de forma rápida evitando que los clientes tengan que esperar, así mismo se capacitara a los empleados para su adecuado manejo permitiéndoles hacer su trabajo de una manera más cómoda y rápida.

## **1.6 METODOLOGÍA**

### **1.6.1 Scrum**

La metodología Scrum es un marco de trabajo o framework que se utiliza dentro de equipos que manejan proyectos complejos. Es decir, se trata de una metodología de trabajo ágil que tiene como finalidad la entrega de valor en períodos cortos de tiempo y para ello se basa en tres pilares: la transparencia, inspección y adaptación. Esto permite al cliente, junto con su equipo comercial, insertar el producto en el mercado pronto, rápido y empezar a obtener ventas. (Ramírez, 2019)

### **1.6.2 ISO 27001:2022**

El Sistema de Gestión de la Seguridad de la Información o SGSI es una norma internacional que permite el aseguramiento, la confidencialidad e integridad de los datos, hace posible evaluar el riesgo y aplicar los controles imprescindibles para mitigarlos o eliminarlos. (ISO, 2023)

### **1.6.3 ISO 25000**

ISO/IEC 25000, conocida como SQuaRE (System and Software Quality Requirements and Evaluation), es una familia de normas que tiene por objetivo la creación de un marco de trabajo común para evaluar la calidad del producto software.

La familia ISO/IEC 25000 es el resultado de la evolución de otras normas anteriores, especialmente de las normas ISO/IEC 9126, que describe las particularidades de un modelo de calidad del producto software, e ISO/IEC 14598, que abordaba el proceso de evaluación de productos software. (Roa, 2015)

- Prueba de caja negra: Es un método de prueba que se enfoca en la funcionalidad externa de un sistema sin conocer los detalles internos de su diseño o estructura, en este tipo de prueba, los probadores se enfocan en probar la interfaz de usuario, los requisitos funcionales y la entrada y salida esperada del sistema.

El objetivo principal es verificar que el sistema cumpla con los requisitos y expectativas establecidos, independientemente de cómo se implemente internamente. Utilice técnicas como la selección de casos de prueba basada en especificaciones y las pruebas de caja negra para garantizar una amplia cobertura de pruebas.

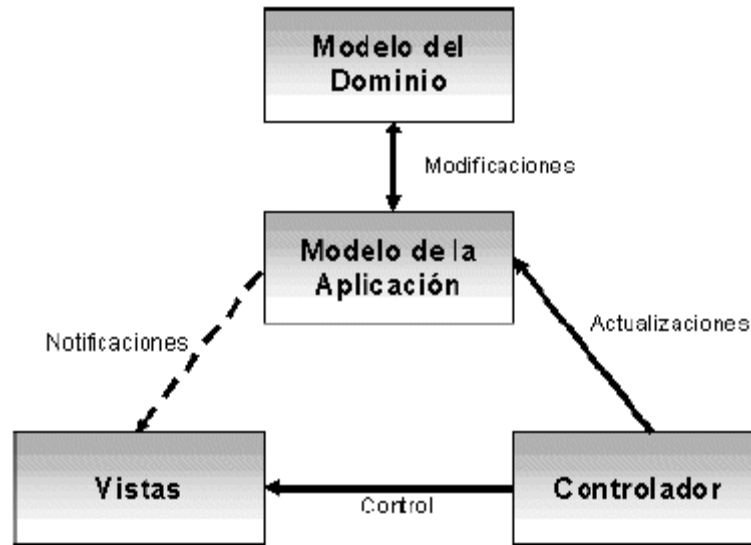
- Pruebas de caja blanca: Se centran en examinar y evaluar la estructura interna de un sistema de software, con este enfoque, el evaluador comprende los detalles internos del diseño, la lógica y el código fuente del sistema.

El objetivo principal de las pruebas de caja blanca es verificar la calidad interna del software, incluida la cobertura del código, la estructura y el flujo de datos. Se utilizan técnicas como pruebas unitarias, pruebas de ruta de decisión y análisis de cobertura de código para identificar posibles errores o vulnerabilidades del sistema. (Roa, 2015)

#### **1.6.4 Arquitectura MVC**

El patrón de arquitectura MVC convierte el desarrollo de aplicaciones complejas en un proceso mucho más manejable. Permite a varios desarrolladores trabajar simultáneamente en la aplicación. (González, 2012)

**Figura 1.2**  
*Arquitectura MVC*



Nota Estructura de la arquitectura modelo vista controlador (Hernández, 2023)

### **1.6.5 Arquitectura de micro servicios**

La arquitectura de microservicios (Micro Services Architecture, MSA) es una aproximación para el desarrollo de software que consiste en construir una aplicación como un conjunto de pequeños servicios, los cuales se ejecutan en su propio proceso y se comunican con mecanismos ligeros (normalmente una API de recursos HTTP). Cada servicio se encarga de implementar una funcionalidad completa del negocio. Cada servicio es desplegado de forma independiente y puede estar programado en distintos lenguajes y usar diferentes tecnologías de almacenamiento de dato. (Babelcube Inc., 2017)

### **1.6.6 Métodos de recolección de datos**

#### **La entrevista**

La entrevista es la más pública de las conversaciones privadas. Funciona con todas las reglas del diálogo privado, pero está construida para el ámbito público. El sujeto entrevistado sabe que se expone a la opinión de la gente. Por otra parte, no es un diálogo libre con dos sujetos. Es una conversación radial, o sea centrada en uno de

los interlocutores, y en la que uno tiene el derecho de preguntar y el otro de ser escuchado. (Folgueiras Bertomeu, 2016)

### **1.6.7 Estimación de costos**

#### **COCOMO II**

En la estimación del tamaño de software COCOMO II utiliza tres técnicas: Puntos Objeto, Puntos Función No Ajustados y Líneas de Código Fuente. Además se emplean otros parámetros relativos al tamaño que contemplan aspectos tales como: reuso, reingeniería, conversión, y mantenimiento. Es necesario unificar criterios de medición de tamaño, tanto para poder planificar y controlar proyectos, como para realizar estudios y análisis entre proyectos en pro de la mejora de procesos. (Gómez, 2010)

### **1.6.8 Pruebas de software**

#### **Prueba de caja negra**

Es el método en el cual el elemento es estudiado desde el punto de vista de las entradas que recibe y las salidas o respuestas que produce, sin tener en cuenta su funcionamiento interno. Estas pruebas son realizadas desde la interfaz gráfica. (Tester Moderno, 2023)

#### **Prueba de caja blanca**

El método de pruebas White box (también conocido como clear box testing, glass box testing, transparent box testing, and structural testing) es el cual mira el código y la estructura del producto que se va a probar y usa ese conocimiento para la realización de las pruebas.

Este método se usa en la fase de Unit testing, aunque también puede ocurrir en otras fases como en las pruebas de sistema o de integración. Para la ejecución de este método es necesario que el tester o la persona que vaya a usar el método tengan

amplios conocimientos de la tecnología y arquitectura usada para desarrollar el programa. (Tester Moderno, 2023)

### **1.6.9 Pruebas de estrés**

Las pruebas de estrés de software también conocidas como pruebas de resistencia son un recurso utilizado durante la etapa de testing de un sistema.

Las pruebas de estrés de software son herramientas diseñadas con el objetivo de probar los límites de un sistema y prever escenarios de riesgo ante cargas extremas como la saturación, pérdida de datos o violación de la seguridad.

Gracias a ello se pueden prevenir errores en etapas finales del ciclo de vida del software, sin poner en juego la calidad y la reputación del proyecto realizado. La clave está en medir la capacidad de respuesta de un sistema en situaciones críticas.

Para ello los desarrolladores simulan el uso del producto informático mediante el envío de peticiones constantes si el sistema no se encuentra en condiciones óptimas responderá con errores o comportamientos anormales, como fallos en el código o bloqueos de la información. (Tamushi, 2022)

## **1.7 HERRAMIENTAS**

### **1.7.1 Lenguaje de programación**

#### **Java**

Java es un lenguaje de programación orientado a objetos el cual es uno de los tres más utilizados en la actualidad. Gracias al amplio soporte con el que cuenta, así como también con la gran variedad de clases y colecciones que tiene; Java es uno de los lenguajes más robustos y utilizados en el mundo del desarrollo de software multiplataforma.

Lo anterior debido a que permite al desarrollador el crear aplicaciones o sistemas de información locales, aplicaciones o sistemas dentro de ambientes web e incluso aplicaciones para móviles. Con lo cual se consolida como uno de los mejores lenguajes



de programación en la actualidad y en un futuro quizá el lenguaje de programación universal. (Corcuera, Introducción a la tecnología Java., 2017)

### **1.7.2 Framework**

#### **Java FX**

JavaFX es una plataforma de desarrollo de software y un conjunto de herramientas de interfaz de usuario para la creación de aplicaciones de escritorio en el lenguaje de programación Java. Esta plataforma proporciona un conjunto completo de bibliotecas para la creación de gráficos, animaciones, efectos visuales y una amplia variedad de componentes de interfaz de usuario. Además, JavaFX permite una integración fácil con otras tecnologías de Java, como Swing y Java 2D. JavaFX también es conocido por su capacidad para crear aplicaciones altamente interactivas y visualmente atractivas. (Corcuera, Introducción a la tecnología Java., 2017)

### **1.7.3 Jasper Soft**

Jaspersoft es una suite de Business Intelligence cuyo núcleo es software open source. La plataforma de BI proporciona herramientas para la creación de una capa de metadatos sobre la que se crean informes de detalle, analíticos o cuadros de mando. La versión Community (Open source) está compuesta por el servidor de BI JasperReports server, el entorno de desarrollo de informes basado en Eclipse Jaspersoft Studio, la librería para creación de informes de detalle JasperReports Library y la herramienta para creación de procesos de carga del data warehouse Jaspersoft ETL. Las versiones empresariales de la plataforma se orientan a la gestión completa desde web, a proporcionar también herramientas fáciles de utilizar para creación de cuadros de mando, BI de autoservicio o informes para incrustar en otras aplicaciones. (TIBCO, 2023)

### **1.7.4 Lenguaje de diseño**

#### **Css**

El CSS (hojas de estilo en cascada) es un lenguaje que define la apariencia de un documento escrito en un lenguaje de marcado (por ejemplo, HTML). Así, a los

elementos de la página web creados con HTML se les dará la apariencia que se desee utilizando CSS: colores, espacios entre elementos, tipos de letra, separando de esta forma la estructura de la presentación.

Esta separación entre la estructura y la presentación es muy importante, ya que permite que sólo cambiando los CSS se modifique completamente el aspecto de una página web. Esto posibilita, entre otras cosas, que los usuarios puedan usar hojas de estilo personalizadas (como hojas de estilo de alto contraste o de accesibilidad). (Durango, 2015)

### **1.7.5 Scene Builder**

Scene Builder es una herramienta de diseño de interfaces gráficas de usuario para JavaFX. Con esta herramienta, los desarrolladores pueden crear y diseñar fácilmente la interfaz de usuario de sus aplicaciones de manera visual y sin tener que escribir código. Scene Builder permite arrastrar y soltar componentes de interfaz de usuario, como botones, cajas de texto y etiquetas, y personalizar sus propiedades. Además, ofrece una vista previa en tiempo real de cómo se verá la interfaz de usuario en tiempo de ejecución. Scene Builder es compatible con cualquier IDE de Java, incluyendo NetBeans, Eclipse y IntelliJ IDEA. (Gluon, 2023)

### **1.7.6 Editor de código**

#### **IntelliJ IDEA**

IntelliJ IDEA es un entorno de desarrollo integrado (IDE, por sus siglas en inglés) para la programación en Java y otros lenguajes de programación, como Kotlin, Groovy y Scala. IntelliJ IDEA es conocido por su gran cantidad de características y herramientas avanzadas que ayudan a los desarrolladores a escribir código de manera más rápida y eficiente. Algunas de sus características incluyen la finalización de código, refactorización de código, análisis estático, integración con control de versiones, y depuración avanzada. IntelliJ IDEA también es altamente personalizable y ofrece una gran variedad de plugins para una experiencia de programación aún más personalizada. (Jet Brains, 2023)

### **1.7.7 Base de datos**

#### **Oracle database**

Oracle es una empresa que proporciona soluciones de software y hardware para empresas en todo el mundo. Oracle Database es un sistema de gestión de bases de datos relacional que se utiliza ampliamente en el mundo empresarial. Oracle Database es conocido por su alta escalabilidad, fiabilidad y seguridad. Ofrece una amplia variedad de herramientas para administrar bases de datos, como Oracle Enterprise Manager y SQL Developer. Además, Oracle Database admite múltiples lenguajes de programación, incluyendo Java, PL/SQL y Python. Es compatible con múltiples sistemas operativos y también se puede ejecutar en la nube a través de Oracle Cloud. (Oracle, 2023)

## **1.8 LÍMITES Y ALCANCES**

### **1.8.1 Limites**

- El sistema CRM se enfocará en administrar, gestionar y controlar los procesos específicos de la empresa, como la gestión de clientes, seguimiento de trabajos realizados por técnicos, generación de informes, etc.
- El sistema no podrá verificar el contenido de información ingresada por los técnicos.
- El sistema ofrecerá un buscador en la base de datos para acceder y recuperar información
- El sistema tendrá acceso a datos de los clientes sin embargo no podrá realizar funciones de contabilidad.


### **1.8.2 Alcances**

El sistema propone una manera más veloz y optima de trabajar, de esta manera se propone los siguientes módulos:

- Módulo de usuario, el administrador podrá controlar a todos los usuarios y los privilegios que estos tengan dentro del sistema evitando así cambios dentro de la base de datos.
- Módulo de taller, se ocupa de almacenar y consultar los reportes laborales de los empleados mediante el llenado de formularios además de administrar todo el inventario de la empresa, se registrará las entradas salidas de los equipos que llegan para su reparación.
- Módulo de técnicos,
- Módulo de clientes, se ocupa del registro de nuevos clientes, así como la edición o eliminación de clientes ya almacenados.
- Módulo de equipos, En este módulo se registran los equipos en los que trabajaran los técnicos como ser sus nombres, marcas, fallas, etc.,
- Módulo de servicios, Se administran los servicios dados por los técnicos, en este módulo se relacionan los módulos: técnicos, equipos, clientes
- Módulo de reportes, En este módulo se podrá generar reportes técnicos y o ejecutivos ya sea para los empleados o para los clientes.
- Interfaz gráfica, que se adapte a cualquier dispositivo móvil, de fácil comprensión y uso.
- Un buscador, en el cual se podrá buscar la información que sea necesaria sobre algún cliente o equipo en el inventario de la empresa.
- Módulo de Dashboard, En el cual los empleados podrán revisar una lista de servicios pendientes que tienen con los clientes de la empresa.

## **1.9 APORTES**

El principal aporte es crear un sistema para mejorar y optimizar la calidad de trabajo de los empleados permitiendo que estos puedan realizar sus servicios de forma más rápida y precisa además de mejorar la atención a los clientes entregando informes ejecutivos de manera rápida.



# CAPÍTULO II MARCO TEÓRICO

## **2. MARCO TEÓRICO**

En este capítulo se darán a conocer las definiciones de las metodologías, herramientas y técnicas que serán utilizadas en cada etapa de desarrollo del proyecto que se dividirá en diferentes fases como ser: análisis, diseño, implementación y pruebas.

### **2.1 CONCEPTOS BÁSICOS**

#### **2.1.1 Dato**

Un dato es una información numérica, alfabética o simbólica que se registra y almacena en un sistema de información o en una base de datos. Los datos pueden ser de diferente tipo, como fechas, nombres, números y cualquier otra información que permita la identificación de algo. (Carrion, 2017)

Un dato es una representación simbólica de un hecho, una entidad o un atributo. Los datos son la unidad básica de información, puede ser procesado y transformado para obtener información. Los datos pueden ser numéricos, alfabéticos, alfanuméricos, fechas, entre otros.

#### **2.1.2 Información**

La información es un conjunto de datos y conocimientos significativos que se obtienen a partir de experiencias, investigaciones o hechos y que son utilizados para transmitir conocimientos, tomar decisiones o resolver problemas. (Pontis, 2011).

Es el resultado del procesamiento y análisis de los datos, es el conjunto de datos organizados y estructurados de manera significativa y útil para una determinada finalidad, la información proporciona conocimiento y permite tomar decisiones informadas.

#### **2.1.3 Sistema**

Un sistema informático es un conjunto de elementos físicos y lógicos capaces de guardar y procesar información. La parte física alude al hardware (todo lo tangible) y la parte lógica alude al software (todo lo intangible)

Los sistemas informáticos son importantes debido a que nos permiten acceder a diversas funcionalidades útiles de cualquier ámbito. Así mismo, a nivel empresarial, se encargan de automatizar procesos, delegar actividades repetitivas y abaratar costes. (Chavez, 2020).

Algunos ejemplos de sistemas informáticos son:

- Ordenadores.
- Teléfonos celulares.
- Tablets.
- Relojes inteligentes.
- Cámaras de seguridad.
- Cámaras fotográficas.
- Sistemas expertos.
- Agentes inteligentes.
- Sistemas de posicionamiento global (GPS).
- Sistemas de alarma.
- Robots industriales.
- Consolas de videojuegos.

#### **2.1.4 Seguimiento y control**

Se refiere a las herramientas digitales que sirven para monitorear la ejecución y evolución de un proyecto. Todo esto se realiza teniendo en cuenta la planificación previa, los documentos relacionados, el registro de las actividades y la evolución de las mismas.

En la actualidad, las firmas y departamentos legales cuentan con herramientas de teletrabajo especializadas en el seguimiento y control de procesos, las cuales automatizan gran parte de estas tareas permitiendo que los abogados puedan seguir causas judiciales vía online sin tener que reparar en los aspectos técnicos y repetitivos del monitoreo (LemonTech, 2021).

#### **2.1.5 Gestión de clientes**

Para cualquier empresa o negocio el cliente siempre es lo más importante, la gestión de clientes es el proceso conformado por un conjunto de tácticas que van orientadas

a la satisfacción del cliente, la gestión de clientes es un proceso fundamental donde se sostendrá el proceso y desarrollo del proyecto.

El objetivo principal de la gestión de clientes es mejorar las relaciones con los clientes, lo que puede llevar a una mayor fidelidad y retención de los mismos, así como a un aumento en las ventas y la rentabilidad dentro de la empresa.

En el sistema desarrollado en Java FX y base de datos Oracle la gestión de clientes podría incluir la recopilación de información de los clientes como su número telefónico, historiales de servicios realizados en la empresa, registros de interacciones con el servicio al cliente. La integración de esta información en el sistema CRM desarrollado podrá ayudar a la empresa a mejorar sus relaciones con los clientes y aumentar la eficiencia de los procesos.

#### **2.1.6 CRM**

CRM son las iniciales para Customer Relationship Management, o, en español, gestión de relacionamiento con el cliente. En la práctica, es un software que auxilia en la gestión de los clientes por la empresa, trayendo beneficios como organización, control comercial en la base de datos.

No se puede hablar de CRM sin tocar el tema de la organización. Esta es una de las principales ventajas de utilizar un sistema CRM, porque el software trae un panel con toda la información importante sobre el historial de Leads y clientes, además de los objetivos.

Tener esta información organizada facilita la gestión de los clientes, ya que los datos no están repartidos en varios archivos, lo que ocurre en el control de planillas de Excel, por ejemplo. Todo está reunido en un solo lugar (RD Station, 2023).

#### **2.1.7 Microservicios**

Una arquitectura de microservicios consta de una colección de servicios autónomos y pequeños. Cada uno de servicio es independiente y debe implementar una funcionalidad de negocio individual dentro de un contexto delimitado. Un contexto delimitado es una división natural de una empresa y proporciona un límite explícito dentro del cual existe un modelo de dominio. (Microsoft, 2023).



¿Qué son los microservicios?

- Los microservicios son pequeños e independientes, y están acoplados de forma imprecisa. Un único equipo reducido de programadores puede escribir y mantener un servicio.
- Cada servicio es un código base independiente, que puede administrarse por un equipo de desarrollo pequeño.
- Los servicios pueden implementarse de manera independiente. Un equipo puede actualizar un servicio existente sin tener que volver a generar e implementar toda la aplicación.
- Los servicios son los responsables de conservar sus propios datos o estado externo. Esto difiere del modelo tradicional, donde una capa de datos independiente controla la persistencia de los datos.
- Admite la programación políglota. Por ejemplo, no es necesario que los servicios compartan la misma pila de tecnología, las bibliotecas o los marcos.

### **2.1.8 Gestión de clientes**

Con la gestión de clientes se puede crear una buena imagen de marca, porque muestra la importancia para los consumidores, la calidad del servicio y la flexibilidad del proceso de servicios

La ventaja de la gestión de clientes para las empresas es también la posibilidad de producir parámetros de calidad y mejorar la atención de los equipos y los procesos de servicio. Como resultado, puede esperar ahorros operativos y mejores registros de trabajo.

### **2.19 Modelo de negocio**

Los modelos de negocio son un plan organizado que busca establecer orden y disciplina en el complicado proceso de concebir, crecer y administrar un negocio además de generar valor para sus clientes y cómo obtienen una recompensa a cambio. Un modelo de negocio comprende el producto o servicio, el cliente y el mercado, el papel de la empresa en la cadena de valor y el motor económico que le permite lograr rentabilidad y crecimiento. Los startups suelen emplear los modelos de negocio como herramientas complementarias para el diseño, la creación de prototipos y la organización de sus nuevas operaciones. Las empresas establecidas también los

utilizan para planificar, desarrollar y mejorar sus procesos de innovación. (Turner, 2015) .

## 2.2 Metodologías

### 2.2.1 Scrum

**Figura 2.1**  
*Procesos de Scrum*



Nota Las 5 fases para implementar la metodología Scrum (Mesa, 2022)

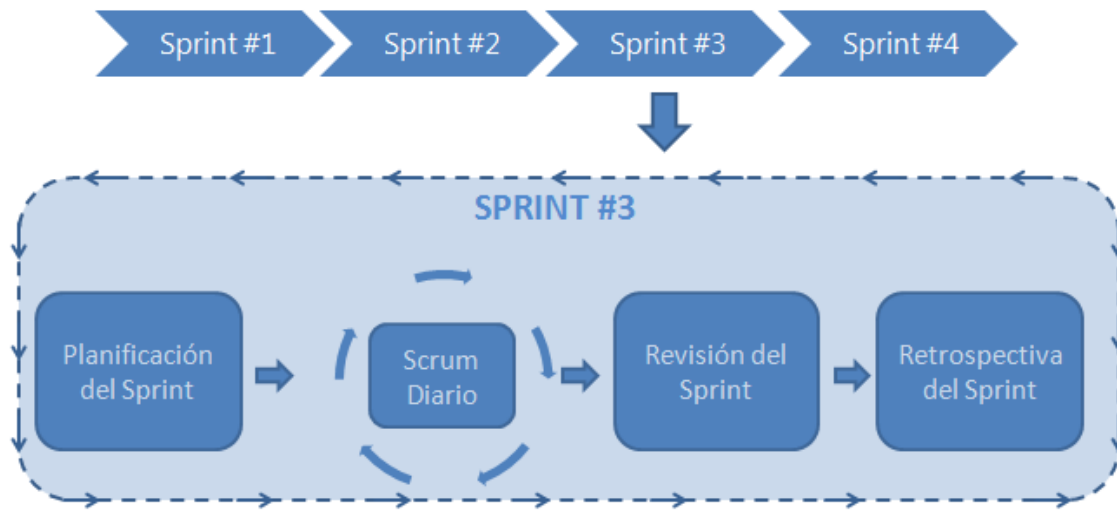
La metodología Scrum es ágil puede ser aplicada a proyectos de diversos segmentos, actualmente es muy usada por empresas de tecnología y desarrollo de software.

Scrum también se utiliza para resolver situaciones en que no se está entregando al cliente lo que necesita, cuando las entregas se alargan demasiado, los costes se disparan o la calidad no es aceptable, cuando se necesita capacidad de reacción ante la competencia, cuando la moral de los equipos es baja y la rotación alta, cuando es necesario identificar y solucionar ineficiencias sistemáticamente o cuando se quiere trabajar utilizando un proceso especializado en el desarrollo de producto (Dimes, 2015).

## Etapas de la metodología SCRUM

- Planificación del sprint

**Figura 2.2**  
*Elementos del sprint*



Nota: Ciclo de vida de un sprint en la metodología Scrum (Mesa, 2022)

En la primera etapa de la metodología scrum se definirán aspectos como la funcionalidad, objetivos, riesgos del sprint, plazos de entrega, entre otros posteriormente se realiza una junta entre el equipo y el jefe del proyecto para explicar cómo se desarrollará cada punto del intervalo. Aquí se evaluarán cambios, toma de decisiones, mejoras y más factores (Dimes, 2015).

- Etapa de desarrollo

Cuando el trabajo del sprint está en curso, se deben garantizar que no se generen cambios de último momento que puedan afectar los objetivos del mismo. Por otra parte, se asegura el cumplimiento de los plazos establecidos para su término (Dimes, 2015).

- Revisión del sprint

Al final del desarrollo del intervalo, es posible analizar y evaluar los resultados si se ve necesario se hará un análisis para saber qué aspectos necesitan ser

cambiados. Es en esta parte de la metodología que se fomenta la retroalimentación (Dimes, 2015).

➤ **Retroalimentación**

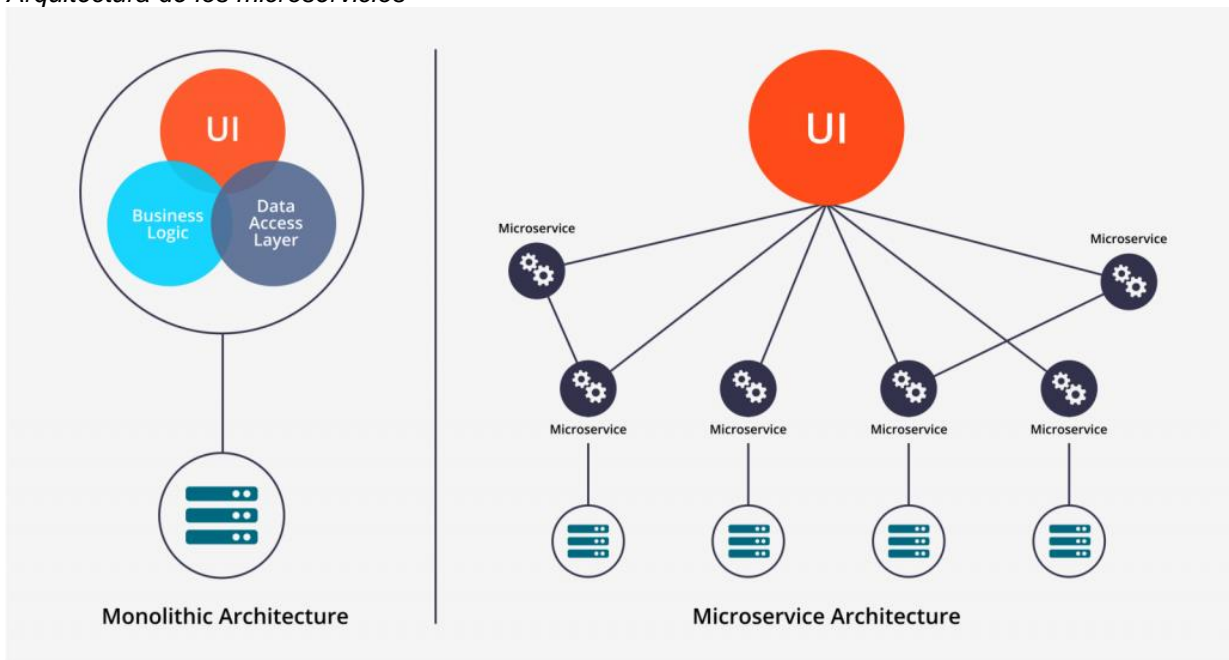
Los resultados pueden ser entregados para recibir un feedback no solo por parte de los profesionales dentro del proyecto, sino también de las personas que utilizarán directamente lo que se desea lograr; es decir, los clientes potenciales. Las lecciones aprendidas durante esta etapa permitirán que el siguiente sprint pueda ser mucho más efectivo y ágil. (Dimes, 2015).

### 2.2.3 Arquitectura de microservicios

En el presente proyecto, se ha adoptado la arquitectura de microservicios para su desarrollo y funcionamiento, los microservicios son una arquitectura de desarrollo que permite dividir una aplicación en componentes independientes y autónomos, denominados microservicios, cada microservicio se encarga de una funcionalidad específica y se comunica con otros microservicios a través de interfaces.

**Figura 2.3**

*Arquitectura de los microservicios*



Nota: Comparación entre una arquitectura monolítica y microservicios (Hiberus, 2022)

La arquitectura de microservicios (del inglés Micro Services Architecture) es un método de desarrollo de software que consiste en construir una aplicación como un conjunto de pequeños servicios, con operaciones bien definidas e independientes entre sí.

Cada microservicio ejecuta su propio proceso y se encarga de implementar una funcionalidad completa del negocio. Puede estar programado en distintos lenguajes y usar diferentes tecnologías de almacenamiento de datos. A la hora de hacer un despliegue, cada servicio se hace de forma independiente. (López, 2017).

### **2.2.3 Ciclo de vida de los microservicios**

El ciclo de vida de los microservicios es un enfoque de desarrollo de software que se centra en la creación, lanzamiento y expansión de servicios independientes y altamente cohesivos. El ciclo de vida de los microservicios consta de varias etapas, desde el diseño y la implementación hasta el lanzamiento y la expansión.

El ciclo de vida de los microservicios es importante porque permite a las organizaciones desarrollar e implementar servicios de manera más rápida y eficiente. Al separar el software en servicios más pequeños e independientes, se reduce la complejidad y se facilita el mantenimiento y la actualización de los servicios. Además, al adoptar un enfoque modular, las organizaciones pueden escalar y actualizar cada microservicio de manera independiente para adaptarse a los cambios en las necesidades del negocio. (López, 2017).

- **Lanzamiento**

En esta fase se debe confirmar la estrategia definida para el desarrollo en base a micro servicios así como identificar los cambios necesarios, además se debe preparar los fundamentos para aprovechar el máximo la fase de expansión y obtener los beneficios prometidos.

- **Expansión**

La fase de expansión debe centrarse en el control y gestión de la expansión del uso de los microservicios que es el punto clave para obtener los beneficios prometidos

## 2.2.4 Arquitectura MVC

El patrón de diseño Modelo Vista Controlador (MVC) es una de las arquitecturas más utilizadas en el desarrollo de aplicaciones. MVC se enfoca en separar la aplicación en tres componentes principales: el modelo, que representa la lógica de negocio y los datos de la aplicación; la vista, que es la interfaz de usuario que presenta los datos del modelo al usuario; y el controlador, que es el intermediario entre la vista y el modelo y se encarga de gestionar las acciones del usuario y actualizar el modelo y la vista en consecuencia. (Hervella, 2015).

### ➤ Modelo

El modelo define qué datos debe contener la aplicación. Si el estado de estos datos cambia, el modelo generalmente notificará a la vista (para que la pantalla pueda cambiar según sea necesario) y, a veces, el controlador (si se necesita una lógica diferente para controlar la vista actualizada).

El modelo especificará qué datos deben contener los módulos a desarrollar

### ➤ Vista

La vista define cómo se deben mostrar los datos de la aplicación.

En nuestra aplicación, la vista definiría cómo se presenta la lista de usuarios, técnicos pantallas de navegación entre otros y recibiría los datos para mostrar desde el modelo.

### ➤ Controlador

El controlador contiene una lógica que actualiza el modelo y/o vista en respuesta a las entradas de los usuarios de la aplicación.

Entonces, por ejemplo, nuestra lista de compras podría tener formularios de entrada y botones que nos permitan agregar o eliminar artículos. Estas acciones requieren que

se actualice el modelo, por lo que la entrada se envía al controlador, que luego manipula el modelo según corresponda, que luego envía datos actualizados a la vista.

### **Ventajas:**

- La implementación se realiza de forma modular.
- Sus vistas muestran información actualizada siempre. El programador no debe preocuparse de solicitar que las vistas se actualicen, ya que este proceso es realizado automáticamente por el modelo de la aplicación.
- Cualquier modificación que afecte al dominio, como aumentar métodos o datos contenidos, implica una modificación sólo en el modelo y las interfaces del mismo con las vistas, no todo el mecanismo de comunicación y de actualización entre modelos.
- Las modificaciones a las vistas no afectan al modelo de dominio, simplemente se modifica la representación de la información, no su tratamiento.
- MVC está demostrando ser un patrón de diseño bien elaborado pues las aplicaciones que lo implementan presentan una extensibilidad y una mantenibilidad únicas comparadas con otras aplicaciones basadas en otros patrones.

### **Desventajas:**

- Para desarrollar una aplicación bajo el patrón de diseño MVC es necesario una mayor dedicación en los tiempos iniciales del desarrollo. Normalmente el patrón exige al programador desarrollar un mayor número de clases que, en otros entornos de desarrollo, no son necesarias. Sin embargo, esta desventaja es muy relativa ya que posteriormente, en la etapa de mantenimiento de la aplicación, una aplicación MVC es mucho más mantenible, extensible y modificable que una aplicación que no lo implementa.
- MVC requiere la existencia de una arquitectura inicial sobre la que se deben construir clases e interfaces para modificar y comunicar los módulos de una aplicación. Esta arquitectura inicial debe incluir, por lo menos, un mecanismo de eventos para poder proporcionar las notificaciones que genera el modelo de

aplicación; una clase Modelo, otra clase Vista y una clase Controlador genéricas que realicen todas las tareas de comunicación, notificación y actualización que serán luego transparentes para el desarrollo de la aplicación.

- MVC es un patrón de diseño orientado a objetos por lo que su implementación es sumamente costosa y difícil en lenguajes que no siguen este paradigma.

### **2.2.5 Customer relationship management “CRM”**

Un CRM es una solución de gestión de las relaciones con clientes, orientada normalmente a gestionar tres áreas básicas: la gestión comercial, el marketing y el servicio postventa o de atención al cliente.

El uso de un CRM forma parte de una estrategia orientada al cliente en la cual todas las acciones tienen el objetivo final de mejorar la atención y las relaciones con clientes y potenciales. La herramienta CRM y la orientación al cliente proporcionan resultados demostrables, tanto por disponer de una gestión comercial estructurada y que potencia la productividad en las ventas como por ofrecer un conocimiento profundo del cliente que permite plantear campañas de marketing más efectivas.

Las funciones de atención al cliente de una herramienta CRM potencian además la fidelización y satisfacción de los clientes, lo que tiene un impacto muy positivo en términos de ventas recurrentes y cruzadas. (Reinares, 2017).

Customer Relationship Management (CRM) es un enfoque de gestión empresarial que se centra en la creación y mantenimiento de relaciones sólidas y duraderas con los clientes.

El objetivo principal del CRM es mejorar la satisfacción del cliente y aumentar la lealtad, lo que, a su vez, puede generar más ingresos y rentabilidad para la empresa. Las herramientas y técnicas de CRM incluyen la recopilación y análisis de datos de clientes, la segmentación de clientes, el seguimiento de las interacciones con los clientes y la personalización de la experiencia del cliente.



Estas herramientas ayudan a las empresas a comprender mejor las necesidades y deseos de los clientes y a proporcionar una experiencia de servicio excepcional.

Esta estrategia se basa en la recolección, análisis y uso efectivo de los datos del cliente para mejorar la satisfacción del mismo, aumentar la fidelidad, y en última instancia, aumentar los ingresos de la empresa.

La implementación de un sistema CRM en una empresa permite centralizar la información de los clientes en una sola base de datos, lo que permite un acceso rápido y fácil a la información relevante. Además, permite a la empresa realizar un seguimiento de las interacciones del cliente y personalizar las comunicaciones y ofertas, lo que mejora la relación entre la empresa y el cliente.

El éxito de un sistema CRM depende de la calidad de la información recolectada, la integración con otros sistemas de la empresa, la capacidad de personalización y el uso efectivo de los datos. Además, la adopción del sistema por parte de los empleados y la gestión efectiva del cambio son factores clave para una implementación exitosa. (Cali, 2015).

### **Características del CRM:**

Un software de Customer Relationship Management ayuda a automatizar y gestionar el ciclo de vida del cliente para los departamentos de ventas y marketing.

El programa consolida la información del cliente en un único repositorio para que los encargados puedan organizar y gestionar mejor las relaciones, automatizar procesos comunes, y monitorear el rendimiento o la productividad. (Ortegón, 2019).

#### ➤ **Gestión de contactos**

Nos permite almacenar información de contacto de todos tus leads o prospectos, como nombres, direcciones y cuentas de redes sociales, en una base de datos de fácil búsqueda. (Ortegón, 2019).

#### ➤ **Seguimiento de interacción**

Las plataformas que contemplan a mayor profundidad la interacción entre Marketing y Ventas, como el CRM de Hubspot, permiten rastrear el historial de interacción para documentar conversaciones con contactos específicos. (Ortegón, 2019).

➤ **Lead scoring**

Similar a la función de gestión de contactos, la clave aquí es garantizar que el proceso de gestión de leads fluya y transite de una manera que tenga sentido para el negocio. (Ortegón, 2019).

➤ **Integración de correo electrónico**

Integrar el correo electrónico permite la gestión y segmentación de listas de contactos, así como crea una base centralizada para las comunicaciones tanto internas como externas. (Ortegón, 2019).

➤ **Gestión documental**

La administración de documentos es una excelente característica de CRM, si tienes un gran equipo involucrado en tus acciones de marketing y ventas. Recopilar, cargar, almacenar y compartir documentos en una plataforma centralizada, lo que facilita el acceso de todos los involucrados a la información. (Ortegón, 2019).

➤ **Funciones de ventas**

Si la naturaleza de sus productos o servicios a menudo requiere de idas y vueltas sobre contratos, cotizaciones y propuestas, entonces implementar ese proceso dentro del software CRM es una característica muy útil para desarrollar los procesos de ventas. (Ortegón, 2019).

➤ **Gestión del pipeline de ventas**

Si tiene un ciclo de ventas muy largo, puede ser una buena idea priorizar esta función al adquirir un buen software de CRM. Tener una vista panorámica de todo tu pipeline de ventas, ayudará a evitar que los prospectos se detengan o abandonen el proceso. (Ortegón, 2019).

Automatización del flujo de trabajo, es el diseño y la administración de procesos automatizados para completar tareas o para enrutar datos y archivos entre personas y sistemas basados en reglas comerciales. Las plataformas con

mayor gama de funciones, permiten la automatización de las tareas repetitivas, creando flujos de trabajo que activen acciones o envíen recordatorios de seguimiento para las próximas acciones. (Ortegón, 2019).

➤ **Informes**

Con la funcionalidad de informes o análisis en tiempo real, puedes generar documentos basados en factores como la geografía, la etapa del ciclo en la que se encuentra el potencial cliente, e incluso reportes un miembro individual del equipo de ventas. (Ortegón, 2019).

### **Fases para implementar CRM**

➤ Conoce tu proceso de venta.

Se debe describir con detalle cómo es el proceso de venta, es decir, cómo se realiza la comunicación con el cliente y la comunicación con tu equipo. (Luque, 2017).

➤ Identifica las necesidades que debe cubrir el sistema CRM.

Se debe identificar cuáles son las necesidades del CRM, desde las necesidades más técnicas de la entidad como disponer de un call center propio o disponer de un sistema de alertas para el seguimiento hasta las necesidades más operativas (Luque, 2017).

➤ Desarrollar un plan de automatización del CRM.

Una vez que se haya identificado las necesidades de tu entidad debes desarrollar un plan de automatización. El objetivo de esta etapa es establecer los pasos a seguir para cubrir las necesidades. (Luque, 2017).

➤ Analizar y dar seguimiento

La automatización de un proceso de ventas requiere un análisis y una correcta interpretación de los datos. El CRM no es más que una potente herramienta pensada para aumentar el beneficio de una empresa. (Luque, 2017).

## 2.2.6 Tipos de CRM

Actualmente existen cuatro tipos de CRM: CRM Operacional, Analítico, Colaborativo y Estratégico. Si bien todos ellos están interesados en unificar los datos de los clientes en un solo lugar y hacer que los procesos internos sean más eficientes, cada uno de ellos tiene una característica destacada que los hace más adecuados para contextos de uso específicos. (Silva, 2023).

### ➤ CRM Operacional

El CRM Operacional es un sistema enfocado a optimizar los procesos relacionados con el cliente. Su objetivo es unificar las actividades que realizan los diferentes departamentos en una interfaz compartida, con el fin de incrementar la colaboración y la productividad.

Como característica destacada de los tipos de CRM, el CRM Operacional suele ofrecer herramientas de automatización de procesos para que los profesionales puedan manejar tareas repetitivas con más eficiencia y en menos pasos.

### ➤ CRM Analítico

El CRM Analítico es un sistema enfocado en recopilar, almacenar y analizar todos los datos que una empresa procesa. Su objetivo es transformar grandes cantidades de datos en conocimientos útiles para tu estrategia de negocio.

Como característica destacada de los tipos de CRM, el CRM Analítico puede ayudarte a comprender los clientes en profundidad: su ubicación, edad, sexo, estado civil, ingresos, hábitos de compra y mucho más.

### ➤ CRM Colaborativo

El CRM Colaborativo es un sistema que alinea recursos y estrategias en diferentes departamentos a través de una interfaz unificada. Es decir, el CRM colaborativo se encuentra entre los tipos de CRM que garantizan una comunicación interna fluida, donde todos los profesionales tienen acceso a los datos que necesitan.

Como característica destacada de los tipos de CRM, el CRM Colaborativo cuenta con muchas herramientas de integración — tanto entre personas como con otras tecnologías que utiliza en tu negocio. Por ejemplo, el CRM de ventas de Zendesk puede integrarse con Mailchimp para enviar email marketing.

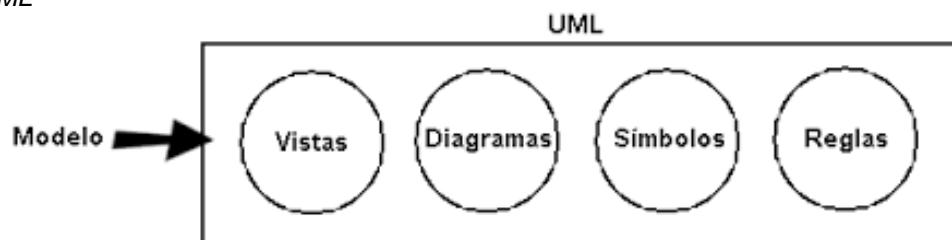
### ➤ CRM Estratégico

El CRM Estratégico es un sistema centrado en el cliente, cuyo objetivo es adquirir y retener clientes a largo plazo. Para ello, utiliza datos que revelan las preferencias, el comportamiento y las tendencias de los clientes de una empresa.

Como característica destacada de los tipos de CRM, el CRM Estratégico se enfoca en las necesidades del cliente y busca la mejor manera de satisfacer sus expectativas.

## 2.2.7 UML

**Figura 2.4**  
*Modelo UML*



Nota Estructura del modelo UML (Hans-Erik, 2023).

El Proceso Unificado es un proceso de realización o de evolución de software enteramente basado en UML, de ahí el interés de presentarlo en esta obra. Está constituido por un conjunto de directivas que permiten producir software a partir del pliego de condiciones, cada directiva define quién hace qué y en qué momento. Un proceso permite, por tanto, estructurar las diferentes etapas de un proyecto informático.

Un proyecto que emplee UML puede utilizar otro proceso diferente del Proceso Unificado o no emplear ninguno, el Proceso Unificado es conducido por los casos de uso estos se utilizan para describir los requisitos del proyecto y se describen con ayuda de una representación específica del Proceso Unificado más rica que la contenida en UML.

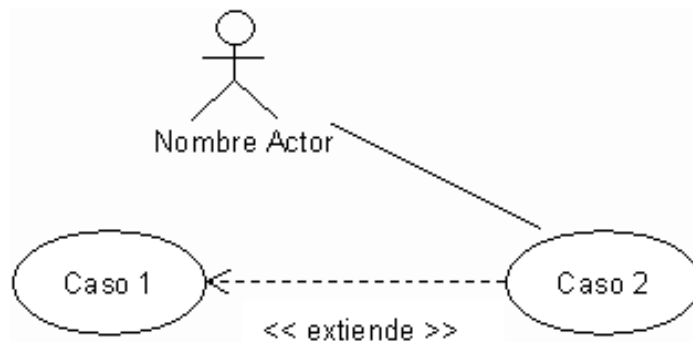
El lenguaje unificado de modelado es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad, respaldado por el Object Management

Group. Es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema. (Debrauwer, 2016).

Los diagramas de clases ofrecen una serie de beneficios para toda organización. Usa los diagramas de clases UML para:

- Ilustrar modelos de datos para sistemas de información, sin importar qué tan simples o complejos sean.
- Comprender mejor la visión general de los esquemas de una aplicación.
- Expresar visualmente cualesquier necesidades específicas de un sistema y divulgar esa información en toda la empresa.
- Crear diagramas detallados que resalten cualquier código específico que será necesario programar e implementar en la estructura descrita.
- Ofrecer una descripción independiente de la implementación sobre los tipos empleados en un sistema que son posteriormente transferidos entre sus componentes.

**Figura 2.5**  
*Ejemplo de diagramas UML*



Nota Ejemplo básico de un diagrama UML (Debrauwer, 2016).

## 2.3 HERRAMIENTAS

### 2.3.1 Lenguaje de programación Java

Java es un lenguaje de propósito general similar a cualquier otro y con características que lo hacen idóneo para programación orientada a objetos,

Un applet es realmente un pequeño programa Java, descargado dinámicamente por la red, tal como una imagen, un archivo de sonido, un archivo musical MP3 o divX, o una secuencia de vídeo: pero con una notable propiedad, es un programa inteligente que puede reaccionar dinámicamente a entradas y cambios del usuario. Java es un lenguaje idóneo para resolver los problemas de seguridad y portabilidad inherentes a los sistemas que trabajan en red. La razón fundamental de este aserto reside en el hecho de que la salida de un compilador Java no es un código ejecutable, sino códigos de bytes (bytecode). (Abenza, 2015).

### **2.3.2 Framework Java Fx**

Java FX es un conjunto de herramientas y librerías de programación para la creación de interfaces gráficas de usuario (GUI, por sus siglas en inglés) en aplicaciones de escritorio. Java FX es parte de la plataforma Java y ha sido desarrollado por Oracle. (Oracle, 2021).

Es utilizado en aplicaciones de escritorio, web y móviles utilizando el lenguaje de programación Java. JavaFX proporciona una amplia variedad de componentes y controles de interfaz de usuario, así como una API para la manipulación de gráficos 2D y 3D. Además, permite la integración de efectos visuales, animaciones y multimedia en las interfaces de usuario. JavaFX también es compatible con múltiples plataformas, incluyendo Windows, Linux, macOS y dispositivos móviles.

### **2.3.3 Lenguaje de diseño gráfico CSS**

Es un lenguaje de diseño gráfico para definir y crear la presentación de un documento estructurado escrito en un lenguaje de marcado. Actualmente es un lenguaje muy usado para establecer el diseño visual de los documentos e interfaces de usuario escritas en HTML o XHTML; el lenguaje puede ser aplicado a cualquier documento XML, incluyendo XHTML, SVG, XUL, RSS, etcétera. Junto con HTML y JavaScript, CSS es una tecnología usada por muchos sitios web para crear páginas visualmente atractivas, interfaces de usuario para aplicaciones web y GUIs para muchas aplicaciones.

CSS está diseñado principalmente para marcar la separación del contenido del documento y la forma de presentación de este, características tales como las capas o layouts, los colores y las fuentes.<sup>4</sup> Esta separación busca mejorar la accesibilidad del documento, proveer más flexibilidad y control en la especificación de características presentacionales, permitir que varios documentos HTML compartan un mismo estilo usando una sola hoja de estilos separada en un archivo .css, y reducir la complejidad y la repetición de código en la estructura del documento. (Tinoco, 2014).

### **2.3.4 Entorno de desarrollo IntelliJ IDEA**

IntelliJ IDEA es un entorno de desarrollo integrado (IDE) para el desarrollo de programas informáticos. Es desarrollado por JetBrains (anteriormente conocido como IntelliJ), y está disponible en dos ediciones: edición para la comunidad y edición comercial.

Aunque está diseñado principalmente para el desarrollo de Java y Kotlin, IntelliJ IDEA entiende muchos otros lenguajes de programación, incluyendo Groovy, Scala, JavaScript, TypeScript y SQL, y proporciona asistencia inteligente a la codificación para cada uno de ellos. La indexación inicial del código fuente permite al IDE crear un mapa virtual de su proyecto. Gracias a la información del mapa virtual, puede detectar errores sobre la marcha, sugerir variantes de finalización de código con un conocimiento del contexto preciso, realizar refactorizaciones y más.

En un informe de Infoworld en 2010, IntelliJ recibió la puntuación más alta entre las cuatro mejores herramientas de programación de Java: Eclipse, IntelliJ IDEA, NetBeans y Oracle JDeveloper.<sup>4</sup> (Jet Brains, 2023).

### **2.3.5 Oracle Database**

Una base de datos Oracle es un sistema de gestión de bases de datos relacional que permite almacenar y acceder a grandes cantidades de información de manera eficiente y segura. Oracle Database es uno de los sistemas de bases de datos más populares en el mundo empresarial debido a su alta escalabilidad, confiabilidad y seguridad. Ofrece una amplia gama de herramientas para administrar y trabajar con bases de datos, como Oracle Enterprise Manager y SQL Developer. Oracle Database también



es compatible con múltiples lenguajes de programación, como Java, PL/SQL y Python, lo que lo hace versátil y adaptable a diferentes tipos de aplicaciones y necesidades empresariales.

Oracle Database es compatible con una amplia variedad de plataformas de hardware y software, lo que lo hace una opción muy versátil y adaptable para la mayoría de las empresas. Su arquitectura permite el almacenamiento y la gestión de grandes cantidades de información, permitiendo un acceso rápido y eficiente a la misma. Asimismo, su seguridad es de alta calidad, ofreciendo una protección completa de los datos y una amplia gama de herramientas para la administración de usuarios y permisos.

En resumen, Oracle Database es una herramienta muy poderosa y confiable para la gestión de grandes volúmenes de datos en el mundo empresarial. Su amplia compatibilidad, alto rendimiento y seguridad hacen que sea una de las opciones más populares para la gestión de bases de datos. (Oracle, 2023).

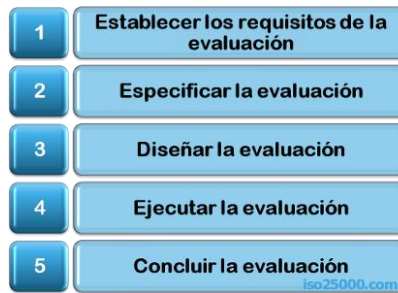
## **2.4 CALIDAD DEL SOFTWARE**

Si bien la calidad de cualquier producto de software depende de todas las personas involucradas durante el proceso de creación del mismo, la mayor responsabilidad sobre la calidad final del producto debería ser garantizada por un profesional del software testing. Dicho profesional buscará apoyar con su metodología, técnicas y herramientas, la calidad del producto mediante actividades de pruebas bien definidas, para proporcionar la confianza adecuada de que un producto o servicio cumplirá las expectativas y necesidades de los usuarios en términos de efectividad, eficiencia, usabilidad y satisfacción.

La calidad del software puede medirse después de elaborado el producto. Pero esto puede resultar muy costoso si se detectan problemas derivados de imperfecciones en el diseño, por lo que es imprescindible tener en cuenta tanto la obtención de la calidad, como su control durante todas las etapas del ciclo de vida del software. (Roa, 2015).

## 2.4.1 ISO 2500

**Figura 2.6**  
*ISO 2500*



Nota Las cinco fases de la norma ISO (ISO 25000, 2023).

La familia ISO/IEC 25000 es el resultado del desarrollo de otras normas anteriores, en particular ISO/IEC 9126, que describe aspectos de los estándares de calidad para productos de software, e ISO/IEC 14598, que se centra en métodos de evaluación de productos informáticos

Según la organización de estandarización internacional: La Norma ISO 9126 establece que cualquier componente de la calidad del software puede ser descrito en términos de una o más de seis características básicas, las cuales son: funcionalidad, confiabilidad, usabilidad, eficiencia, mantenibilidad y portabilidad; cada una de las cuales se detalla a través de un conjunto de sus características que permiten profundizar en la evaluación de la calidad de productos de software. (Roa, 2015).

## 2.5 PRUEBAS DE SOFTWARE

El desarrollo del software requiere de la estimación para controlar y administrar los recursos que se necesita utilizar antes y durante el proyecto.

Son numerosas las variables, relacionadas con los recursos humanos, el contexto y las políticas que intervienen en el proceso de desarrollo que pueden afectar los resultados finales, la estimación no es una tarea que se realiza sólo al inicio de un proyecto de software, dado que en ese momento la información que se dispone es muy precaria, y las funcionalidades requeridas seguramente sufrirán modificaciones por ende el costo también. (ISO 25000, 2023).

**Figura 2.7**

*Evaluación Interna, externa y Calidad de Uso ISO/IEC 2500*



Nota: Componentes del modelo de calidad (ISO 25000, 2023).

La norma ISO/IEC 2500 permite especificar y evaluar la calidad del software desde diferentes criterios asociados con adquisición, requerimientos, desarrollo, uso, evaluación, soporte, mantenimiento, aseguramiento de la calidad y auditoría de software.

### **2.5.1 Métricas**

Una métrica es cualquier medida o conjunto de medidas destinadas a conocer o estimar el tamaño u otra característica de un software o un sistema de información, generalmente para realizar comparativas o para la planificación de proyectos de desarrollo. (Gutiérrez, 2010).

### **2.5.2 Calidad interna y externa**

Especifica 6 características para calidad interna y externa, las cuales, están subdivididas, estas divisiones se manifiestan externamente cuando el software es usado como parte de un sistema Informático, y son el resultado de atributos internos de software.

### **2.5.3 Calidad en uso**

Calidad en uso es el efecto combinado para el usuario final de las 6 características de la calidad interna y externa del software. Especifica 4 características para la calidad en uso.

Al unir la calidad interna y externa con la calidad en uso se define un modelo de evaluación más completo, se puede pensar que la usabilidad del modelo de calidad externa e interna pueda ser igual al modelo de calidad en uso, pero no, la usabilidad es la forma como los profesionales interpretan o asimilan la funcionabilidad del software y la calidad en uso se puede asumir como la forma que lo asimila o maneja el usuario final. Si se unen los dos modelos, se puede definir que los seis indicadores del primer modelo tienen sus atributos y el modelo de calidad en uso.

#### **2.5.4 Prueba de Caja blanca**

El método de pruebas White box (también conocido como clear box testing, glass box testing, transparent box testing, and structural testing) es el cual mira el código y la estructura del producto que se va a probar y usa ese conocimiento para la realización de las pruebas.

Este método se usa en la fase de Unit testing, aunque también puede ocurrir en otras fases como en las pruebas de sistema o de integración para la ejecución de este método es necesario que el tester o la persona que valla a usar el método tengan amplios conocimientos de la tecnología y arquitectura usada para desarrollar el programa.

#### **2.5.5 Prueba de caja negra**

Es el método en el cual el elemento es estudiado desde el punto de vista de las entradas que recibe y las salidas o respuestas que produce, sin tener en cuenta su funcionamiento interno. Estas pruebas son realizadas desde la interfaz gráfica.

#### **2.5.6 Pruebas de estrés**

Las pruebas de estrés tienen como objetivo evaluar el rendimiento del software al someterlo a cargas superiores a sus niveles normales de funcionamiento esto significa que la aplicación se somete a una alta carga de trabajo durante las pruebas de estrés de software, las pruebas de estrés permiten medir la capacidad de respuesta de un sistema en condiciones extremas.

De este modo, el flujo de información, transacciones y usuarios aumenta a un ritmo que supera la tasa usual. Los desarrolladores evalúan su producto mediante pruebas de estrés para analizar cómo funciona el software en situaciones que exceden los límites de trabajo previstos.

## **Ventajas de las pruebas de estrés del software**

Al someter el sistema a cargas elevadas, se puede anticipar el momento en que el programa falla y, al mismo tiempo, comprender cómo se recupera del error, el proceso de pruebas de estrés de software implica la imposición de cargas a los componentes de hardware, como la CPU, la memoria y el espacio en disco, con el fin de evaluar la resistencia del software bajo condiciones extremas de uso de recursos.

El aumento en la demanda de respuesta de la aplicación a un nivel sin precedentes tiene ciertos beneficios, en primer lugar, permite conocer con precisión los picos de uso del producto. Esto significa comprender el comportamiento de la aplicación en situaciones críticas y cómo maneja los errores, todo gracias a las pruebas de estrés de software.

Además, una vez que los conceptos básicos de rendimiento estén claros, la empresa podrá desarrollar una estrategia empresarial más efectiva para el uso y el diseño del sistema, se puede evaluar la velocidad de las respuestas y detectar posibles fallos. Esto es una oportunidad para mejorar el producto antes de su lanzamiento, mediante pruebas de stress testing o participación en eventos importantes, con el fin de satisfacer las necesidades de los usuarios. (Team, 2023).

## **2.6 COSTOS DEL SOFTWARE**

### **2.6.1 COCOMO II**

Estandarizado bajo la ISO 25000, el método COCOMO II puede aplicarse a diversos tipos de software, incluyendo aplicaciones de negocios, sistemas de información gerencial, software en tiempo real, infraestructura, e inclusive software científico y de ingeniería.

Tiene la ventaja que no establece límites arbitrarios al tamaño funcional, por lo cual pueden medirse componentes de software independientemente de si son muy grandes o pequeños. Adicionalmente, el análisis para la medición está basado en el desglose funcional de los componentes de software, por lo que está alineado con las prácticas de Ingeniería de software.

Toda gestión de proyecto de software necesita hacer una debida planificación. Existe una gran necesidad de obtener datos objetivos que permiten evaluar, predecir y mejorar la calidad del software, así como el tiempo y coste de desarrollo del mismo. En este caso se usará el modelo de estimación de COCOMO II (Rincón, 2019).

**Tabla 2.1**  
*Aplicaciones en COCOMO II*

<b>Aplicaciones desarrolladas por usuarios finales</b>		
<b>Generadores de Aplicaciones</b>	<b>Aplicaciones con Componentes</b>	<b>Sistemas Integrados</b>
<b>Infraestructura</b>		

Nota Tipos de aplicaciones para COCOMO II (Gracia, 2022).

### **Modelos de estimación**

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

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

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

### **Puntos objeto**

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

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

- Clasificar cada instancia de un objeto según sus niveles de complejidad (simple, media o difícil)
- Dar el peso a cada objeto según el nivel de complejidad. Los pesos reflejan el esfuerzo relativo requerido para implementar una instancia de ese nivel de complejidad.
- Determinar la cantidad de Puntos Objeto, sumando todos los pesos de las instancias de los tipos de objetos especificados.
- Simulaciones y Pruebas de Esfuerzo: realizar simulaciones o pruebas de esfuerzo basadas en los puntos objeto para validar la estimación y realizar ajustes finales.
- Seguimiento y Aprendizaje: Realizar un seguimiento continuo del progreso del proyecto y aprender de las diferencias entre las estimaciones y los resultados reales para mejorar futuras estimaciones.

**Tabla 2.2**  
*Puntos objeto en COCOMO II*

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

Nota Tabla de puntos objeto método COCOMO II (Gracia, 2022).

## **Puntos Función**

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

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

COCOMO II considera solamente UFP (Puntos Función no ajustados).

$$FP = UFP \times TCF$$

### **Donde**

UFP: Puntos Función no Ajustados

TCF: Factor de Complejidad Técnica

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

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



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

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

El objetivo es medir la cantidad de trabajo intelectual puesto en el desarrollo de el presente sistema desarrollado.

Definir una línea de código es difícil debido a que existen diferencias conceptuales cuando se cuentan sentencias ejecutables y de declaraciones de datos en lenguajes diferentes.

A los efectos de COCOMO II, se eliminan las categorías de software que consumen poco esfuerzo. Así no están incluidas librerías de soporte, sistemas operativos, librerías comerciales, etc., ni tampoco el código generado con generadores de código fuente. (Gracia, 2022).

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

Esta parte se realiza para los modelos Diseño Temprano y Post Arquitectura, para determinar el esfuerzo nominal en el modelo COCOMO II los puntos función no ajustados tienen que ser convertidos a líneas de código fuente considerando el lenguaje de implementación (Gracia, 2022).

## **2.7 SEGURIDAD**

La seguridad del software es una disciplina que se encarga de proteger la integridad, privacidad de los datos del sistema, el software de buena calidad disminuye el riesgo de que personas desconocidas entren al sistema disminuyendo así el costo que provocaría arreglar este tipo de problemas.

“La seguridad del software se relaciona por completo con la calidad. Debe pensarse en seguridad, confiabilidad, disponibilidad y dependencia, en la fase inicial, en la de diseño, en la de arquitectura, pruebas y codificación, durante todo el ciclo de vida del software” (Pajaro, 2020).

### 2.7.1 ISO 27000:2022

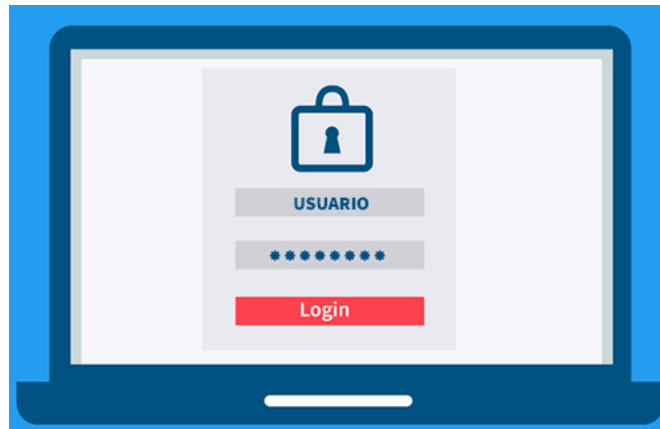
La ISO 27000 es un conjunto de estándares internacionales sobre la seguridad de la Información que nos ayudan en la implementación segura de los sistemas.

La familia de la ISO 27000 contiene un amplio conjunto de buenas prácticas para el establecimiento, implementación, mantenimiento y mejora de sistemas de gestión de la seguridad de la Información entre las cuales destaca el cifrado de la información el respaldo de bases de datos entre otros. (ISO, 2023).

### 2.7.2 Autenticar al usuario

**Figura 2.8**

*Autenticación de un usuario*



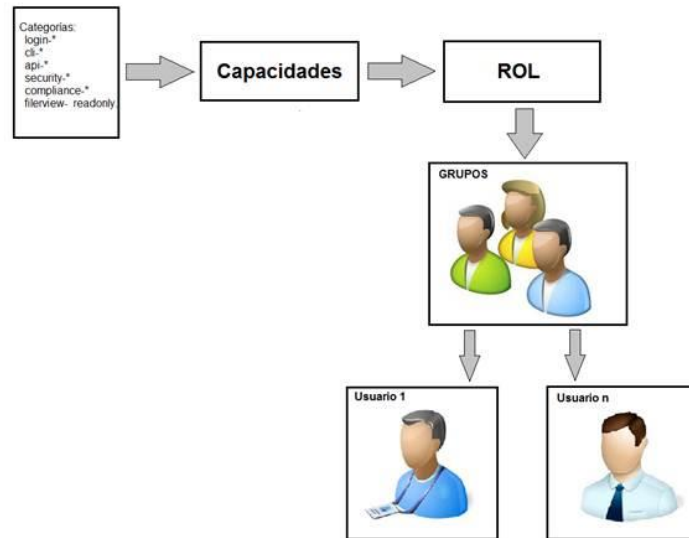
Nota (Tecno Soluciones, 2023).

Hoy día los usuarios de diferentes páginas y sistemas hacen uso de una multitud de servicios en la web, algo que los obliga a tener una credencial para su seguridad, en el presente trabajo se implementó una pantalla de identificación conocida como “Login” que permite a los usuarios del sistema identificarse inequívocamente mediante el uso de un usuario y contraseña, este login se implementó con las medidas de seguridad necesarias.

### 2.7.3 Roles de los usuarios

**Figura 2.9**

*Esquema de roles de un sistema*



Nota Ejemplo de roles para un sistema (Mundo Creativo, 2022)

Se pueden crear varios roles de usuario y luego asignar tareas específicas a los distintos roles, de esta manera se pueden tener diferentes usuarios con distintas tareas en el sistema.

La aplicación de roles de usuario permitirá lo siguiente:

- Ver qué tipos de usuarios existe.
- Dar tareas específicas a usuarios
- Tener listado a los usuarios responsables de algún trabajo

### 2.7.4 Encriptación de contraseñas

Para la encriptación de contraseñas se utilizó el conjunto de funciones hash criptográficas llamada SHA-2

Cuando se utiliza el cifrado SHA, los ID de usuario y las contraseñas de las herramientas de servicio tienen las características siguientes:

- Se utilizan ID de usuario de 10 dígitos y en mayúsculas.

- Se utilizan contraseñas de 128 dígitos y sensibles a las mayúsculas y minúsculas. La contraseña debe seguir las reglas de contraseña establecidas en SST. Esto incluye la longitud mínima y máxima de la contraseña.
- De forma predeterminada, las contraseñas se establecen inicialmente como no caducadas. (IBM, 2022).

### **2.7.5 Backup**

El presente trabajo usa una herramienta de Oracle llamada RMAN (Recovery Manager) con el propósito diseñar

RMAN es una herramienta por línea de comandos que permite realizar copias de seguridad y restauraciones de una base de datos llamada base de datos de destino (target database).

RMAN utiliza un repositorio (repository) para almacenar la información de su configuración, las copias de seguridad realizadas, la estructura de la base de datos de destino, los archivos de traza almacenados, etc. (Heurtel, 2015).

## **2.8 MANTENIMIENTO DEL SOFTWARE**

Aun cuando nuestro software está en las últimas en su ciclo de vida, las actividades de mantenimiento de dicho software no son las menos importantes al contrario el mantenimiento del software es una actividad principal que tiene costes y consume recursos dentro de una empresa.

Es una parte natural del ciclo de vida del desarrollo de software (SDLC). Los desarrolladores no pueden simplemente lanzar un producto, sino que deben estar constantemente atentos a mejorar su software para seguir siendo competitivos.

Según la terminología ANSI-IEEE, el mantenimiento del software es: “la modificación de un producto software después de su entrega al cliente o usuario para corregir defectos, para mejorar el rendimiento u otras propiedades deseables, o para adaptarlo a un cambio de entorno”.

### **2.8.1 Pruebas unitarias**

Las pruebas unitarias consisten en la descomposición de una parte del código para verificar su funcionamiento óptimo. Se trata de pruebas pequeñas que comprueban el funcionamiento de un objeto y su lógica. Normalmente, el unit testing se lleva a cabo durante la etapa de desarrollo de aplicaciones de software o móviles. Por lo general, son realizadas por los desarrolladores, aunque en la práctica, también pueden ser llevadas a cabo por los responsables de QA. Existe una especie de mito en torno a las pruebas unitarias. Unos desarrolladores creen que las reuniones son una pérdida de tiempo y las evitan para ser más eficientes. Nada puede estar más lejos de la realidad. Hacer pruebas unitarias de manera regular resulta en un ahorro de tiempo y dinero a largo plazo.

### **2.8.2 Pruebas de interfaz de usuario (UI)**

Las pruebas de interfaz de usuario (UI), a veces denominadas pruebas de interfaz gráfica de usuario (GUI) en contexto, son una serie de actividades que miden la eficacia y la funcionalidad general de los elementos visuales de una aplicación. Su propósito es verificar y validar varias funciones de la interfaz de usuario y garantizar que no haya resultados inesperados, errores o fallas. Las pruebas se utilizan principalmente para verificar aspectos como la usabilidad, la funcionalidad y el rendimiento de la UI para garantizar que sea adecuada para su propósito. En algunos casos también se comprueba la compatibilidad o coherencia visual con los conceptos de diseño generales del sistema.

Las pruebas de la interfaz de usuario suelen ser más efectivas antes de lanzar una aplicación a producción. Esto garantiza que el usuario final obtenga la mejor experiencia con la menor cantidad de errores y errores posibles. Los usuarios finales no son los mejores probadores de software, por lo que es importante resolver cualquier problema antes de llegar a ellos. Las pruebas de interfaz de usuario son una forma útil de evaluar cómo su aplicación maneja ciertas funciones, como el uso de un teclado y un mouse con menús. Esto ayuda a verificar los elementos visuales de la aplicación para garantizar que se muestren correctamente. Las pruebas de la interfaz de usuario

también son una buena forma de medir el rendimiento y asegurarse de que no haya errores ni problemas con la aplicación. (Zap, 2023).

### **2.8.3 Pruebas de Integración en JavaFX**


Las pruebas de integración o pruebas integradas son un método de prueba de software que consiste en analizar la unión y el comportamiento de los componentes en el sistema, incluyendo sus interacciones con otros componentes, archivos operativos y hardware.

Por lo tanto, es responsabilidad de las pruebas de integración examinar las interfaces entre los subsistemas o grupos de componentes del programa o aplicación en cuestión, con el fin de asegurar su correcto funcionamiento. Las pruebas de integración tienen la capacidad de verificar si los componentes del sistema interactúan adecuadamente, utilizando tanto sus interfaces internas como externas. Esto se cuenta entre sus principales características.

Estas pruebas también garantizan que el sistema y sus componentes funcionen correctamente, cumplan con todas las tareas asignadas y se ajusten a los requisitos establecidos.

Otro aspecto notable de este tipo de pruebas es que se pueden utilizar sistemas reales, que están preparados y dedicados específicamente para este propósito, lo que facilitará enormemente el proceso de aplicación de la prueba.

Las pruebas de integración se llevan a cabo después de las pruebas unitarias, en caso de que estas no sean suficientes y sea necesario probar el software de forma más amplia. (Team, 2023).



# CAPÍTULO III

## MARCO APLICATIVO

### 3. MARCO APLICATIVO

#### 3.1 INTRODUCCIÓN

En este capítulo se llevará a cabo el análisis y diseño del sistema desarrollado, utilizando la metodología ágil Scrum. Dicha metodología se basa en un proceso de desarrollo incremental, en el cual se realizan iteraciones o "Sprint" para construir las funcionalidades prioritarias para el negocio y la cantidad de trabajo abordable en cada iteración. Para la ingeniería de requerimientos, se aplicarán las actividades de conceptualización, análisis, diseño del sistema e implementación de acuerdo con el proceso de desarrollo.

#### 3.2 APLICACIÓN DE LA METODOLOGÍA SCRUM

Para el desarrollo del presente proyecto se implementó la metodología de desarrollo ágil Scrum, la aplicación de esta metodología consta de 3 etapas, cada etapa tiene una duración aproximada de 1 mes.

Las etapas que se desarrollaron fueron:

**Etapas de planificación** Se realizan reuniones con el cliente haciendo un análisis de requerimientos y prioridades definiendo de esta manera las tareas a cumplir.

**Etapas de ejecución** Se realizan pequeñas reuniones para mostrar el avance de las tareas al cliente y así diseñar soluciones a los posibles fallos.

**Etapas de prueba y evaluación** Se presenta al cliente los resultados, teniendo en cuenta la eficacia de los resultados se decidirá si habrá alguna modificación.

#### 3.3 EQUIPO SCRUM

Se han identificado a los siguientes roles para conformar el equipo scrum:

**Scrum Manager.** Se ha definido como Scrum Manager a la Lic. María Magdalena Aguilar Guanto por su conocimiento y su supervisión en el desarrollo del presente proyecto



**Product Owner.** En el papel del propietario del sistema desarrollado se ha definido al Gerente de la empresa Electronic, printer & services S.R.L el señor Jaime Tola Quispe, por el conocimiento del negocio y de lo que se desea como producto final.

**Developer.** Para el desarrollo del sistema el rol caerá en el Univ. Ricardo Franco Sanchez Quispe por ser el autor del presente proyecto de grado además de tener el conocimiento de las herramientas empleadas.

El equipo scrum seria conformado por:

**Tabla 3.1**  
*Equipo scrum*

<b>Actores</b>	<b>Responsables</b>
Scrum Manager	Lic. María Magdalena Aguilar Guanto
Product Owner	Jaime Tola Quispe
Developer	Univ. Ricardo Sanchez Quispe

Nota: Equipo scrum para el desarrollo del proyecto

### **3.4 APLICANDO EL MODELO CRM**

Con base en los requerimientos anteriormente obtenidos, se procedió a la implementación del modelo CRM ajustando el sistema y los módulos a desarrollar para satisfacer las necesidades de la empresa E.P.S S.R.L.

Según el análisis del esquema y la retroalimentación se definió 3 iteraciones las cuales son las siguientes:

- Primera Iteración: Se desarrolló el sistema base para la gestión de clientes
- Segunda iteración: Se adiciono el módulo seguimiento del trabajo de los clientes y reportes
- Tercera Iteración: Se adiciono el Dashboard para mejorar el trabajo de los técnicos y se agregó los reportes en Excel

Se desarrolló cada iteración identificando y definiendo las características requeridas por el cliente para posteriormente implementarlo junto a la base de datos Oracle

### **3.4.1 Descripción y características del sistema**

#### **Interfaz gráfica**

La interfaz desarrollada es intuitiva, optimizada para agilizar todas las tareas que gestiona el sistema, la información introducida estará en una sola base de datos donde cada uno de sus usuarios podrá acceder a ella con diferentes privilegios y limitaciones por cuestiones de seguridad, el apartado grafico está basado en la imagen de la empresa.

#### **Clientes**

Para gestionar los clientes el sistema contara con una opción búsqueda rápida, en los datos del cliente estarán su nombre, dirección, teléfono, correo y contacto.

#### **Equipos**

El sistema tendrá una presentación intuitiva que le permitirá técnico registrar todos los equipos del cliente que lleguen a la empresa para su reparación y o mantenimiento, cada equipo está separado en tipos de equipo el cual nos detalla si es una computadora, una laptop, fotocopiadora, etc. El campo de modelos que nos detalla que modelo es el equipo y por último el campo de marcas el cual se registra la marca del equipo.

Todos estos datos permitirán un correcto registro de los equipos del cliente en la base de datos.

#### **Servicios**

En este módulo se registrarán los servicios los cuales corresponden a cada equipo que llega a la empresa, en este módulo se registra el estado del equipo, el responsable el lugar donde se realizara el trabajo, el tipo de trabajo que se realizara al equipo y el técnico asignado al equipo.

## **Seguimiento**

En este módulo se hace el seguimiento de los equipos correspondiente al usuario que inicio la sesión en el sistema, también se puede reasignar un servicio a otro técnico, así como dar de alta un servicio ya completado

Este módulo cuenta con un historial de todos los cambios realizados a un servicio.

## **Opciones**

En esta parte podremos registrar los tipos de estado de los equipos, el tipo de trabajo realizados por la empresa y por último el lugar de atención.

## **Reportes**

En el módulo de reportes podemos generar un pdf o una hoja de Excel, con los datos de un cliente las fechas que cubre el reporte y seleccionamos el tipo de reporte que se requiere estos pueden ser de tipo preventivo y correctivo general, por equipos o por servicios.

## **Dashboard**

En este módulo se podrán observar los trabajos pendientes que tiene el personal de la empresa, ordenados por fechas desde los trabajos con fecha de finalización ya por caducar hasta los trabajos que tienen más tiempo para ser completados, dichos trabajos al ser dados de alta desaparecerán del dashborad.

## **3.5 PRIMERA ITERACIÓN**

### **3.6.1 Etapa de planificación**

#### **3.6.1.1 Análisis de requerimientos**

Después de reuniones entre el Scrum Master, Owner Product y el Developer se llegó un Backlog del producto este contiene los siguientes requerimientos y características finales del sistema, ordenados por prioridad.

Los cuales se detallan a continuación:

**Tabla 3.2**  
backlog usuarios

Back log usuarios (BU)			
ID	Descripción	Prioridad	Estado
BU1	El usuario debe ser capaz de logearse correctamente	Alta	Concluido
BU2	El usuario debe ser capaz de crear nuevos usuarios	Alta	Concluido
BU3	El usuario debe ser capaz de modificar roles de usuarios	Media	Concluido
BU4	El usuario debe ser capaz de asignar roles a usuarios	Media	Concluido
BU5	El usuario deber modificar usuarios	Alta	Concluido
BU6	El usuario debe ser capaz de modificar su contraseña	Media	Concluido

Nota: Tabla de requerimientos para el módulo de usuarios

**Tabla 3.3**  
Backlog servicios

Back log servicios (BS)			
ID	Descripción	Prioridad	Estado
BS1	El usuario debe ser capaz crear, editar y eliminar Servicios.	Alta	Concluido
BS 2	El usuario debe ser capaz de buscar los servicios realizados	Media	Concluido

BS 3	El usuario debe ser capaz asignar técnicos a los servicios a realizar	Alta	Concluido
BS 4	El usuario debe comprender el funcionamiento de la interfaz de servicios	Alta	Concluido

Nota: Tabla de requerimientos para el módulo de servicios

**Tabla 3.4** *backlog equipos*

Back log equipos (BE)

ID	Descripción	Prioridad	Estado
BE1	El usuario debe ser capaz crear, editar y eliminar equipos.	Alta	Concluido
BE 2	El usuario debe ser capaz de buscar los equipos registrados	Media	Concluido
BE 3	El usuario debe ser capaz realizar seguimiento de los equipos	Alta	Concluido
BE 4	El usuario debe comprender el funcionamiento de la interfaz de registro de equipos	Alta	Concluido

Nota: Tabla de requerimientos para el módulo de equipos

**Tabla 3.5**  
*Backlog técnicos*

Back log técnicos (BT)

ID	Descripción	Prioridad	Estado
BT 1	El usuario debe ser capaz crear, editar y eliminar a los técnicos.	Alta	Concluido

BT 2	El usuario debe ser capaz de buscar los técnicos registrados	Media	Concluido
BT 3	El usuario debe comprender el funcionamiento de la interfaz de técnicos.	Alta	Concluido

Nota: Tabla de requerimientos para el módulo de técnicos

**Tabla 3.6**  
*Backlog clientes*

Back log clientes (BC)			
ID	Descripción	Prioridad	Estado
BC 1	El usuario debe ser capaz de crear, editar y eliminar a los clientes.	Alta	Concluido
BC 2	El usuario debe ser capaz de buscar los clientes registrados	Media	Concluido
BC 3	El usuario debe comprender el funcionamiento de la interfaz de clientes	Alta	Concluido

Nota: Tabla de requerimientos para el módulo de clientes

**Tabla 3.7**  
*Backlog seguimiento*

Back log seguimiento (BS)			
ID	Descripción	Prioridad	Estado
BS 1	El usuario debe ser capaz de ver el seguimiento de los equipos	Alta	Concluido
BS 2	El usuario debe ser capaz de dar de alta los equipos trabajados	Alta	Concluido

BS 3	El usuario debe ser capaz de cambiar ver el historial de seguimiento	Media	Concluido
------	--	-------	-----------

Nota: Tabla de requerimientos para el módulo de seguimiento

**Tabla 3.8**  
*Backlog reportes*

Back log reportes (BR)			
ID	Descripción	Prioridad	Estado
BR 1	El usuario debe ser capaz de generar reportes ejecutivos para los clientes	Alta	Concluido
BR 2	El usuario debe ser capaz de generar reportes completos para los técnicos	Alta	Concluido
BR 3	El usuario debe ser capaz de generar reportes sobre su trabajo personal.	Media	Concluido

Nota: Tabla de requerimientos para el módulo de reportes

### Requerimientos

**Facilidad de uso:** Se implementarán interfaces sencillas, menús, enlaces, etc., que facilitarán el uso y comprensión del sistema.

**Rendimiento:** El sistema presentara un tiempo de respuesta veloz y aceptable a las peticiones realizadas por los usuarios, por solicitud.

**Escalabilidad:** Está asegurada una manera fácil de adicionar nuevas funcionalidades a través de nuevos módulos que surjan de las necesidades presentadas por los usuarios.

**Disponibilidad:** El sistema será compartido por medio de la red local para garantizar la disponibilidad del mismo a todo el personal de la empresa.

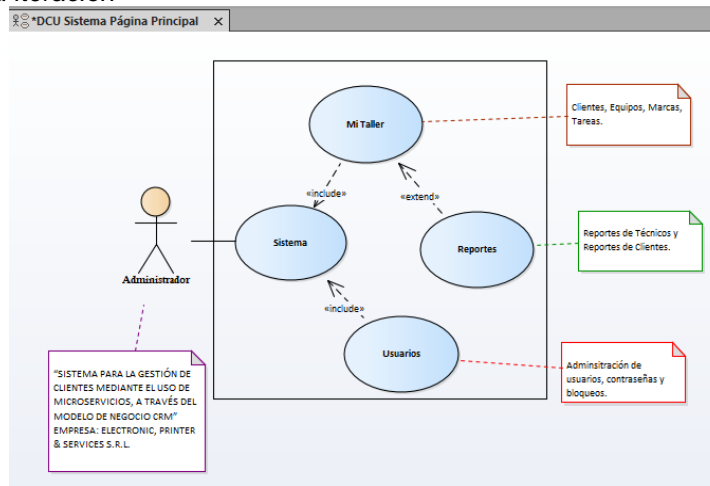
### 3.6.2 Etapa de implementación

#### 3.6.2.1 Casos de uso

En el siguiente caso de uso se detallará las actividades desarrolladas en esta iteración donde podemos observar los módulos principales del sistema.

**Figura 3.1**

Caso de uso primera iteración



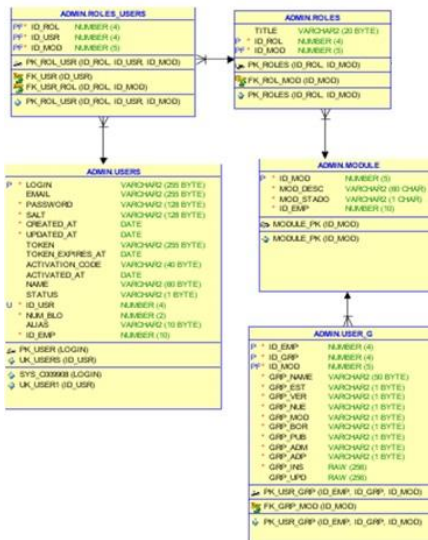
Nota Caso de uso para la primera iteración del sistema

#### 3.6.2.2 Estructura de las tablas

En este diagrama se detalla las tablas correspondientes a esta iteración.

**Figura 3.2**

Diagrama de clases primera iteración





Nota Primer diagrama de clases para el sistema (Elaboración propia)

### 3.6.2.3 Modelo entidad relación

En este paso se muestra el modelo entidad relación para esta iteración.

Figura 3.3

Modelo entidad relación primera iteración



Nota Primer diseño de la base de datos

### 3.6.2.4 Código resultado

Se muestra una imagen del código obtenido para el módulo de usuarios.

**Figura 3.4**

*Código modulo primera iteración*

```
colLogin.setCellValueFactory(new PropertyValueFactory<>( s: "login"));
colNombre.setCellValueFactory(new PropertyValueFactory<>( s: "name"));
colEmail.setCellValueFactory(new PropertyValueFactory<>( s: "email"));
colAlias.setCellValueFactory(new PropertyValueFactory<>( s: "alias"));
colStatus.setCellValueFactory(new PropertyValueFactory<>( s: "dspEstado"));
colID.setCellValueFactory(new PropertyValueFactory<>( s: "idUser"));

ProgInd.setVisible(false);

//Cargar Tabla
listData = FXCollections.observableArrayList();
loadData();
tableData.getSelectionModel().clearSelection();
attachEvents();
bindFieldsToModel();

///// Inicio de Validaciones
txtSearch.textProperty().addListener((ov, oldValue, newValue) -> {
    txtSearch.setText(newValue.toUpperCase());
});
txtLogin.textProperty().addListener((ov, oldValue, newValue) -> {
    txtLogin.setText(newValue.toUpperCase());
});
txtNombre.textProperty().addListener((ov, oldValue, newValue) -> {
```

Nota Fragmento de código del módulo usuarios

## 3.7 SEGUNDA ITERACIÓN

### 3.7.1 Etapa de planificación

#### Módulo de administración de usuarios

En esta etapa se desarrollaron los elementos pertenecientes al backlog usuarios mediante los datos obteniendo anteriormente realizando mejoras y corrigiendo errores encontrados en el anterior sprint así generamos el siguiente sprint:

**Tabla 3.9**  
*Sprint backlog de usuarios*

**Sprint 1- Backlog usuarios (SBU)**

<b>ID</b>	<b>Actividad</b>	<b>Tipo</b>	<b>Días</b>	<b>Estado</b>
SBU1	Realizar la planificación de la iteración	Planificación	1	Terminado
SBU2	Analizar los requerimientos del backlog usuarios	Planificación	2	Terminado
SBU3	Especificación de requerimientos	Desarrollo	2	Terminado
SBU4	Diseño de datos	Desarrollo	2	Terminado
SBU5	Diseño de la arquitectura	Desarrollo	2	Terminado
SBU6	Implementación del modulo	Desarrollo	9	Terminado
SBU7	Prueba y evaluación	Desarrollo	1	Terminado

Nota: Tabla de pruebas modulo usuarios

Con este Sprint L. Backlog de usuarios, procedemos a desarrollar cada una de las fases.

- **Identificación de usuarios**

Los usuarios deberán autenticarse para poder usar el sistema en la siguiente tabla veremos las especificaciones para el grupo de usuarios:

**Tabla 3.10**  
*Grupo de usuario administrador*

<b>Nombre</b>	<b>Administrador</b>
Descripción	Usuario administrador, administración de usuarios y administración de roles.
Perfil de datos	Nombre, contraseña
Historias de usuario	Login Adicionar, Modificar y eliminar usuarios Adicionar, Modificar y eliminar roles

Nota Tabla de pruebas de login

- **Requerimientos funcionales**

En esta parte identificaremos los procesos a tratar por el sistema mediante el uso de historias de usuario, para organizar y clasificar mediante la elaboración de tablas.

**Tabla 3.11**

*Inicio de sesión usuario*

### HISTORIA DE USUARIO

---

**Nro.:** 1

**Nombre:** Inicio de sesión

**Prioridad:** Alta

**Riesgos de desarrollo:** Medio

**Descripción:** Login de inicio

**Observaciones:** Funciona correctamente

*Nota: Prueba historial de usuario inicio de sesión*

**Tabla 3.12**

*Creación de usuarios*

### HISTORIA DE USUARIO

---

**Nro.:** 2

**Nombre:** Creación de usuario

**Prioridad:** Alta

**Riesgos de desarrollo:** Medio

**Descripción:** El usuario creara nuevos usuarios para el sistema

**Observaciones:** Funciona correctamente

*Nota: Elaboración propia*

**Tabla 3.13**

*Prueba historial de usuario creación de usuario*

### HISTORIA DE USUARIO

---

**Nro.:** 3

**Nombre:** Administración de usuarios

**Prioridad:** Media

**Riesgos de desarrollo:** Medio

**Descripción:** Se podrá buscar y administrar los usuarios del sistema

**Observaciones:** Funciona correctamente

*Nota: Prueba historial de usuario administracion*

**Tabla 3.14**

*Asignación de roles de usuario*

### HISTORIA DE USUARIO

---

**Nro.:** 4

**Nombre:** Asignación de roles

**Prioridad:** Alta

**Riesgos de desarrollo:** Medio

**Descripción:** Asignación de roles a los usuarios

**Observaciones:** Funciona correctamente

*Nota: Prueba historial de usuario asignando roles*

**Tabla 3.15**

*Bloqueo de usuarios*

### HISTORIA DE USUARIO

---

**Nro.:** 5

**Nombre:** Bloqueo de usuarios

**Prioridad:** Media

**Riesgos de desarrollo:** Medio

**Descripción:** Se bloqueará a los usuarios

**Observaciones:** Funciona correctamente

*Nota: Prueba historial de usuario bloqueo de usuarios*

**Tabla 3.16**  
*Cambio de contraseñas del usuario*

### HISTORIA DE USUARIO

---

**Nro.:** 6

**Nombre:** Cambio de contraseña del usuario

**Prioridad:** Alta

**Riesgos de desarrollo:** Medio

**Descripción:** El usuario cambiara su contraseña por una nueva

**Observaciones:** Funciona correctamente

*Nota: Prueba historial de usuario inicio de sesion*

**Tabla 3.17**  
*Desbloqueo de usuarios*

### HISTORIA DE USUARIO

---

**Nro.:** 7

**Nombre:** Desbloqueo de usuarios

**Prioridad:** Alta

**Riesgos de desarrollo:** Medio

**Descripción:** Se desbloqueará a los usuarios bloqueados

**Observaciones:** Funciona correctamente

*Nota: Prueba historial de usuario desbloqueo de usuarios*

**Tabla 3.18**  
*Reseteo de contraseñas*

### HISTORIA DE USUARIO

---

**Nro.:** 8

**Nombre:** Reseteo de contraseñas

**Prioridad:** Alta

**Riesgos de desarrollo:** Medio

**Descripción:** Se podrá resetear las contraseñas de los usuarios

**Observaciones:** Funciona correctamente

*Nota: Prueba historial de usuario reseteo de contraseñas*

### 3.6.2 Etapa de ejecución

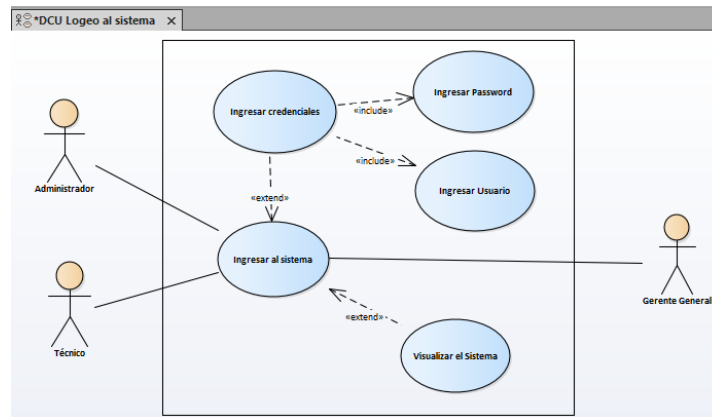
En esta iteración el desarrollo fue enfocado más en el ingreso al sistema, así como el módulo de usuarios.

Según los requerimientos obtenidos se mejoró el módulo de usuarios implementando roles, cifrado de contraseñas, bloqueo de usuarios y restablecimiento de contraseñas. Posteriormente se diseñó y desarrollo la iteración para cumplir con todas las tareas requeridas por el Product Owner, para el diseño de este módulo se usó el modelo UML.

#### 3.6.2.1 Casos de uso

**Figura 3.5**

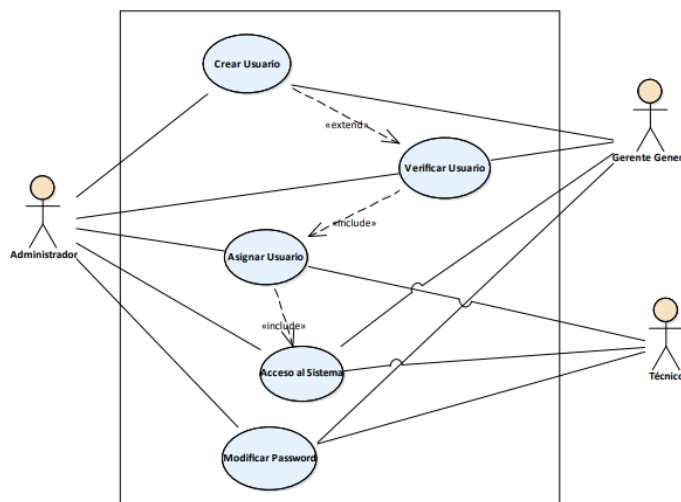
Casos de uso usuarios



Nota Diagrama de caso de uso para los usuarios del sistema

**Figura 3.6**

Casos de uso administración de usuarios



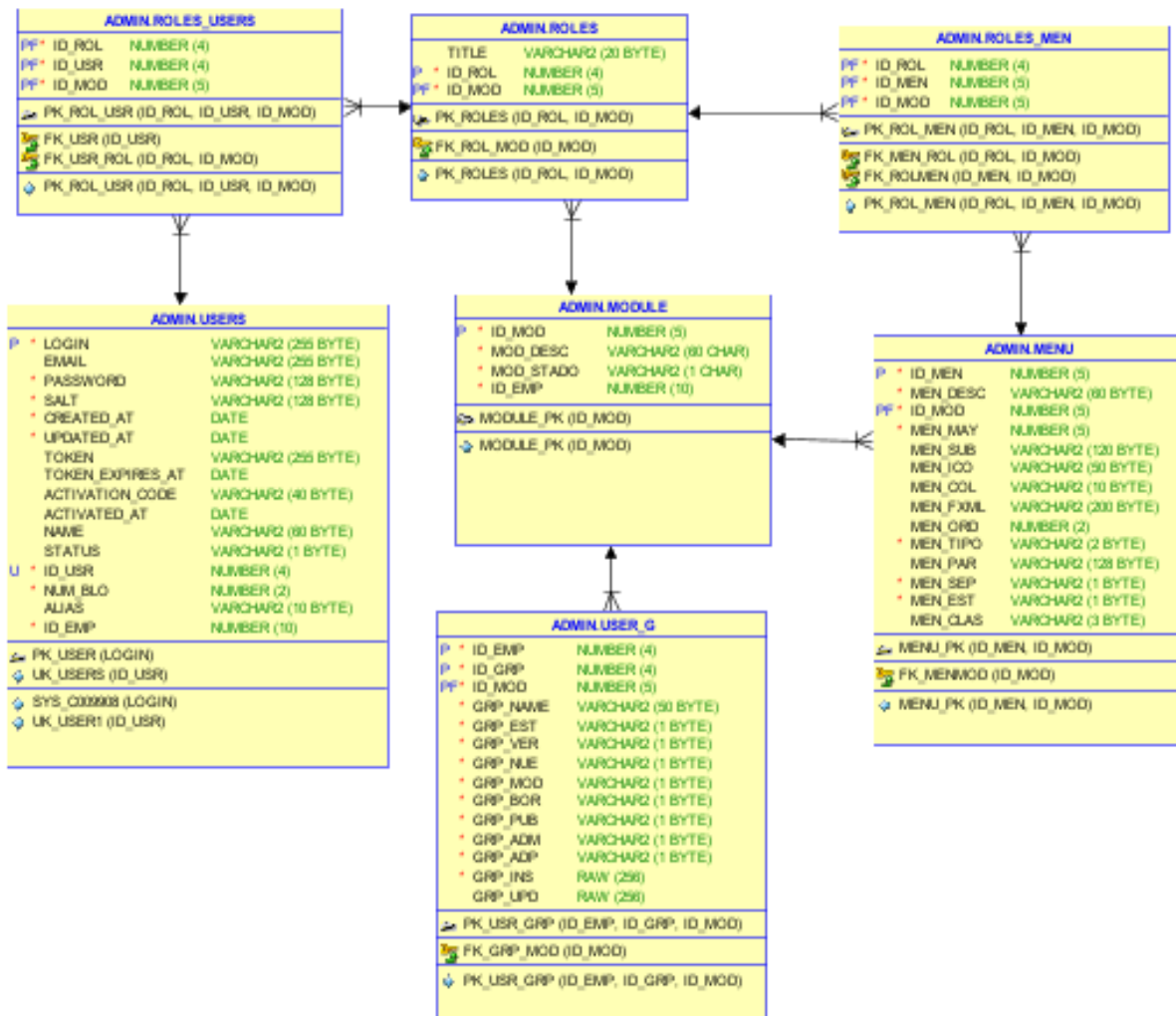
Nota Diagrama de casos de uso actividades de los usuarios

### 3.7.2.2 Estructura de las tablas

En este diagrama se detalla las tablas correspondientes a esta iteración para el correcto manejo de los datos según las modificaciones requeridas.

Se agregó las tablas necesarias para administra los roles de los usuarios, así como los menús del sistema.

**Figura 3.7**  
Diagrama de clases modulo usuarios



Nota Diagrama de clases para la segunda iteración





### 3.7.2.4 Código resultado

Se muestra una imagen del código del módulo taller.

**Figura 3.9**  
Código modulo taller

```
UserController.java
Listas.setToolTipText(butRefresh, cTipo: "UPD");

colLogin.setCellValueFactory(new PropertyValueFactory<>( s: "login"));
colNombre.setCellValueFactory(new PropertyValueFactory<>( s: "name"));
colEmail.setCellValueFactory(new PropertyValueFactory<>( s: "email"));
colAlias.setCellValueFactory(new PropertyValueFactory<>( s: "alias"));
colStatus.setCellValueFactory(new PropertyValueFactory<>( s: "dspEstado"));
colID.setCellValueFactory(new PropertyValueFactory<>( s: "idUser"));

ProgInd.setVisible(false);

//Cargar Tabla
listData = FXCollections.observableArrayList();
loadData();
tableData.getSelectionModel().clearSelection();
attachEvents();
bindFieldsToModel();

///// Inicio de Validaciones
txtSearch.textProperty().addListener((ov, oldValue, newValue) -> {
    txtSearch.setText(newValue.toUpperCase());
});
txtLogin.textProperty().addListener((ov, oldValue, newValue) -> {
    txtLogin.setText(newValue.toUpperCase());
});
txtNombre.textProperty().addListener((ov, oldValue, newValue) -> {
```

Nota Fragmento de código módulo de taller

### 3.7.3 Etapa de prueba y evaluación

Las pruebas realizadas corresponden al comportamiento del sistema con respecto a los requerimientos funcionales.

Como la iteración actual corresponde al módulo de administración de usuarios, este módulo debe ser probado rigurosamente debido a que puede crear problemas en la seguridad del sistema

A continuación, se detalla las pruebas realizadas en esta iteración:

**Tabla 3.19**  
*Prueba de modulo usuarios*

**Prueba**

Nro.:	1
Nombre:	Ingreso al sistema
Descripción:	Validar que el usuario pueda entrar correctamente al sistema con sus datos
Pasos de la prueba:	El usuario entra al login del sistema
	El usuario coloca sus credenciales y presiona ingresar
Resultados esperados:	Que el usuario entre al sistema con los roles correspondientes a sus datos
Resultados obtenidos:	El usuario ingreso correctamente

Se realizó la reunión con el Product Owner demostrando y verificando que los cambios adicionado en esta iteracion esté funcionando correctamente.

**3.8 TERCERA ITERACIÓN**

**3.8.1 Etapa de planificación**

En esta etapa se desarrollaron los elementos pertenecientes a los backlogs técnicos, servicios, clientes y equipos ya que estos forman el Módulo denominado taller así generamos el siguiente sprint:

**Tabla 3.20**  
 Sprint backlog de técnicos

**Sprint 1- Backlog taller (SBT)**

ID	Actividad	Tipo	Días	Estado
SBT1	Realizar la planificación de la iteración	Planificación	1	Terminado

SBT2	Analizar los requerimientos del backlog	Planificación	4	Terminado
SBT3	Especificación de requerimientos	Desarrollo	3	Terminado
SBT4	Diseño de datos	Desarrollo	7	Terminado
SBT5	Diseño de la arquitectura	Desarrollo	7	Terminado
SBT6	Implementación del modulo	Desarrollo	14	Terminado
SBT7	Prueba y evaluación	Desarrollo	7	Terminado

Nota Tabla de pruebas del módulo de taller

**Tabla 3.21**  
*Sprint backlog de clientes*

### Sprint 1- Backlog clientes (SBC)

ID	Actividad	Tipo	Días	Estado
SBC1	Realizar la planificación de la iteración	Planificación	1	Terminado
SBC2	Analizar los requerimientos del backlog clientes	Planificación	2	Terminado
SBC3	Especificación de requerimientos	Desarrollo	2	Terminado
SBC4	Diseño de datos	Desarrollo	3	Terminado
SBC5	Diseño de la arquitectura	Desarrollo	4	Terminado
SBC6	Implementación del modulo	Desarrollo	5	Terminado
SBC7	Prueba y evaluación	Desarrollo	1	Terminado

Nota Tabla de pruebas para el módulo clientes

Con este Sprint L. Backlog, procedemos a desarrollar cada una de las fases.

- **Requerimientos funcionales**

En esta parte identificaremos los procesos a tratar por el sistema mediante el uso de historias de procesos, para organizar y clasificar mediante la elaboración de tablas.

**Tabla 3.22**  
*Crud de servicios*

### HISTORIA DE PROCESOS

---

**Nro.:** 1

**Nombre:** Crud de servicios

**Prioridad:** Alta

**Riesgos de desarrollo:** Medio

**Descripción:** Crud de creación edición y eliminación de servicios

**Observaciones:**

Nota Prueba para el módulo de servicios

**Tabla 3.23**  
*Crud de equipos*

### HISTORIA DE PROCESOS

---

**Nro.:** 2

**Nombre:** Crud de equipos

**Prioridad:** Alta

**Riesgos de desarrollo:** Medio

**Descripción:** Crud de creación edición y eliminación de equipos

**Observaciones:**

Nota Prueba para el módulo de equipos

**Tabla 3.24**  
*Crud de clientes*

### HISTORIA DE PROCESOS

---

**Nro.:** 3

**Nombre:** Crud de clientes

**Prioridad:** Alta

**Riesgos de desarrollo:** Alto

**Descripción:** Crud de creación edición y eliminación de clientes

**Observaciones:**

Nota Prueba para el módulo de clientes

**Tabla 3.25**  
*Crud de seguimiento*

## HISTORIA DE PROCESOS

**Nro.:** 4

**Nombre:** Pantalla de seguimiento de servicios

**Prioridad:** Alta

**Riesgos de desarrollo:** Alto

**Descripción:** Interfaz gráfica para el seguimiento de los servicios en espera

**Observaciones:**

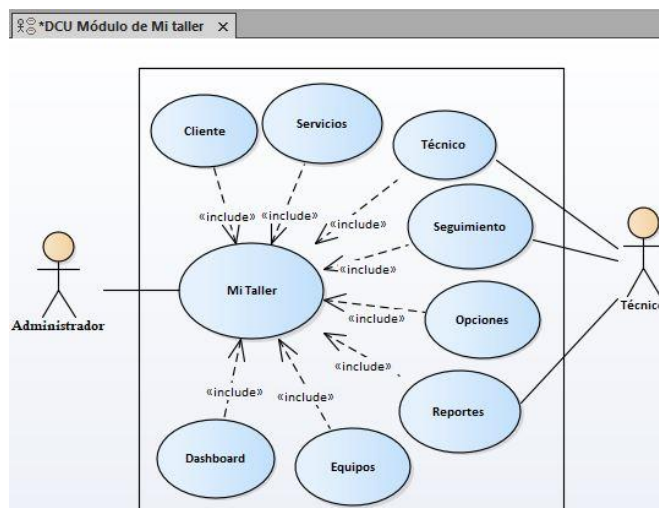
Nota Prueba para el módulo de servicios

### 3.8.2 Etapa de ejecución

Según los requerimientos obtenidos se diseñó y desarrollo la iteración para cumplir con todas las tareas requeridas por el Product Owner, para el diseño de este módulo se usó el modelo UML.

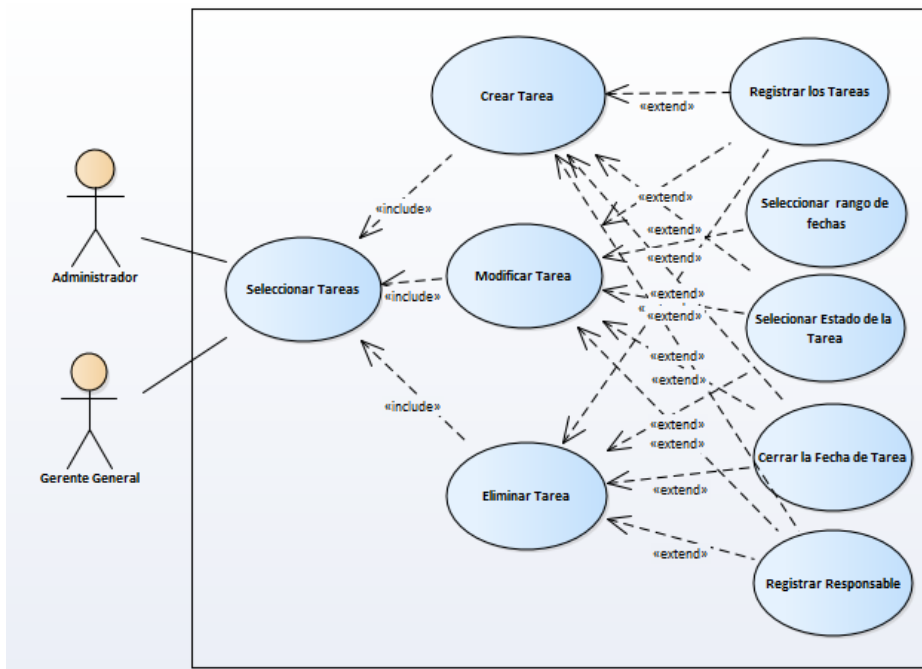
#### 3.8.2.1 Casos de uso

**Figura 3.10** Casos de uso entrada de usuarios al módulo taller



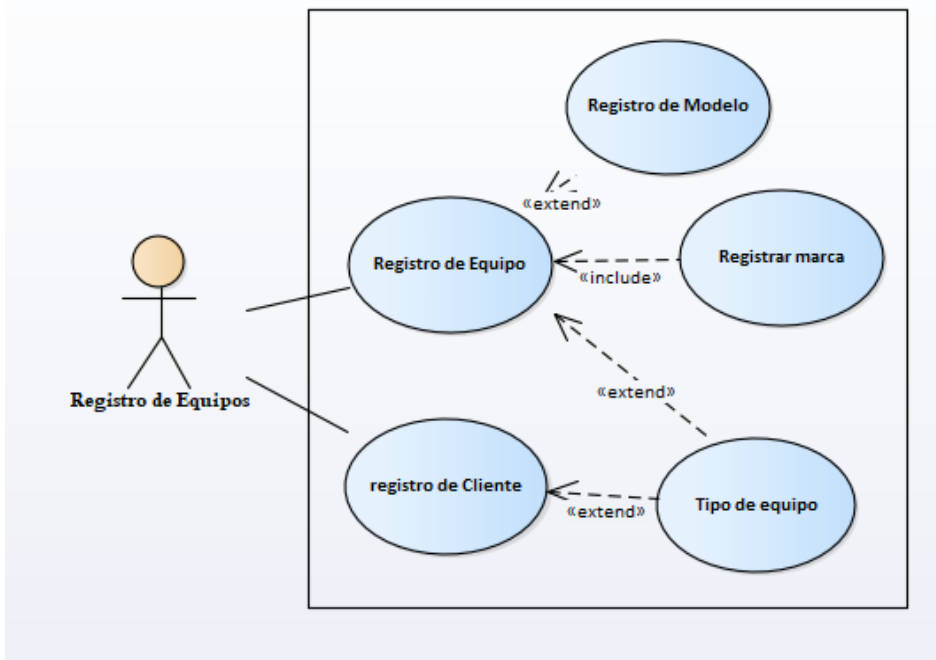
Nota Casos de uso para la tercera iteración usuarios del sistema

**Figura 3.11**  
Casos de uso tareas de los usuarios



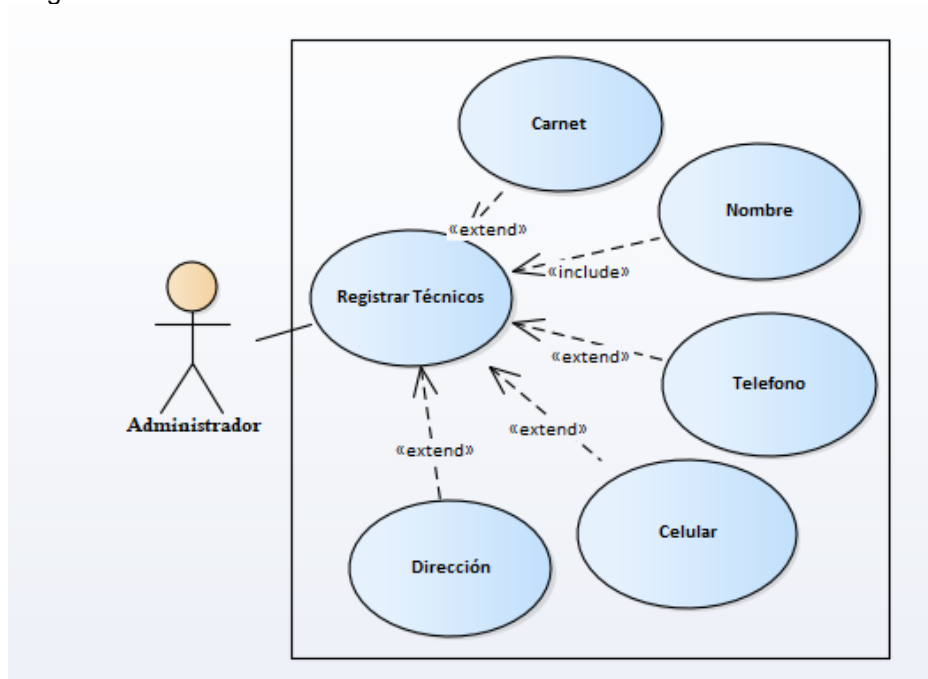
Nota Casos de uso para las tareas de los usuarios

**Figura 3.12**  
Caso de uso registro de equipos



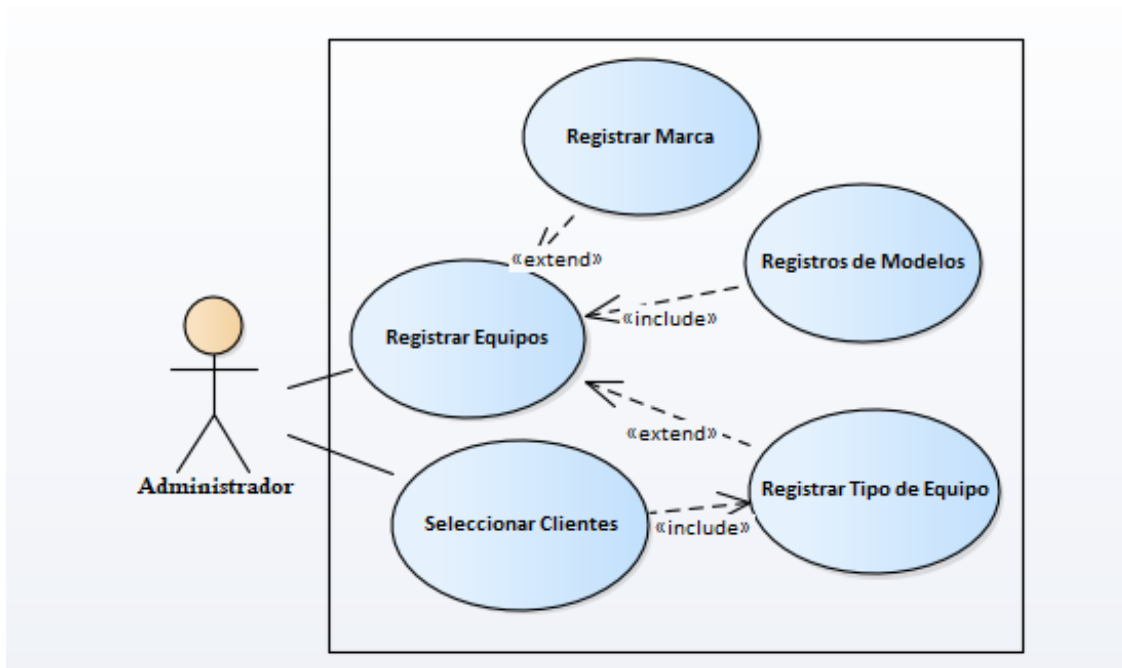
Nota Casos de uso para el registro de equipos

**Figura 3.13**  
Caso de uso registro de técnicos



Nota Casos de uso para el registro de técnicos

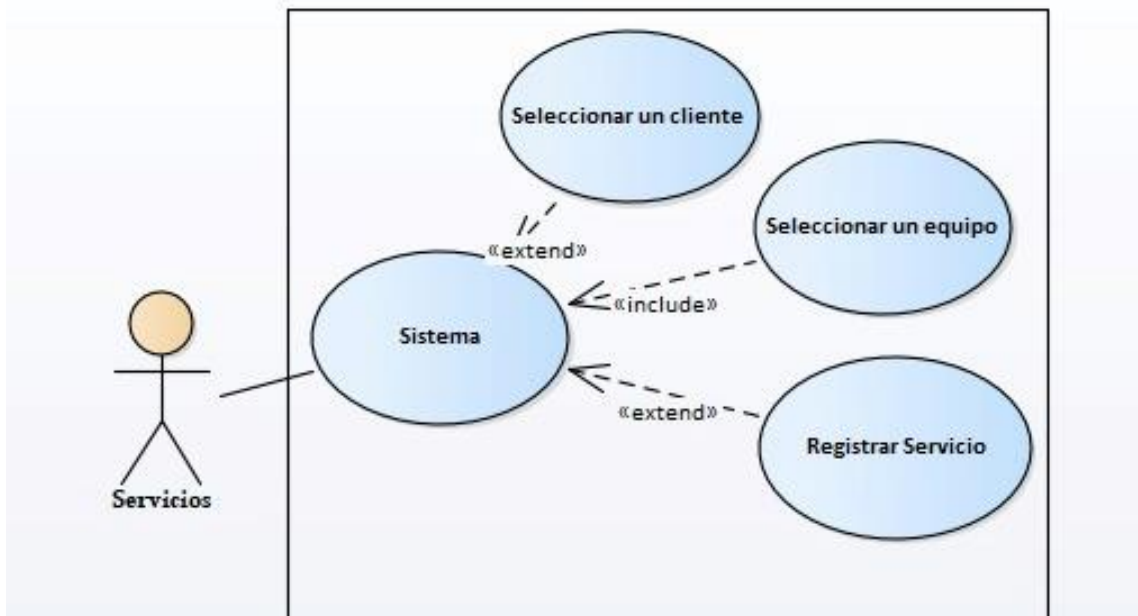
**Figura 3.14**  
Caso de uso registro de clientes



Nota Casos de uso para el registro de clientes

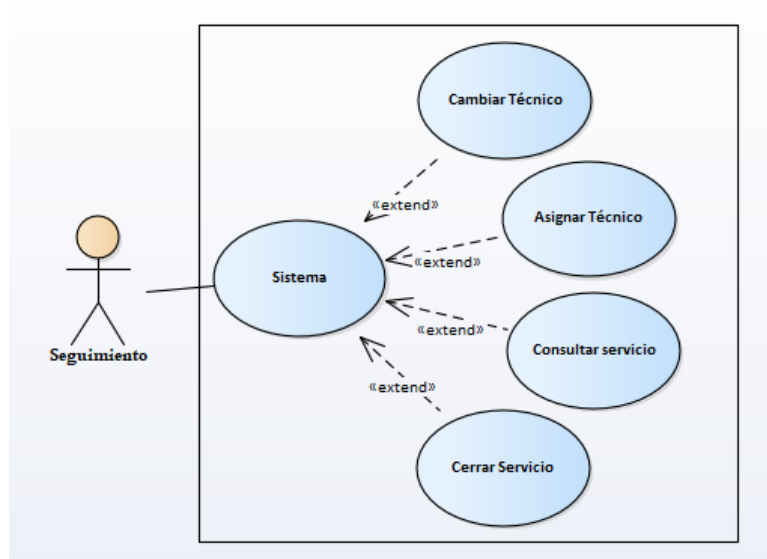


**Figura 3.15**  
Caso de uso servicios



Nota Casos de uso para el módulo de servicios

**Figura 3.16**  
Caso de uso seguimiento

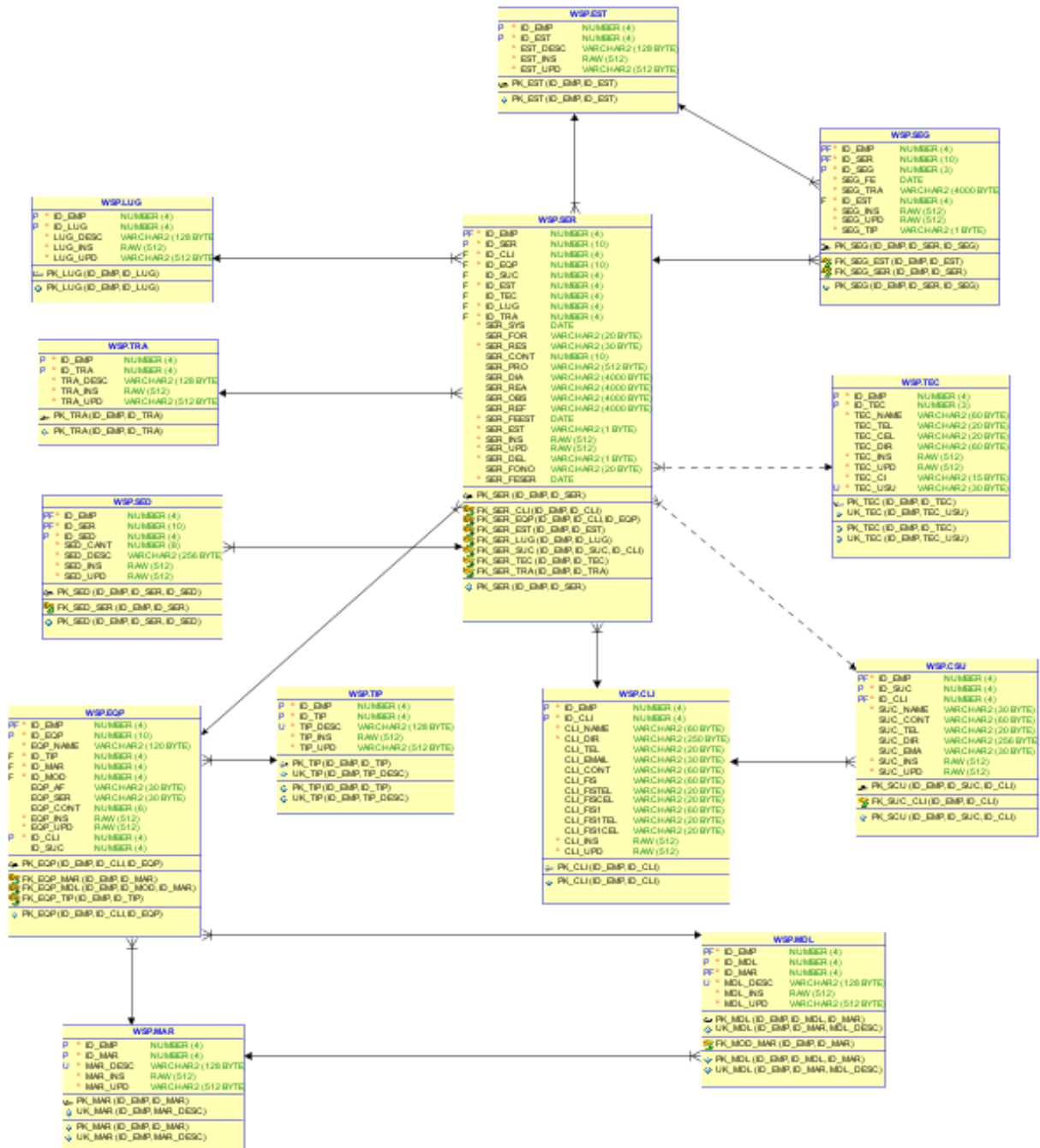


Nota Casos de uso para el módulo de seguimiento del sistema

### 3.8.2.2 Diagrama de clases

En este diagrama se detalla las tablas correspondientes a esta iteración.

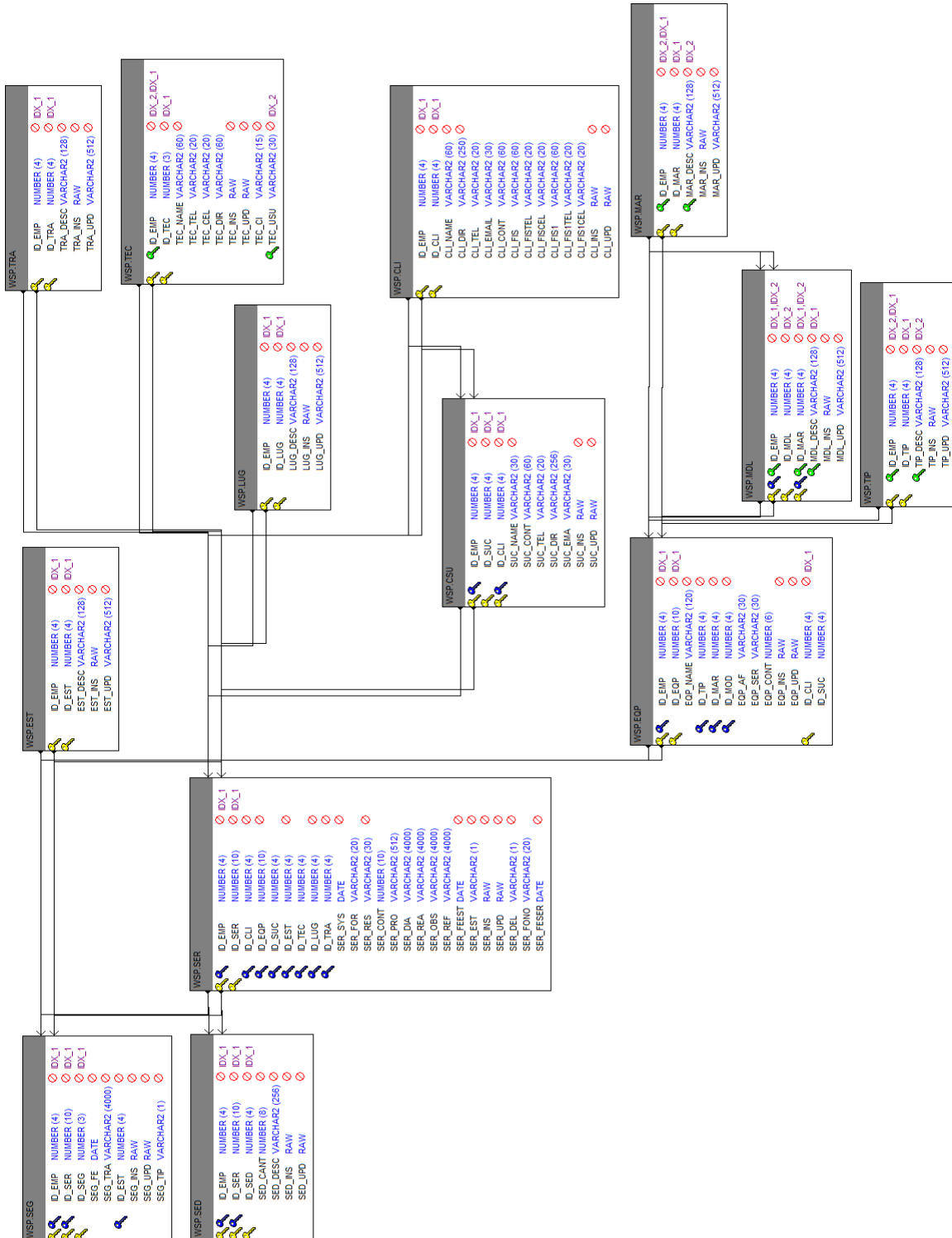
**Figura 3.17**  
Diagrama de clases tercera iteración



Nota Diseño de diagrama de clases para la tercera iteración

### 3.8.2.3 Modelo entidad relación

Figura 3.18  
Modelo entidad relación módulo de taller



Nota Diseño de base de datos para la tercera iteración

### 3.8.2.4 Código resultado

Se muestra una imagen del código del módulo Taller.

**Figura 3.19**

*Código modulo taller*

```
//constructores
| usage
public Cli(Integer id, String name, String dir, String fono, String email, String cont, String fis,
            String tel,String cel, String fisc, String telf, String celu) {
    this.id.set(id);
    this.name.set(name);
    this.dir.set(dir);
    this.fono.set(fono);
    this.email.set(email);
    this.cont.set(cont);
    this.fis.set(fis);
    this.tel.set(tel);
    this.cel.set(cel);
    this.fisc.set(fisc);
    this.telf.set(telf);
    this.celu.set(celu);
}

public Cli() { this( id: 0, name: null, dir: null, fono: null, email: null, cont: null, fis: null, tel: null, cel: nu
```

Nota *Fragmento de código de los cambios realizados*

### 3.8.3 Etapa de pruebas y evaluaciones

Las pruebas realizadas corresponden al comportamiento del sistema con respecto a los requerimientos funcionales.

A continuación, se detalla las pruebas realizadas en esta iteración:

**Tabla 3.26**

*Prueba registro de datos modulo taller*

<b>Prueba</b>	
Nro.:	1
Nombre:	Registro de datos
Descripción:	Validar que el usuario pueda entrar registrar correctamente la información
Pasos de la prueba:	El usuario entra al módulo del sistema

Resultados esperados:	El usuario introduce los datos en los sub módulos técnicos, equipos y servicios presiona guardar
Resultados obtenidos:	Que los datos se registren correctamente en la base de datos
Nota <i>Tabla de pruebas del módulo de taller</i>	

**Tabla 3.27**  
*Prueba edición de datos modulo taller*

<b>Prueba</b>	
Nro.:	2
Nombre:	Edición de datos
Descripción:	Validar que el usuario pueda editar registrar correctamente la información
Pasos de la prueba:	El usuario entra al módulo del sistema
Resultados esperados:	El usuario introduce los datos en los sub módulos técnicos, equipos y servicios presiona guardar
Resultados obtenidos:	Que los datos se registren correctamente en la base de datos
Nota <i>Tabla de pruebas del módulo de taller edición de datos</i>	

**Tabla 3.28**  
*Prueba edición de datos modulo taller*

<b>Prueba</b>	
Nro.:	3
Nombre:	Eliminación de datos
Descripción:	Validar que el usuario pueda eliminar correctamente la información errónea

Pasos de la prueba:	El usuario entra al módulo del sistema
	El usuario introduce los datos en los sub módulos técnicos, equipos y servicios presiona guardar
Resultados esperados:	Que los datos se eliminaron correctamente en la base de datos
Resultados obtenidos:	Los datos se eliminaron correctamente

Nota *Tabla de pruebas del módulo de taller eliminar datos*

Se realizó la reunión con el Product Owner demostrando y verificando que el módulo usuarios esté funcionando correctamente.

### 3.9 CUARTA ITERACIÓN

#### 3.9.1 Etapa de planificación

En esta etapa se desarrollarlo un dashboard para el personal técnico de la empresa, así como los elementos pertenecientes al backlog reportes generando el siguiente sprint:

**Tabla 3.29**  
*Sprint backlog de reportes*

Sprint 1- Backlog reportes (SBR)				
ID	Actividad	Tipo	Días	Estado
SBR1	Realizar la planificación de la iteración	Planificación	1	Terminado
SBR2	Analizar los requerimientos del backlog reportes	Planificación	2	Terminado
SBR3	Especificación de requerimientos	Desarrollo	2	Terminado
SBR4	Diseño de datos	Desarrollo	7	Terminado
SBR5	Diseño de la arquitectura	Desarrollo	7	Terminado

SBR6	Implementación del modulo	Desarrollo	9	Terminado
SBR7	Prueba y evaluación	Desarrollo	2	Terminado

Nota: *Tabla de pruebas del módulo de reportes*

Con este Sprint L. Backlog de usuarios, procedemos a desarrollar cada una de las fases.

- **Requerimientos funcionales**

En esta parte identificaremos los procesos a tratar por el sistema mediante el uso de historias de usuario, para organizar y clasificar mediante la elaboración de tablas.

**Tabla 3.30**

*Selección de cliente para reporte*

### HISTORIA DE USUARIO

---

**Nro.:** 1

**Nombre:** Selección de cliente

**Prioridad:** Alta

**Riesgos de desarrollo:** Medio

**Descripción:** Selecciona al cliente

**Observaciones:** Funciona correctamente

Nota: *Tabla de pruebas selección de cliente*

**Tabla 3.31**

*Selección de fecha para reporte*

### HISTORIA DE USUARIO

---

**Nro.:** 2

**Nombre:** Selección de fecha para reporte

**Prioridad:** Alta

**Riesgos de desarrollo:** Medio

**Descripción:** Se selecciona la fecha para generar el reporte

**Observaciones:** Funciona correctamente

Nota: *Tabla de pruebas fechas de reportes*

**Tabla 3.32**  
Informe de servicios técnico preventivo y correctivo

### HISTORIA DE USUARIO

---

**Nro.:** 3

**Nombre:** Informe de servicios técnico preventivo y correctivo

**Prioridad:** Alta

**Riesgos de desarrollo:** Medio

**Descripción:** Se selecciona el tipo de reporte “Informe de servicios técnico preventivo y correctivo”

**Observaciones:** Funciona correctamente

*Nota: Tabla de pruebas informes preventivos y correctivos*

**Tabla 3.33**  
Informe de servicios técnico preventivo y correctivo equipo

### HISTORIA DE USUARIO

---

**Nro.:** 4

**Nombre:** Informe de servicios técnico preventivo y correctivo equipo

**Prioridad:** Alta

**Riesgos de desarrollo:** Medio

**Descripción:** Se selecciona el tipo de reporte “Informe de servicios técnico preventivo y correctivo equipo”

**Observaciones:** Funciona correctamente

*Nota: Tabla de pruebas informes*

**Tabla 3.34**  
Informe técnico

### HISTORIA DE USUARIO

---

**Nro.:** 5

**Nombre:** Informe técnico

**Prioridad:** Alta

**Riesgos de desarrollo:** Medio

**Descripción:** Se selecciona el tipo de reporte “Informe técnico”

**Observaciones:** Funciona correctamente

*Nota: Tabla de pruebas informes técnicos*



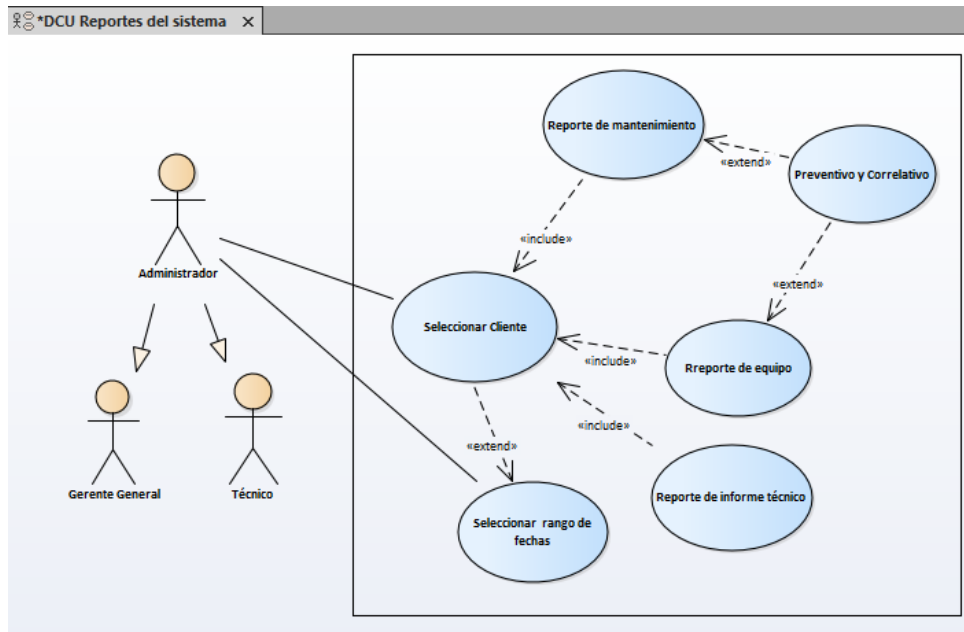
### 3.9.2 Etapa de ejecución

Según los requerimientos obtenidos se diseñó y desarrollo la iteración para cumplir con todas las tareas requeridas por el Product Owner, para el diseño de este módulo se usó el modelo UML.

#### 3.9.2.1 Casos de uso

**Figura 3.20**

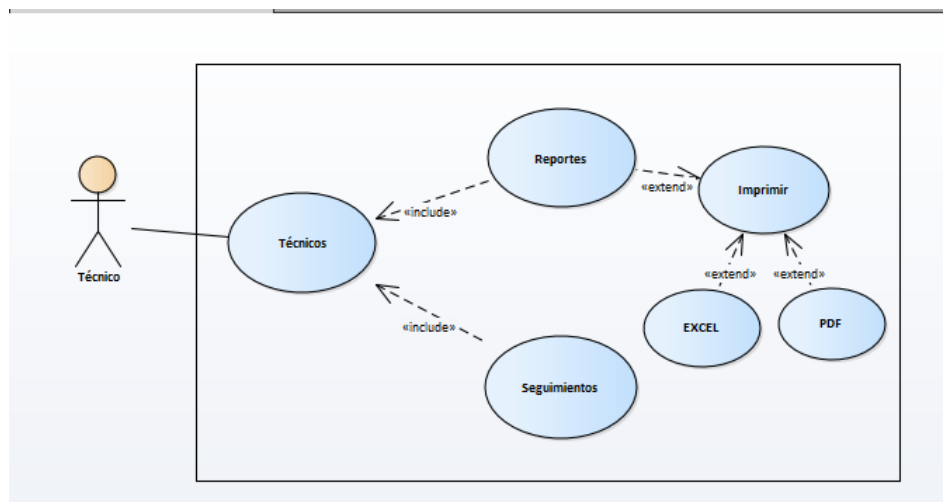
Caso de uso cuarta iteración



Nota Casos de uso cuarta iteración para los usuarios

**Figura 3.21**

Caso de uso realización de reportes



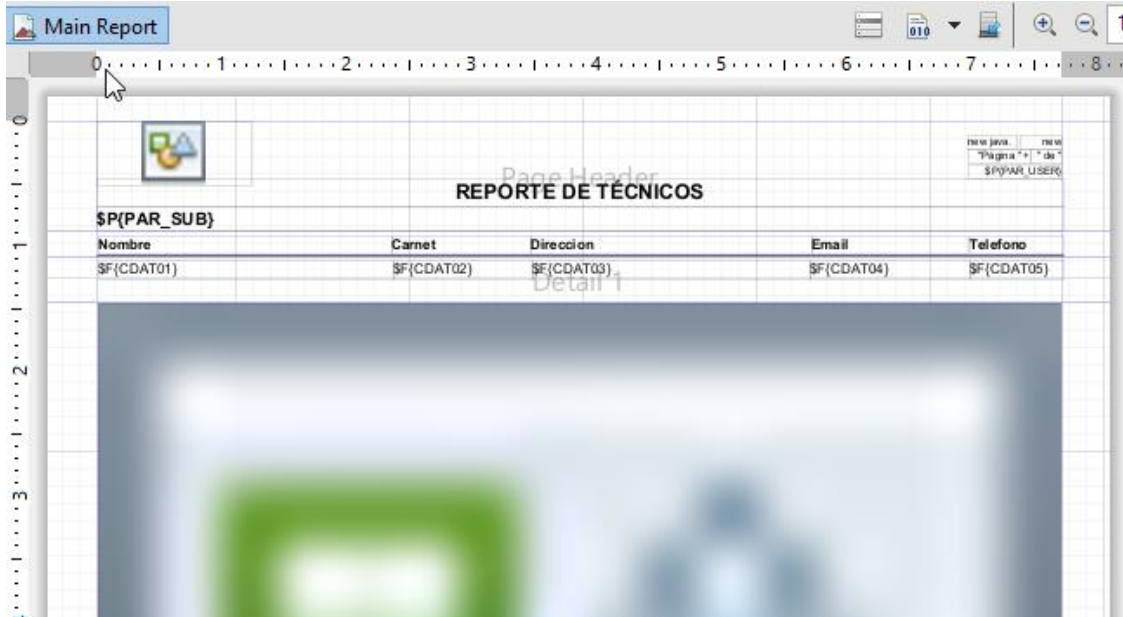
Nota Casos de uso tareas del tecnico

### 3.9.2.2 Diagrama

Construcción de la estructura de los reportes.

Figura 3.22

Construcción de la estructura de reporte



Nota Estructura del reporte de tecnicos

### 3.9.2.3 Código resultado

Se muestra una imagen del código del módulo de reportes

Figura 3.23

Código resultado de módulo de reportes

```
16 </font>
17 </TextField>
18 <Label text="Carnet Identidad" GridPane.rowIndex="1" />
19 <TextField fx:id="txtC1" GridPane.columnIndex="1" GridPane.rowIndex="1" />
20 <Label text="Nombre" GridPane.rowIndex="2" />
21 <TextField fx:id="txtName" GridPane.columnIndex="1" GridPane.columnSpan="4" GridPane.rowIndex="2" />
22 <Label text="Teléfono" GridPane.rowIndex="3" />
23 <TextField fx:id="txtTel" GridPane.columnIndex="1" GridPane.rowIndex="3" />
24 <Label text="Celular" GridPane.columnIndex="3" GridPane.rowIndex="3" />
25 <TextField fx:id="txtCel" GridPane.columnIndex="4" GridPane.rowIndex="3" />
26 <Label text="Dirección" GridPane.rowIndex="4" />
27 <TextArea fx:id="txtDir" prefHeight="200.0" prefWidth="200.0" GridPane.columnIndex="1" GridPane.colum
28 <Label text="Usuario Sistema" GridPane.rowIndex="6" />
29 <TextField fx:id="txtUsu" GridPane.columnIndex="1" GridPane.rowIndex="6" />
30 </children>
31 <padding>
32 <Insets bottom="10.0" left="20.0" right="10.0" top="20.0" />
33 </padding>
34 </GridPane>
35 <HBox alignment="CENTER">
36 <children>
37 <ProgressIndicator fx:id="ProgInd" maxHeight="-Infinity" maxWidth="-Infinity" minHeight="-Infinity" mi
38 <HBox.margin>
```

Nota Fragmento de código creación de reportes

### 3.9.2.4 Reportes generados

#### Generación de reporte en pdf

**Figura 3.24**

*Generación de reporte en pdf*

Nº	Fecha:	Ubicacion	Marca/Modelo	Act/Serie	Problema	Nº Form	Trabajo realizado	Observaciones
<b>COMPUTADORA</b>								
1	03/06/23	villa dolores	HP / COMPUTADORA I7	001 / 1	EQUIPO EN MAL ESTADO	2023/002	SE CAMBIO LA PASTA	SE CAMBIO LA PASTA
TOTAL COMPUTADORA: 1								
<b>IMPRESORA</b>								
1	29/05/23		EPSON / LASER JET 1505	001 / 23	NO ENCIENDE	2023/53		
2	30/05/23	Villa esperanza	EPSON / LASER JET 1505	001 / 23	NO IMPRIME	2023/54	SE LE CAMBIO TONER	EL TONER ESTABA DAÑADO
TOTAL IMPRESORA: 2								
<b>LAPTOP</b>								
1	06/06/23	Viacha	DELL / LAPTOP I3	1 /	LAPTOP NO ENCIENDE	2023/56		
TOTAL LAPTOP: 1								

Nota: Informe final de mantenimiento preventivo y correctivo

#### Generación de reporte en Excel

**Figura 3.25**

*Generación de reporte en Excel*

Nº	Fecha:	Ubicacion	Marca/Modelo	Act/Serie	Problema	Nº Form	Trabajo realizado	Observaciones
<b>COMPUTADORA</b>								
1	03/06/23	villa dolores	HP / COMPUTADORA I7	001 / 1	EQUIPO EN MAL ESTADO	2023/002	SE CAMBIO LA PASTA	SE CAMBIO LA PASTA
TOTAL COMPUTADORA: 1								
<b>IMPRESORA</b>								
1	29/05/23		EPSON / LASER JET 1505	001 / 23	NO ENCIENDE	2023/53		
2	30/05/23	Villa esperanza	EPSON / LASER JET 1505	001 / 23	NO IMPRIME	2023/54	SE LE CAMBIO TONER	EL TONER ESTABA
TOTAL IMPRESORA: 2								
<b>LAPTOP</b>								
1	06/06/23	Viacha	DELL / LAPTOP I3	1 /	LAPTOP NO ENCIENDE	2023/56		
TOTAL LAPTOP: 1								

Nota: Informe final servicios de mantenimiento

### 3.9.3 Etapa de prueba y evaluación

Las pruebas realizadas corresponden al comportamiento del sistema con respecto a los requerimientos funcionales.

Como la iteración actual corresponde al módulo de administración de usuarios, este módulo debe ser probado rigurosamente debido a que puede crear problemas en la seguridad del sistema

A continuación, se detalla las pruebas realizadas en esta iteración:

**Tabla 3.35**  
*Prueba de reportes*

<b>Prueba</b>	
Nombre:	Ingreso al modulo
Descripción:	Validar que el usuario pueda entrar correctamente al módulo y generar sus reportes.
Pasos de la prueba:	El usuario entra al módulo de reportes, genera los reportes que necesita en PDF o Excel
Resultados esperados:	Los reportes se generan correctamente con los correspondientes datos solicitados.
Resultados obtenidos:	El reporte se genera correctamente

*Nota Prueba obtenida para los reportes*

Se realizó la reunión con el Product Owner demostrando y verificando que el módulo de reportes esté funcionando correctamente sin ningún inconveniente.

De esta manera quedaron diseñados y estructurados los módulos para posteriormente implementar y realizar las pruebas correspondientes al sistema

### **3.10 IMPLEMENTACIÓN**

En esta sección se mostrará las pantallas que se generaron como resultado del desarrollo y la implementación del presente sistema en la empresa Electronic, Printer & Services:

Primeramente tendremos los módulos de login y de usuarios para luego mostrar los módulos donde trabajara el personal y finalizaremos con los reportes generados para la empresa así como el modulo de Dashboard.

## Página de login

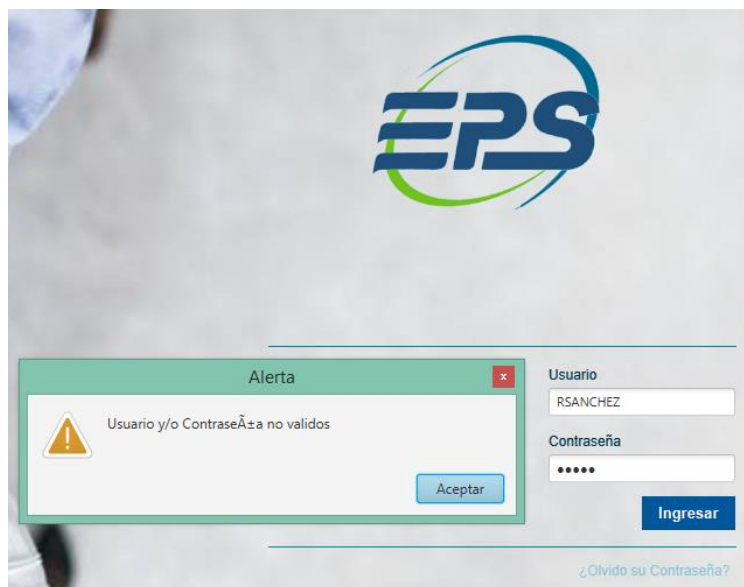
**Figura 3.26**  
*Página de login*



Nota *Pantalla final de inicio de sesión*

## Página de acceso denegado

**Figura 3.27**  
*Página de acceso denegado*



Nota *Alerta de datos incorrectos*

## Página nuevo usuario

**Figura 3.28**

*Página de nuevos usuarios*

The screenshot shows the 'Administración de Usuarios' interface. At the top, there is a search bar with a 'Buscar' button. Below it is a table with the following data:

Usuario	Alias	Nombre	Email	Estado
FSANCHEZ		FRANCO SANCHEZ	buyercatbolivia@gmail.com	Activo
JSANCHEZ		RICARDO SANCHEZ QUISPE	ids.sanchez@gmail.com	Activo

Below the table is a registration form with the following fields:

- Usuario: Código
- Alias: Alias
- Nombre: Nombre del usuario
- Email: Correo Electrónico Institucional

At the bottom right, there are three buttons: 'Nuevo', 'Guardar', and 'Borrar'.

Nota Formulario de registro de nuevos usuarios

## Página administrar roles

**Figura 3.29**

*Administración de roles de usuario*

The screenshot shows the 'Administración de Roles' interface. At the top, there is a navigation bar with 'Usuarios' and icons for user, home, and power. The main content area is titled 'Administración de Roles' and includes the following elements:

- Modulo: ADMINISTRACIÓN DE USUARIOS
- Rol: SUPER USUARIO
- Guardar button
- Two search bars: 'Opciones de Menú' and 'Opciones de Menú Asignadas al Rol'.
- Table 'Opciones de Menú' with columns 'Opción de Menú' and 'Asignar al Rol'.
- Table 'Opciones de Menú Asignadas al Rol' with columns 'Opción de Menú' and 'Quitar del Rol'.

The 'Opciones de Menú' table contains the following data:

Opción de Menú	Asignar al Rol
Reportes	<input type="checkbox"/>

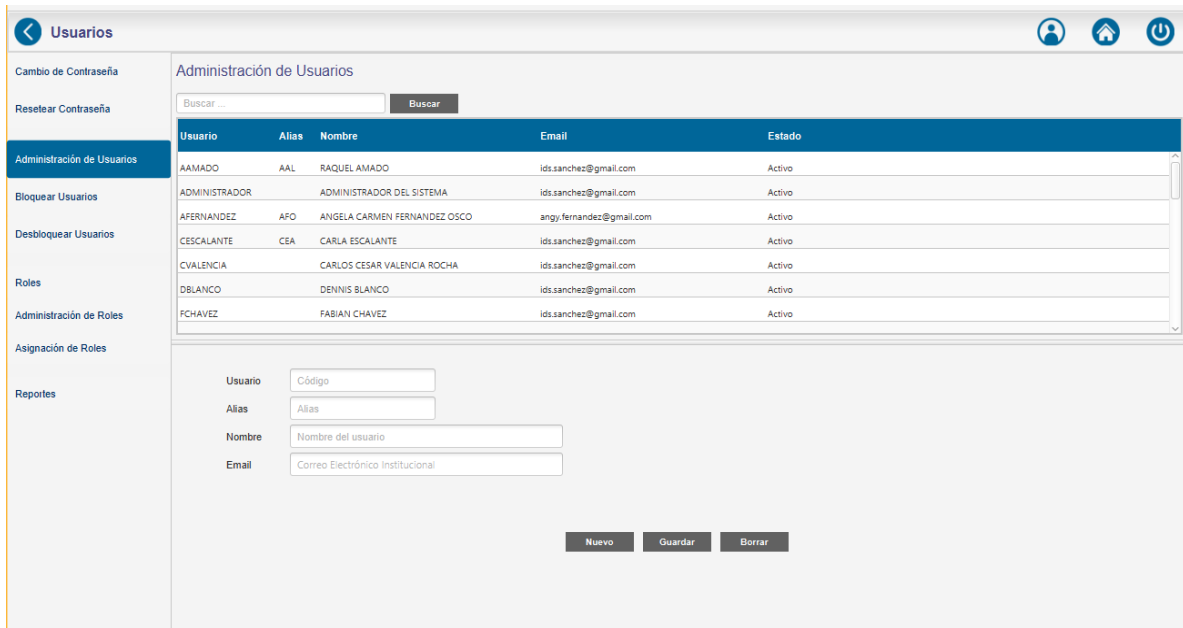
The 'Opciones de Menú Asignadas al Rol' table contains the following data:

Opción de Menú	Quitar del Rol
Menú de Usuarios	<input type="checkbox"/>
Registro de Usuarios	<input type="checkbox"/>
Cambio de Contraseña	<input type="checkbox"/>
Bloquear Usuarios	<input type="checkbox"/>
Asignación de Roles	<input type="checkbox"/>
Administración de Roles	<input type="checkbox"/>
Roles	<input type="checkbox"/>
Resetear Contraseña	<input type="checkbox"/>
Desbloquear Usuarios	<input type="checkbox"/>

Nota: Pantalla de roles de usuario

## Página administración de usuarios

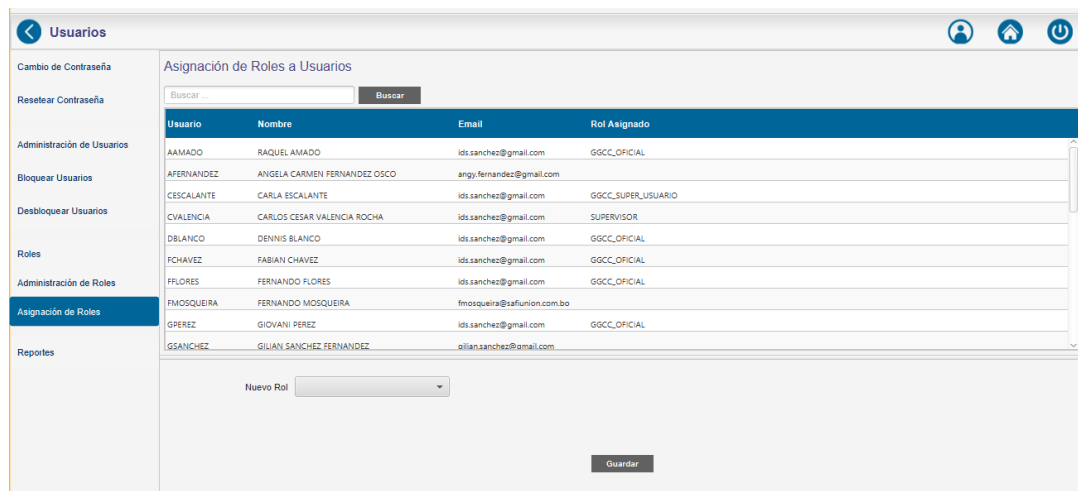
**Figura 3.30**  
*Administración de usuarios*



Nota: Pantalla de administración de usuarios

## Página asignar roles

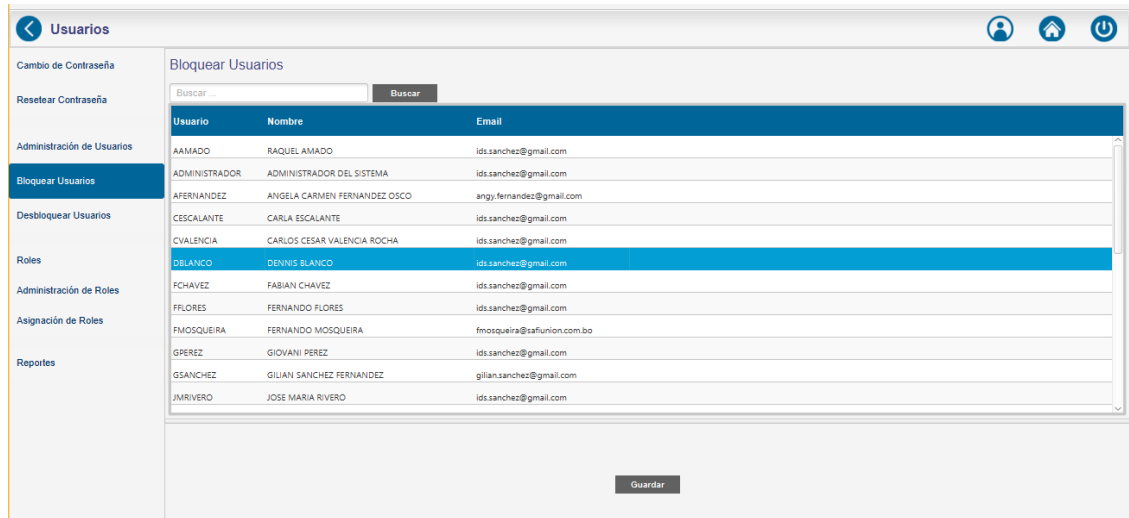
**Figura 3.31**  
*Asignación de roles de usuario*



Nota: Pantalla de asignación de roles a los usuarios

## Página bloquear cuenta

**Figura 3.32**  
*Bloquear cuenta de usuario*



Nota: Pantalla de bloqueo de usuarios

## Página cambio de password

**Figura 3.33**  
*Cambio de password de usuario*

Contraseña Anterior

Nueva Contraseña

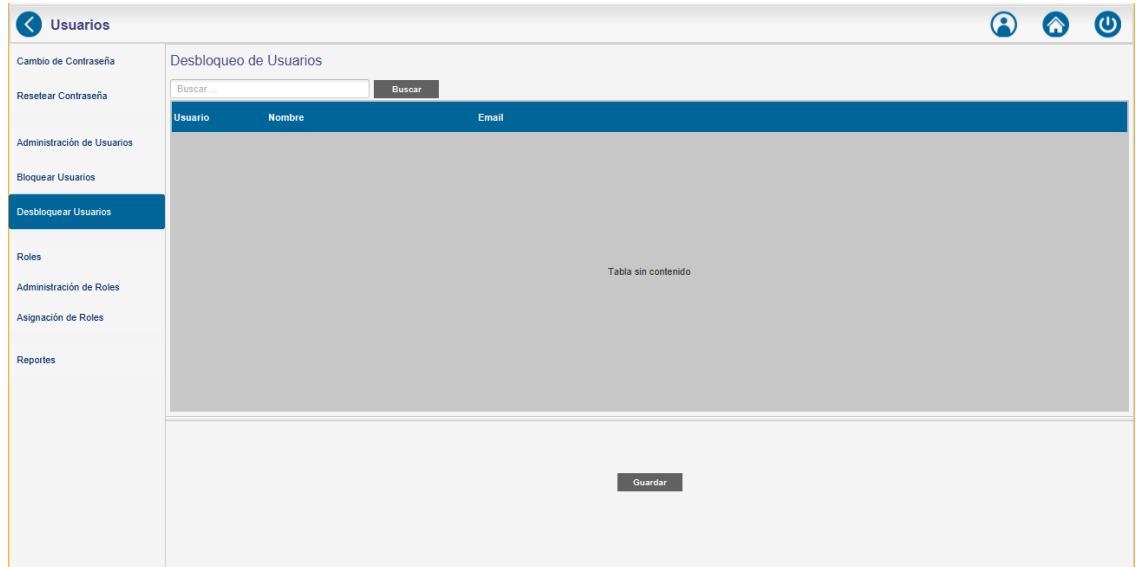
Nueva Contraseña

Nota: Formulario de cambio de contraseña



## Página desbloquear usuarios

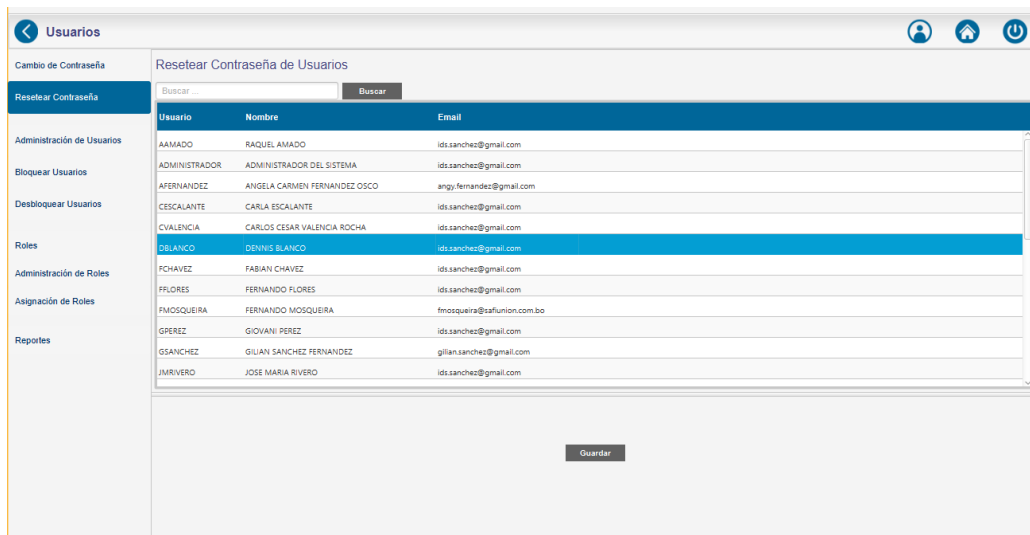
**Figura 3.34**  
Desbloquear usuarios



Nota: Pantalla de desbloqueo de usuarios

## Página resetear password

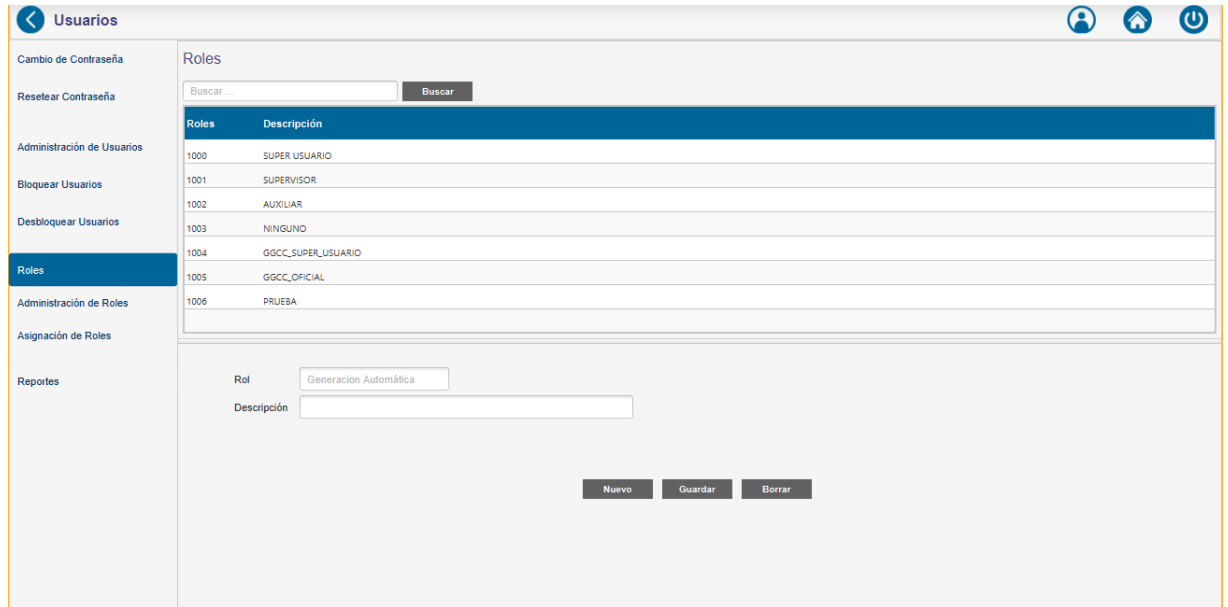
**Figura 3.35**  
*Reseteo de password de usuario*



Nota: Pantalla de reseteo de password

## Página roles de usuario

**Figura 3.36**  
*Roles de usuario*



Nota: pantalla de creación de roles para los usuarios

## Página de reportes

**Figura 3.37**

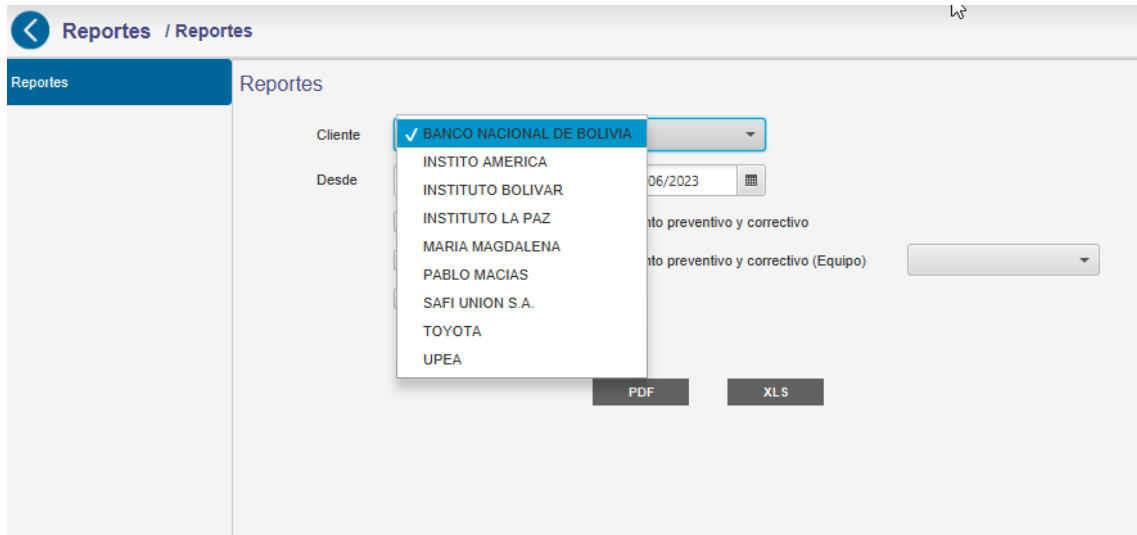


*Página de reportes*

Nota Pantalla de generación de reportes

## Selección de cliente para reporte

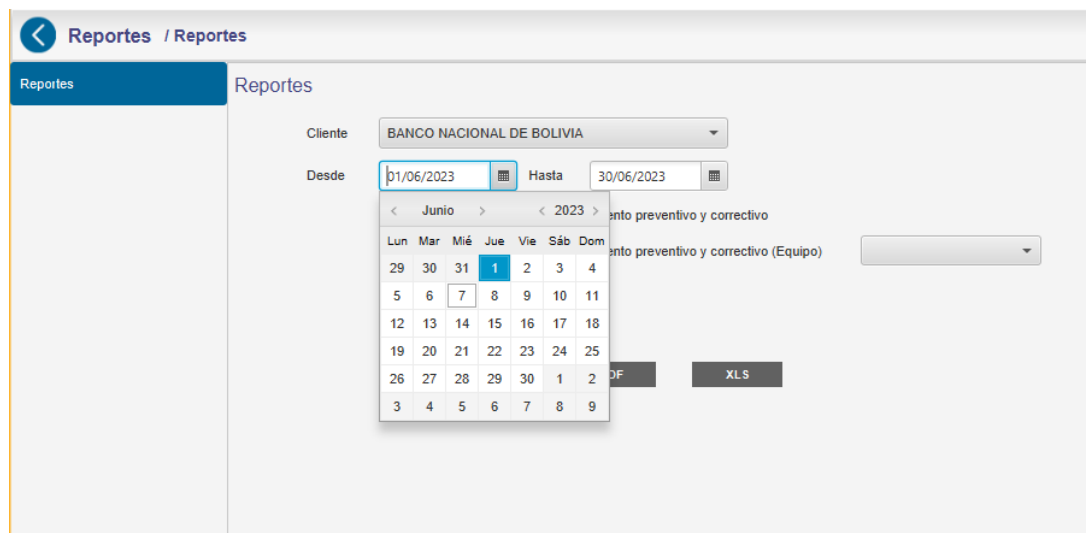
**Figura 3.38**  
*Selección de cliente para reporte*



Nota Seleccionando cliente para el reporte

## Selección de fecha para reporte

**Figura 3.39**  
*Selección de fecha para reporte*



Nota Selección de rango de fechas para el reporte

## Selección de informe de mantenimiento preventivo y correctivo

**Figura 3.40**

Selección de informe de mantenimiento preventivo y correctivo

The screenshot shows a web interface for generating reports. At the top left, there is a navigation breadcrumb 'Reportes / Reportes' with a back arrow. Below this is a sidebar with 'Reportes' highlighted. The main content area is titled 'Reportes' and contains the following elements:

- Ciente:** A dropdown menu with 'BANCO NACIONAL DE BOLIVIA' selected.
- Desde:** A date input field with '01/06/2023' and a calendar icon.
- Hasta:** A date input field with '30/06/2023' and a calendar icon.
- Report Type Selection:**
  - Informe de servicios de mantenimiento preventivo y correctivo
  - Informe de servicios de mantenimiento preventivo y correctivo (Equipo)
  - Informe Técnico # Servicio
- Export Buttons:** Two buttons labeled 'PDF' and 'XLS'.

Nota: Selección de tipo de reporte requerido

## Selección de informe de mantenimiento preventivo y correctivo por equipo

**Figura 3.41**

Selección de informe de mantenimiento por equipo

This screenshot is similar to the previous one, but with the following differences:

- The **Report Type Selection** is now:
  - Informe de servicios de mantenimiento preventivo y correctivo
  - Informe de servicios de mantenimiento preventivo y correctivo (Equipo)
  - Informe Técnico # Servicio
- The **Equipment Selection:** A dropdown menu is open, showing a list of equipment types: 'CELULAR', 'COMPUTADORA', 'FOTOCOPIADORA', 'IMPRESORA', 'LAPTOP', 'MULTIFUNCIONAL', and 'TELEVISION'. The 'CELULAR' option is selected, indicated by a checkmark at the bottom of the dropdown.
- The **Export Buttons:** 'PDF' and 'XLS' buttons are still present.

Nota Selección de segundo tipo de reporte

## Selección de informe técnico

**Figura 3.42**  
*Selección de informe técnico*

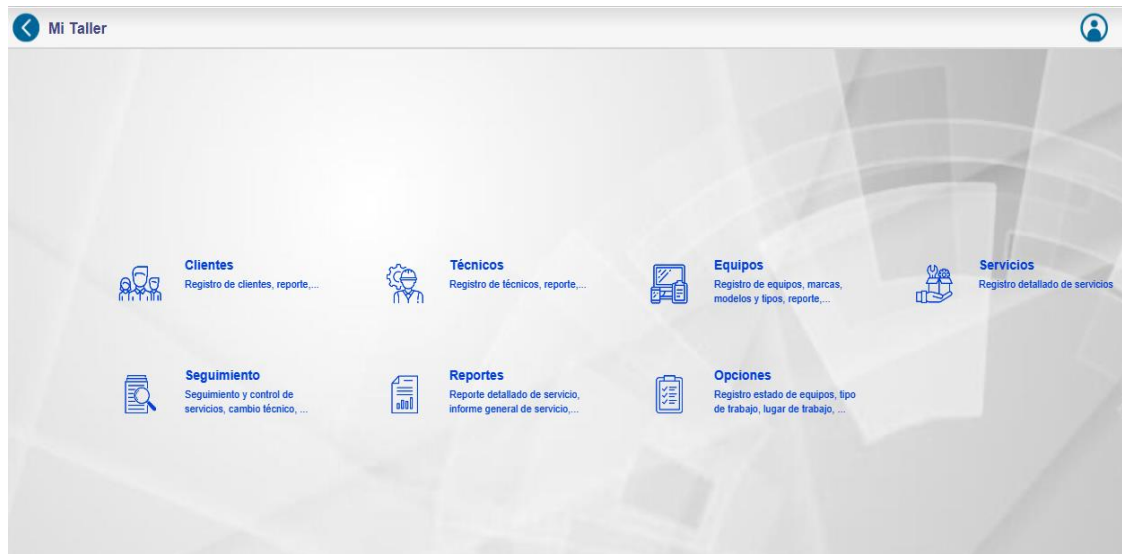
The screenshot shows a web interface for selecting a report. At the top left, there is a back arrow and the text 'Reportes / Reportes'. Below this, a blue header bar contains the word 'Reportes'. The main content area is titled 'Reportes' and contains the following elements:

- Cliente:** A dropdown menu with 'BANCO NACIONAL DE BOLIVIA' selected.
- Desde:** A date picker showing '01/06/2023'.
- Hasta:** A date picker showing '30/06/2023'.
- Informe de servicios de mantenimiento preventivo y correctivo:** An unchecked checkbox.
- Informe de servicios de mantenimiento preventivo y correctivo (Equipo):** An unchecked checkbox.
- Informe Técnico:** A checked checkbox with a text input field containing '30'.
- Buttons:** Two buttons labeled 'PDF' and 'XLS' are positioned at the bottom right.

Nota: Selección de reporte por técnico

## Pantalla de menú principal

**Figura 3.43**  
*Página de menú principal*



Nota Menú del modulo de taller

## Pantalla de clientes

**Figura 3.44**  
*Página de clientes*

The screenshot displays the 'Clientes' page. At the top, there is a navigation bar with a back arrow, the title 'Clientes', and three icons (user, home, power). Below this, there are two tabs: 'Registro de Clientes' and 'Reporte de Clientes'. The 'Registro de Clientes' tab is active, showing a search bar with the text 'Buscar...' and a 'Buscar' button. Below the search bar is a table with the following columns: Código, Nombre, Dirección, Telefono, Correo, Contacto, Fiscal de Serv..., Teléfono, and Celular. The table contains two rows of data:

Código	Nombre	Dirección	Telefono	Correo	Contacto	Fiscal de Serv...	Teléfono	Celular
1	BANCO NACIONAL DE BOLIVIA	C/COLOM 1025 Z/CENTRAL	23366363	info@bnb.com.bo	LUIS PEREZ	JUANES LOPEZ	2522525	7012585
2	SAFI UNION S.A.	C/FEDERICO SUAZO 1526 Z/CENTRAL	3322855	INFO@SAIUNION.COM	WILDER CAMPOS	ANTONIO SANDOV...	32565656	7552255

Below the table, there are four tabs: 'General', 'Otros Datos', 'Sucursales', and 'Equipos'. The 'General' tab is active, showing a form with the following fields: 'Código' (with a dropdown menu showing '0'), 'Nombre' (with a text input field containing 'Nombre del Cliente'), 'Dirección' (with a text input field), 'Teléfono' (with two sub-inputs: 'Telefono' and 'Correo'), and 'Contacto' (with a text input field containing 'Correo Electronico'). At the bottom of the form, there are three buttons: 'Nuevo', 'Guardar', and 'Borrar'. The footer of the page contains the text 'Acerca de ...'.

Nota Formulario de creación de clientes

## Pantalla técnicos

**Figura 3.45**  
*Página de técnicos*

The screenshot displays the 'Técnicos' page. At the top, there is a navigation bar with a back arrow, the title 'Técnicos', and three icons (user, home, power). Below this, there are two tabs: 'Registro de Técnicos' and 'Reporte de Técnicos'. The 'Registro de Técnicos' tab is active, showing a search bar with the text 'Buscar...' and a 'Buscar' button. Below the search bar is a table with the following columns: ID, Carnet Identidad, Nombre, Teléfono, Celular, Dirección, and Usuario Sistema. The table contains three rows of data:

ID	Carnet Identidad	Nombre	Teléfono	Celular	Dirección	Usuario Sistema
2	25689578 LP	JAIME TOLA	70256985	25698745	C/LOS PALITOS 25698 Z/ MACHETITOS Y PISTOLAS	JTOLA
1	22556932255 LP	JOSE SANCHEZ	2543355	7025485	C/04 JUNIO NRO 26 Z/ALTO MARIISCAL SANTA CRUZ	JSANCHEZ
3	332665262 CB	VICTOR SANCHEZ CALLIZAYA	24444444	6025552	C/LOS ENANITOS VERDES 9696 Z/ALDEA PITUFA	VJSANCHEZ

Below the table, there are several tabs: 'General', 'Otros Datos', 'Sucursales', and 'Equipos'. The 'General' tab is active, showing a form with the following fields: 'Código' (with a dropdown menu showing 'ID Técnico'), 'Carnet Identidad' (with a text input field), 'Nombre' (with a text input field), 'Teléfono' (with two sub-inputs: 'Telefono' and 'Celular'), 'Dirección' (with a text input field), and 'Usuario Sistema' (with a text input field). At the bottom of the form, there are three buttons: 'Nuevo', 'Guardar', and 'Borrar'. The footer of the page contains the text 'Acerca de ...'.

Nota Formulario de creación de técnicos

## Pantalla de equipos

Figura 3.46  
Página de equipos

The screenshot shows the 'Equipos' management interface. On the left, there is a sidebar with navigation links: 'Registro de Equipos', 'Reporte de Equipos', 'Registro Marcas de Equipos', 'Registro Modelos de Equipos', and 'Registro Tipos de Equipos'. The main area contains a table with columns 'Código', 'Nombre', and 'Dirección'. Two rows are visible:

Código	Nombre	Dirección
1	BANCO NACIONAL DE BOLIVIA	C/COLON 1025 Z/CENTRAL
2	SAFI UNION S.A.	C/FEDERICO SUAZO 1526 Z/CENTRAL

Below the table is a form for adding or editing an equipment record. The form includes fields for 'Código' (ID Equipo), 'Referencia' (Nombre del Equipo), 'Sucursal', 'Tipo', 'Marca', 'Modelo', 'N° Activo Fijo', 'N° Serie', and 'Contometro'. At the bottom of the form are buttons for 'Nuevo', 'Guardar', and 'Borrar'.

Nota Elaboración propia

## Pantalla de servicios

Figura 3.47  
Página de servicios

The screenshot displays the 'Servicio' management interface. The left sidebar has navigation options: 'Registro de Servicios', 'Reporte de Servicios', 'Registro Tipos de Servicios', and 'Registro Modelos de Servicios'. The main area shows a table with columns 'ID', 'Activo Fijo', and 'Nro Serie'. Two rows are visible:

ID	Activo Fijo	Nro Serie
2	ASASASSA	1234-W56R-12548W
1	ASAS23232AS	123-AS2-2565

Below the table is a form for creating a service record. The form includes fields for 'Id', 'Fecha Reg', 'Fecha Servicio', 'Formulario', 'Fe Estimada', 'Responsable', 'Nro Tel/Cel', 'Problema', 'Ubicación', 'Estado Equipo', 'Trabajo', 'Lugar Atención', 'Técnico', and 'Contometro'. At the bottom of the form are buttons for 'Nuevo', 'Guardar', and 'Borrar'.

Nota Formulario de creación de servicios

## Pantalla de seguimiento

Figura 3.48  
Página de seguimiento

The screenshot shows the 'Seguimiento' (Tracking) page. On the left, there is a sidebar with navigation options: 'Seguimiento de Servicio', 'Asignación de Técnico', 'Cambio de Técnico', 'Reemplazo de Partes', 'Cierre de Servicio', and 'Consulta de Servicio'. The main area contains a table with columns: ID, Formulario, Fe Servicio, Fe Estimada, and Tecnico. The table lists various service records with their respective IDs, form numbers, and dates. On the right, there is a form with tabs for 'General', 'Otros Datos', and 'Reemplazo de Partes'. The 'General' tab is active, showing fields for 'Id' (22), 'Fecha Reg' (26/03/2021 16:43:43), 'Cliente' (SAFI UNION S.A.), 'Equipo' (HP LAPTOP ENVY 15.9 PULGADAS), 'Formulario' (66969), 'Fe Estimada' (01/01/2021), 'Responsable' (-), 'Ubicación' (ZONA SUR), 'Problema' (SADHSAJDHSIAHDJ), 'Estado Equipo' (EN ESPERA DE PIEZA), 'Lugar Atención' (DOMICILIO), 'Trabajo' (MANTENIMIENTO), and 'Técnico' (VICTOR SANCHEZ CALLIZAYA). Below the form, there is a table with columns: Fecha, Estado Equipo, and Trabajo Realizado. The table shows two entries: one for 'Cierre de Servicio: 22' and another for 'Asignación de Técnico: VICTOR SANCHEZ CALLIZAYA'.

Nota Módulo de seguimiento

## Pantalla de dashboard

Tabla 3. 36

Página de dashboard

The screenshot shows the 'Dashboard' page with a table titled 'Servicios Pendientes'. The table has columns for client information, service details, technician, and dates. The data is as follows:

Y.P.F.B	FOTOCOPIADORA	CANON	IR 2525	Edificio corporativo	NO SACA COPIAS	27/11/202	17/11/202	
Y.P.F.B	FOTOCOPIADORA	CANON	IR 2525	Edificio corporativo	BANDEJA DE FOTOCOPIADORA NO JALA PAPEL	30/10/202	23/11/202	
TIENDA EPS	DATASHOW	EPSON	POWERLITE S12.PLUS	OFICINA EPS	JOSE SANCHEZ CALLISAYA	FOCO DAÑADO TIENE UNA MANCHA AMARILLA	02/11/202	25/11/202
EPSAS	FOTOCOPIADORA	CANON	IR 2525	BATALLON COLORADOS	JAIME TOLA	MANTENIMIENTO GENERAL	31/10/202	10/12/202

Nota Dashboard con tareas pendientes



## 3.11 USO DE MICROSERVICIOS

### 3.11.1 Aplicando microservicios

Se implementó la arquitectura de microservicios para el desarrollo y mejor funcionamiento del sistema, cada microservicio se encarga de una funcionalidad específica como agregar, edita y eliminar datos para cada módulo del sistema y se comunica con otros microservicios a través de la base de datos.

Los microservicios son una arquitectura de software que divide una aplicación en servicios más pequeños y manejables, cuando esta arquitectura se aplica a una base de datos Oracle, cada microservicio se convierte en una entidad independiente.

A continuación, se puede observar los microservicios implementados y dedicados para cada módulo del presente sistema:

**Figura 3.49**

Microservicios módulo de usuarios

```
create or replace NONEDITIONABLE PACKAGE BODY          CONTEXT_API IS
PROCEDURE set_parameter (p_name  IN VARCHAR2,
                        p_value  IN VARCHAR2) IS
BEGIN
    DBMS_SESSION.set_context('parameter_ctx', p_name, p_value);
END set_parameter;

PROCEDURE clear_context (p_name  IN VARCHAR2) IS
BEGIN
    DBMS_SESSION.clear_context('parameter_ctx', attribute => p_name);
END clear_context;

PROCEDURE clear_all_context IS
BEGIN
    DBMS_SESSION.clear_all_context('parameter_ctx');
END clear_all_context;
END context_api;
```

Nota Código fuente generación de microservicios para usuarios

**Figura 3.50**  
Microservicios marca y cliente

```
create or replace NONEDITIONABLE PACKAGE          CRUD IS
  eError EXCEPTION;

FUNCTION Mar( ID NUMBER,
             cDesc VARCHAR2,
             cDML VARCHAR2 ) RETURN VARCHAR2;

FUNCTION Cli ( ID NUMBER,
              names VARCHAR2,
              dir VARCHAR2,
              fono VARCHAR2,
              email VARCHAR2,
              contac VARCHAR2,
              fis VARCHAR2,
              fisTel VARCHAR2,
              fisCel VARCHAR2,
              fis1 VARCHAR2,
              fis1Tel VARCHAR2,
              fis1Cel VARCHAR2,
              cDML VARCHAR2 ) RETURN VARCHAR2;
```

Nota Código fuente generación de microservicios para marca y clientes

**Figura 3.51**  
Microservicios equipos y técnicos

```
FUNCTION Eqp ( ID NUMBER,
              idCli VARCHAR2,
              cName VARCHAR2,
              idTip VARCHAR2,
              idMar VARCHAR2,
              idMdl VARCHAR2,
              aFijo VARCHAR2,
              serie VARCHAR2,
              suc VARCHAR2,
              conto NUMBER,
              cDML VARCHAR2 ) RETURN VARCHAR2;

FUNCTION Tec ( ID NUMBER,
              names VARCHAR2,
              tel VARCHAR2,
              cel VARCHAR2,
              dir VARCHAR2,
              ci VARCHAR2,
              usu VARCHAR2,
              cDML VARCHAR2 ) RETURN VARCHAR2;
```

Nota Código fuente generación de microservicios para equipos y técnicos

**Figura 3.52**  
Microservicios servicios

```
FUNCTION Ser ( ID NUMBER,  
              idCli NUMBER,  
              idEqp NUMBER,  
              idSuc NUMBER,  
              idEst NUMBER,  
              idTe NUMBER,  
              idLug NUMBER,  
              idTra NUMBER,  
              cForm VARCHAR2,  
              cRes VARCHAR2,  
              conto NUMBER,  
              cProb VARCHAR2,  
              cDiag VARCHAR2,  
              cReaT VARCHAR2,  
              cObs VARCHAR2,  
              cRef VARCHAR2,  
              dFeEst DATE,  
              dFeSer DATE,  
              cTel VARCHAR2,  
              cierre VARCHAR2,  
              cDML VARCHAR2 ) RETURN VARCHAR2;
```

Nota Código fuente generación de microservicios para servicios



**CAPÍTULO IV**  
**CALIDAD COSTOS**  
**SEGURIDAD**

## 4. CALIDAD COSTO Y SEGURIDAD

### 4.1 Introducción

Las métricas de calidad son una herramienta fundamental para medir el desempeño y la calidad de un software que se utilizan para asegurar que el software cumpla con los requisitos del usuario, sea confiable, seguro, eficiente, fácil de usar y fácil de mantener. También se describirán las herramientas que se utilizarán para recolectar y analizar los datos de las métricas, lo que permitirá una evaluación precisa y objetiva del software desarrollado para la empresa E.P.S. S.R.L.

### 4.2 Aplicación de la Métrica de Calidad ISO 25000

Es de suma importancia asegurar la calidad del sistema desarrollado. Para lograrlo se trabajó bajo las métricas de calidad de la ISO 25000 que permitan evaluar y medir de manera objetiva el cumplimiento de los estándares y requisitos establecidos. En este sentido, se ha aplicado los métodos propuestos en el libro "Ingeniería del Software. Un Enfoque Práctico" de Roger S. Pressman.

#### 4.2.1 Funcionalidad

Se realiza la evaluación de la funcionalidad del sistema desarrollado de acuerdo a la formula.

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

Para medir la funcionalidad se mide primero las métricas de punto de función (PF) para luego obtener el conteo total de las mismas. Los valores de dominio de información se definen de la siguiente manera:

- Número de entradas de usuario.

Cada entrada se origina de un usuario, las entradas deben distinguirse de las consultas que se cuentan por separado.

Se muestra los resultados en la siguiente tabla:

**Tabla 4.1**  
*Número de entradas de Usuario*

Módulo	Cantidad de entradas
Usuarios	4
Técnicos	5
Equipos	6
Servicios	7
Clientes	8
Reportes generados	3
Total	33

Nota Tabla de entradas generadas por el usuario

- Número de salidas de usuario.

Cada salida se origina de un usuario, las salidas deben distinguirse de las consultas que se cuentan por separado.

Se muestra los resultados en la siguiente tabla:

**Tabla 4.2**  
*Número de salidas de Usuario*

Módulo	Cantidad de salidas
Usuarios	4
Técnicos	5
Equipos	6
Servicios	7
Clientes	8
Reportes generados	3
Total	33

Nota Tabla de salidas generadas por el usuario

➤ Número de consultas de usuario.

Son las entradas interactivas en los módulos que da como resultado la generación de una respuesta al usuario

Se muestra los resultados en la siguiente tabla:

**Tabla 4.3**

*Número de entradas de Usuario*

Módulo	Cantidad de consultas
Usuarios	2
Técnicos	3
Equipos	4
Servicios	5
Clientes	6
Reportes generados	2
<b>Total</b>	<b>22</b>

Nota Tabla de consultas realizadas por el usuario

Con los datos obtenidos anteriormente se realiza el conteo total del punto fusión.

**Tabla 4.4**

*Complejidad funcional de cada modulo*

Módulo	Entradas	Salidas	Consultas	Fi
Usuarios	4	4	2	10
Técnicos	5	5	3	13
Equipos	6	6	4	16
Servicios	7	7	5	19
Clientes	8	8	6	22
Reportes generados	3	3	2	8
<b>Total</b>	<b>33</b>	<b>33</b>	<b>22</b>	<b>88</b>

Nota Tabla punto de fusión por modulo

La columna "Fi" representa la complejidad funcional de cada módulo.

El siguiente paso es medir las preguntas de complejidad, se realiza la ponderación de las variables que serán evaluadas según la siguiente tabla:

**Tabla 4.5**  
*Ponderación de Valores*

<b>Categoría</b>	<b>Valor</b>
No importante	0
Menor importancia	1
Moderado	2
Medio	3
Significativo	4
Esencial	5

Nota Tabla de ponderación de valores

Se responde a las interrogantes las siguientes preguntas de complejidad:

**Tabla 4.6**  
*Preguntas de complejidad*

<b>N°</b>	<b>Preguntas de Complejidad</b>	<b>Valor</b>
1	¿El proceso de registro de un nuevo usuario requiere de validaciones adicionales, como, por ejemplo, la verificación de un correo electrónico?	4
2	¿El proceso de asignación de técnicos a los diferentes servicios es automatizado o debe realizarse de forma manual?	3
3	¿El módulo de equipos tiene la funcionalidad de generar alertas en caso de que algún equipo presente fallas?	5
4	¿El módulo de servicios permite la selección de múltiples servicios para un mismo cliente?	2
5	¿El módulo de clientes cuenta con la funcionalidad de búsqueda de clientes por diferentes criterios, como nombre, apellido o número de teléfono?	2



6	¿Los reportes generados tienen la capacidad de filtrar la información por fechas o por algún otro criterio específico?	3
7	¿El módulo involucra lógica de negocio compleja?	4
8	¿El módulo implica manipulación de grandes volúmenes de datos?	4
9	¿El módulo requiere integración con otros sistemas o servicios externos?	3
10	¿El módulo requiere una interfaz de usuario altamente interactiva y personalizada?	5
11	¿El módulo tiene requisitos de seguridad y privacidad muy exigentes?	4
12	¿El módulo tiene requisitos de disponibilidad y rendimiento muy altos?	4
TOTAL= $\sum(F_i)$ =		43

Nota Preguntas de complejidad elaboradas para el sistema

Calculo de punto fusión PF

$$PF = cuenta (grado de confiabilidad + Tasa de error * \sum (F_i))$$

Donde:

Cuenta total = Total de puntos función

Grado de Confiabilidad = Valor de 0.65

Tasa de Error = Valor 0.01, (Error de confiabilidad de la aplicación)

$\sum (F_i)$  = Valor Total de la complejidad de la aplicación

En este caso, el valor del PF se calcula utilizando la fórmula  $PF = \text{conteo total} * [0.65 + 0.01 \sum (F_i)]$ , donde el conteo total es el número de entradas, salidas, consultas y archivos lógicos del sistema, y  $F_i$  es la ponderación de las preguntas de complejidad. Para el sistema CRM desarrollado anteriormente, se obtiene un conteo total de 88, y la sumatoria de las ponderaciones de las preguntas de complejidad es de 43.

Remplazando valores en la formula, se tiene el siguiente resultado:

$$PF=88 * [0.65 + 0.01 * 43]$$

$$PF= 88 * 1.08$$

$$PF=94.944$$

Por lo tanto, el tamaño del sistema según el método Cosmic Function Point es de aproximadamente 94.9 CFP.

Calculando el punto fusión con el valor máximo de  $\sum (Fi)$

$$PF Esperada=88 * [0.65 + 0.01 * 70]$$

$$PF Esperada=88*1.35$$

$$PF= 118.8$$

Por lo tanto, el Punto de Función ajustado para el sistema CRM sería de 118.8. Este valor representa una medida de la funcionalidad total del sistema, considerando tanto las entradas, salidas y consultas realizadas por cada uno de los módulos, así como los reportes generados.

Con el máximo valor de ajuste de complejidad se tiene la funcionalidad real:

$$Funcionalidad = PF/PFmax$$

Para calcular la funcionalidad del sistema, se divide el punto de función calculado entre el máximo punto de función teórico, lo que resulta en:

$$Funcionalidad = 94.944 / 118.8 = 0.798$$

$$Funcionalidad = 0,798 * 100 = 79.82\%$$

Por lo tanto, la funcionalidad del sistema es del 79.82%.

#### 4.2.2 Usabilidad

La usabilidad del sistema es la facilidad de uso que tiene el mismo, esta métrica nos muestra el costo de aprender el programa desarrollado, lo cual se calcula con la siguiente formula:

$$FU = \left( \frac{\sum X_{in}}{n} \right) * 100$$

Realizamos las preguntas según pressman con una ponderación del 1 al 5 como máximo.

**Tabla 4.7**  
*Ajuste de preguntas*

Nº	Preguntas	Ponderación
1	¿Es fácil para los usuarios aprender a utilizar el sistema?	4
2	¿Es fácil para los usuarios recordar cómo utilizar el sistema?	4
3	¿Es fácil para los usuarios utilizar el sistema correctamente?	3
4	¿Es fácil para los usuarios encontrar la información que necesitan en el sistema?	4
5	¿Es fácil para los usuarios entender la información que se muestra en el sistema?	4
6	¿Es fácil para los usuarios realizar tareas comunes en el sistema?	4
7	¿Es fácil para los usuarios personalizar el sistema para adaptarlo a sus necesidades?	3
8	¿Es fácil para los usuarios recuperarse de errores en el sistema?	4

Nota Tabla de preguntas ajustadas al sistema

Reemplazando en la fórmula:

Entonces, la puntuación total de usabilidad sería:

$$(4 + 4 + 3 + 4 + 4 + 4 + 3 + 4) / 8 = 3,875$$

Puntuación total = 3.875

Puntuación máxima posible = 5

$$\text{Proporción} = \frac{\text{Puntuación total}}{\text{Puntuación máxima posible}} = \frac{3,875}{5}$$

Proporción = 0,775

Para obtener el porcentaje, multiplicamos la proporción por 100:

$$FU = (0.775) * 100 = 77.5\%$$

Porcentaje = Proporción \* 100 Porcentaje = 0,775 \* 100 Porcentaje = 77,5%

Por lo tanto, la usabilidad del sistema desarrollado según las preguntas de Pressman es del 77,5% en una escala de 0% a 100%.

#### **4.2.3 Confiabilidad**

La confiabilidad del sistema se define como la probabilidad de que un sistema, activo o competente lleve a cabo su función adecuadamente durante un periodo bajo condiciones operacionales previamente definidas y constantes.

Para calcular la confiabilidad del sistema se toma en cuenta los valores hallados anteriormente, así como el periodo de tiempo, de la siguiente manera:

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

Donde:

f, funcionalidad del sistema. (80)

$\mu$ , Probabilidad de error que puede tener el sistema. (0.01)

t, tiempo que dura una gestión en el sistema (20 días)

Reemplazando los valores:

$$F(20) = 79.82 * e^{-0.1*20}$$

$$F(20) = 79.82 * 0.1353$$

$$F(20) = 10.83$$

Esto significa que la probabilidad de que el sistema falle en un período de 20 días es del 10.83 %.

La probabilidad de que el sistema funcione sin fallas en lo 20 dias se calcula como:

$$P(T \leq 20) = 1 - F(20)$$

Reemplazando los valores:

$$P(T \leq 20) = 1 - 0.1083 = 0.8917$$

Esto significa que la probabilidad de que el sistema desarrollado funcione sin fallas en un período de 20 días es del 89,17%, por lo tanto, el sistema tiene un índice de confiabilidad del 89,17 % en este período de 20 días de prueba.

#### 4.2.4 Mantenibilidad

El mantenimiento del software se desarrolla para mejorar la respuesta a los nuevos requerimientos que la empresa tenga. El estándar IEE94 nos sugiere un índice de madurez del software (IMS) que proporciona un indicador en la estabilidad de un producto, se lo determina con la siguiente formula:

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

Donde:

Mt = Cantidad de módulos de la versión actual. (5)

Fa = Cantidad de módulos en la versión actual que se han añadido. (1)

Fc = Cantidad Número de módulos en la versión actual que se han cambiado. (0)

Fd = Cantidad Número de módulos de la versión anterior que se han borrado en la versión actual. (0)

Reemplazando los valores obtenidos durante el desarrollo del sistema:

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

$$IMS = 0.8$$

Por tanto, se puede decir el sistema desarrollado tiene una estabilidad de 80% que indica la facilidad de mantenimiento el otro 20% nos indica el margen de error correspondiente a las modificaciones realizadas.

#### **4.2.5 Evaluación global**

Según Roger S. Pressman autor del libro "Ingeniería del software: Un enfoque práctico" un software se considera "satisfactorio" cuando cumple con los requisitos establecidos y proporciona las funcionalidades y características necesarias de manera adecuada.

Dado los anteriores resultados encontramos que la evaluación global del sistema es:

**Tabla 4.8**

*Tabla evaluación global*

<b>Funcionalidad</b>	79.82%
<b>Usabilidad</b>	77.5%
<b>Confiabilidad</b>	89.17%
<b>Mantenibilidad</b>	80%
<b>Total</b>	81.62%

Nota Evaluación final del sistema

Se puede decir que el sistema desarrollado tiene un puntaje de 81.62% por lo tanto se considera satisfactorio.

### **4.3 ESTIMACIÓN DEL COSTO**

#### **4.3.1 COCOMO II**

Aplicando la calidad de costos COCOMO II al sistema desarrollado se necesitaron de varios datos, entre ellos el número estimado de líneas de código, el esfuerzo aplicado en persona-mes y el tiempo de desarrollo.

Las ecuaciones son:

$$E = a * KLDC^b \text{ (mes persona)}$$

$$D = c * Ed \text{ (mes)}$$

$$P = E/D \text{ (persona)}$$

Donde:

E= Es el esfuerzo aplicado en personas-mes,

D= Es el tiempo de desarrollo utilizado en meses cronológicos

KLDC = El número estimado de líneas de código distribuidas para el desarrollo del proyecto.

A, B, C, D = Valores constantes.

Dado que el sistema tiene 6 módulos y se trabajó durante un aproximado de 6 meses con 4 horas al día, podemos estimar que el desarrollo es igual a:

$$6 \text{ meses} \times 4 \text{ horas/día} \times 30 \text{ días/mes} / 8 \text{ horas/día} = 90 \text{ días} = 3 \text{ meses}$$

Para el número estimado de líneas de código se realizó una estimación del tamaño del sistema.

El sistema tiene un tamaño medio,

Tabla

**Tabla 4.9**

*Cantidad de línea de código*

<b>Descripción</b>	<b>Cantidad</b>
Modulo usuarios	2252
Modulo técnicos	1745
Modulo servicios	1453
Modulo equipos	1544
Modulo servicios	1657
Modulo reportes	938
Total	9589

Nota Cantidad de líneas de código generadas para el sistema

Se estima un valor de LDC de alrededor de 9589.

Convirtiendo a KLDC se tiene:

$$KLDC = LDC/1000$$

$$KLDC = 9589/1000$$

$$KLDC = 9.6$$

Para los valores constantes A, B, C y D

Sabiendo que el proyecto es de tamaño medio y que el equipo tiene experiencia y habilidades adecuadas, podemos seleccionar los siguientes valores para las variables:

$$A = 2,94$$

$$B = 0,91$$

$$C = 3,67$$

$$D = 0,28.$$

Con estos datos, podemos calcular el esfuerzo mes-persona de la siguiente manera:



$$E = A \times (KLDC)^B \times (CD)$$

$$E = 2,94 \times (9.6)^{0,91} \times (3,67 \times 0,28)$$

$$E = 4.99 \text{ personas-mes}$$

Con los datos obtenidos se calcula el tiempo de desarrollo en meses.  $D = c * E^d$

$$D = 2.5 (4.99)^{0.38}$$

$$D = 2.96$$

$$D = 2.9 \text{ meses}$$

$$D = 3 \text{ meses}$$

Calculando la cantidad de personas se tiene:

$$P = \frac{E}{D}$$

$$P = \frac{4.99}{3}$$

$$P = 1.66$$

Por lo tanto, redondeado personas se determinó que  $P=2$  personas

Suponiendo que el salario promedio de un profesional junior del área es de 3500 bs, se tiene el costo:

$$\text{Costo total Salario} = \text{Salario} * P * D$$

$$\text{Costo total Salario} = 3500 * 2 * 3$$

$$\text{Costo total Salario} = 21000 \text{ bs.}$$

#### 4.3.2 Costo de elaboración

El gasto realizado en la elaboración del proyecto fue de:

**Tabla 4.10**

*Costo beneficio del proyecto*

<b>Detalle</b>	<b>Costo</b>
Análisis del proyecto	300
Material de escritorio	150
Internet	900
Otros gastos	150
<b>Total</b>	<b>1500</b>

Nota Tabla de costo de elaboración del proyecto

El costo total de elaboración es 1500 bs.

### 4.3.3 Costo total del proyecto

Para obtener el costo total del proyecto se realiza la sumatoria de los distintos costos obtenidos.

Costo Total=Costo Total Salario +Costo Total Elaboración

Costo Total=21000 bs +1500 bs

Costo Total=22500 bs

**Tabla 4.11**

*Costo total del proyecto*

<b>Detalle</b>	<b>Costo</b>
Esfuerzo	2 personas
Tiempo	3 meses
Costo total	22500 bs

Nota Costo en tiempo y esfuerzo del sistema

Por lo que se obtiene que el costo final del proyecto denominado SISTEMA PARA LA GESTIÓN DE CLIENTES MEDIANTE EL USO DE MICROSERVICIOS, A TRAVÉS DEL MODELO DE NEGOCIO CRM es de 22500 Bs.

## 4.4 Seguridad

### 4.4.1 ISO 27000:2022

Aplicando la norma ISO 27001:2022 que establece los requisitos para un Sistema de Gestión de Seguridad de la Información (SGSI) se implementaron los siguientes controles de seguridad de la información en el proyecto.

### Roles de usuarios

Para la implementación del sistema se definieron 3 niveles de roles:

**Usuario admin** Tiene acceso a todas las funcionalidades del sistema, así como el poder crear nuevos usuarios y asignar sus respectivos roles.

**Usuario sistem** Tiene acceso al módulo de clientes, equipos, técnicos, servicios y reportes se encarga de registrar todos los datos, así como registrar y asignar los servicios a los técnicos.

**Usuario técnico** Los usuarios técnicos tienen acceso al módulo de seguimiento y reportes, ellos se encargan de dar de alta los servicios cuando estos terminen dichos trabajos, así como generar reportes para los clientes que consulten.

#### **4.4.2 Seguridad de base de datos**

La base de datos utilizada en el sistema desarrollado es Oracle, se han tomado las medidas de seguridad necesarias según la ISO 27001:2022 para garantizar la protección de los datos almacenados.

Se tomaron las siguientes medidas de seguridad:

- Cuando un usuario solicita acceder a registros de la base de datos, se establece una conexión segura para esta acción, lo que asegura la integridad y confidencialidad de la información.
- Se a encriptado todas las contraseñas en SHA-2, los cuales son considerados seguros y resistentes a diversos tipos de ataques criptográficos, actualmente no se han encontrado vulnerabilidades significativas en SHA-2.
- Además, se ha implementado un sistema de autenticación mediante un nombre de usuario y contraseña, con niveles de acceso determinados según las funciones que se realizan en el sistema
- Asimismo, para garantizar la seguridad de los datos, la conexión a la base de datos se cierra automáticamente después de cada sesión de usuario.

La información es un activo valioso para cualquier empresa, por lo que es esencial tomar medidas de seguridad apropiadas para protegerla.

### **4.4.3 Seguridad lógica**

Según la norma ISO 27001:2022, es necesario implementar controles lógicos para garantizar la protección de la información en los sistemas o software desarrollados

Se ha implementado:

- Una autenticación segura mediante nombre de usuario y contraseña para cada usuario del sistema
- Se han establecido diferentes niveles de acceso según las funciones y responsabilidades de cada usuario tendrá en el sistema.
- Se han implementado medidas de seguridad para prevenir el acceso no autorizado a la base de datos como el uso de firewalls.
- Se ha implementado un sistema de registro de actividad para supervisar las acciones realizadas en el sistema y detectar cualquier posible actividad malintencionada por parte del personal u personas ajenas a la empresa.
- Se ha establecido una política de mantenimiento constante para el software del sistema y la base de datos Oracle, con el fin de corregir posibles vulnerabilidades.

De esta manera, se cumple con los estándares de seguridad lógica establecidos por la norma ISO 27000:2022 para proteger la información del sistema desarrollado.

### **4.4.4 Seguridad física**

Se han tomado las medidas necesarias para garantizar la seguridad física de los equipos y dispositivos involucrados en la implementación del sistema.

Se implementó un servidor dedicado local con las especificaciones necesarias para correr el sistema.

El acceso al servidor donde se aloja la base de datos está restringido a personas autorizadas y se encuentra en una sala con controles del personal de la empresa.

#### 4.4.5 Backup de la base de datos

Durante el desarrollo del proyecto, se implementó medidas para garantizar la seguridad y disponibilidad de la base de datos.

Para llevar a cabo el respaldo de los datos se utilizó Oracle Recovery Manager (RMAN), una herramienta proporcionada por Oracle para la administración de bases de datos. Con esta herramienta se pudo crear copias de seguridad de la base de datos.

Se realizó un horario regular para el respaldo de la base de datos el cual se llevará a cabo de manera automática cada 2 días.

Además, se estableció una política de guardado adecuada para las copias de seguridad, asegurando de que se conserven durante un período de tiempo suficiente para cumplir con los requisitos de recuperación.

#### 4.5 Pruebas del sistema

##### 4.5.1 Prueba unitaria

Para la realización de la prueba unitaria se generó un código el cual simula hacer clic en un botón con el identificador #yourButtonId y luego verifica que el texto de un nodo con el identificador #yourTextNode cambie según lo esperado.

Figura 4.1

*Prueba unitaria*

```
import javafx.fxml.FXMLLoader;
import javafx.scene.Scene;
import javafx.stage.Stage;
import org.junit.jupiter.api.Test;
import org.testfx.api.FxRobot;
import org.testfx.api.FxToolkit;
import static org.junit.jupiter.api.Assertions.assertEquals;

public class MyControllerTest extends ApplicationTest {

    @Override
    public void start(Stage primaryStage) throws Exception {
        FXMLLoader loader = new FXMLLoader(getClass().getResource("/path/to/YourFXML.fxml"));
        primaryStage.setScene(new Scene(loader.load()));
        primaryStage.show();
    }

    @Test
    public void buttonClickShouldChangeText(FxRobot robot) {
        robot.clickOn("#yourButtonId");
        assertEquals("Texto Esperado Después del Clic", robot.lookup("#yourTextNode").getText());
    }

    @Override
    public void stop() throws Exception {
        FxToolkit.hideStage();
    }
}
```

Nota Código fuente del script de concurrencia

**Figura 4.2**

*Resultado obtenido prueba unitaria*

```
Running MyControllerTest
Nov 16, 2023 12:34:56 PM javafx.fxml.FXMLLoader$ValueElement processValue
WARNING: Loading FXML document with JavaFX API of version 16 by JavaFX runtime of version 16
...
Nov 16, 2023 12:34:57 PM com.sun.javafx.stage.StageHelper lambda$runTask$2
WARNING: Unsupported JavaFX configuration: classes were loaded from 'unnamed module @
Nov 16, 2023 12:34:57 PM com.sun.javafx.stage.StageHelper lambda$runTask$2
WARNING: Unsupported JavaFX configuration: classes were loaded from 'unnamed module @
Button clicked! Text is now: Texto Esperado Después del Clic

Tests run: 1, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 1.234 sec
```

Nota Resultado del script de concurrencia

### 4.5.2 Prueba de integración

Para la prueba se utilizó el siguiente código el cual simula realizar una serie de acciones que involucran varios componentes del sistema desarrollado.

**Figura 4.3**

*Prueba de integración*

```
import javafx.fxml.FXMLLoader;
import javafx.scene.Scene;
import javafx.stage.Stage;
import org.junit.jupiter.api.Test;
import org.testfx.api.FxRobot;
import org.testfx.api.FxToolkit;

public class IntegrationTest extends ApplicationTest {

    @Override
    public void start(Stage primaryStage) throws Exception {
        FXMLLoader loader = new FXMLLoader(getClass().getResource("/path/to/YourFXML"));
        primaryStage.setScene(new Scene(loader.load()));
        primaryStage.show();
    }

    @Test
    public void testIntegration(FxRobot robot) {
        robot.clickOn("#firstButton");
        robot.type(KeyCode.SHIFT, KeyCode.A);
        robot.clickOn("#secondButton");
    }

    @Override
    public void stop() throws Exception {
        FxToolkit.hideStage();
    }
}
```

Nota Código fuente del script de integracion

**Figura 4.4**

*Resultado obtenido prueba integración*

```
Running IntegrationTest
Nov 16, 2023 1:23:45 PM javafx.fxml.FXMLLoader$ValueElement processValue
WARNING: Loading FXML document with JavaFX API of version 16 by JavaFX runtime of version 16
...
Nov 16, 2023 1:23:46 PM com.sun.javafx.stage.StageHelper lambda$runTask$2
WARNING: Unsupported JavaFX configuration: classes were loaded from 'unnamed module'
Nov 16, 2023 1:23:46 PM com.sun.javafx.stage.StageHelper lambda$runTask$2
WARNING: Unsupported JavaFX configuration: classes were loaded from 'unnamed module'
Integration test completed successfully.

Tests run: 1, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 2.345 sec
```

Nota Resultados obtenidos script de integración

### 4.5.3 Prueba interfaz de usuario UI

La prueba de interfaz simula el ingreso un nombre de usuario y una contraseña, posteriormente se hace clic a iniciar sesión. Luego, verifica que el mensaje de inicio del sistema coincida con lo esperando.

**Figura 4.5**

*Prueba interfaz de usuario UI*

```
import javafx.fxml.FXMLLoader;
import javafx.scene.Scene;
import javafx.stage.Stage;
import org.junit.jupiter.api.Test;
import org.testfx.api.FxRobot;
import org.testfx.api.FxToolkit;
import org.testfx.framework.junit5.ApplicationTest;
import static org.junit.jupiter.api.Assertions.assertEquals;

public class LoginUITest extends ApplicationTest {

    @Override
    public void start(Stage primaryStage) throws Exception {
        FXMLLoader loader = new FXMLLoader(getClass().getResource("/path/to/Login.fxml"));
        primaryStage.setScene(new Scene(loader.load()));
        primaryStage.show();
    }

    @Test
    public void testLoginUI(FxRobot robot) {
        robot.clickOn("#usernameField").write("nombredeusuario");
        robot.clickOn("#passwordField").write("contraseña");
        robot.clickOn("#loginButton");

        assertEquals("Bienvenido, nombredeusuario", robot.lookup("#welcomeMessage").getText());
    }

    @Override
    public void stop() throws Exception {
        FxToolkit.hideStage();
    }
}
```

Nota Código fuente del script de interfaz UI

## Figura 4.6

Resultado prueba interfaz de usuario UI

```
Running LoginUITest
Nov 16, 2023 2:34:56 PM javafx.fxml.FXMLLoader$ValueElement processValue
WARNING: Loading FXML document with JavaFX API of version 16 by JavaFX runtime of ver
...
Nov 16, 2023 2:34:57 PM com.sun.javafx.stage.StageHelper lambda$runTask$2
WARNING: Unsupported JavaFX configuration: classes were loaded from 'unnamed module @
Nov 16, 2023 2:34:57 PM com.sun.javafx.stage.StageHelper lambda$runTask$2
WARNING: Unsupported JavaFX configuration: classes were loaded from 'unnamed module @
Login UI test completed successfully.

Tests run: 1, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 3.456 sec
```

Nota Resultado obtenido script de interfaz

### 4.5.4 Prueba de caja de blanca

Se vio necesario realizar pruebas de caja blanca al módulo de seguimiento ya que es un módulo que depende de los demás para funcionar y por este motivo cuenta con un alto flujograma de datos.

Estas pruebas de caja blanca son muy importantes para determinar la salud del sistema implementado, se realizaron métodos de prueba o monitoreo de software que analizan el diseño, el código y la estructura interna con el objetivo de mejorar las propiedades del sistema, como el uso seguro y eficiente.

Lista de nodos que formaran el grafo de flujo:

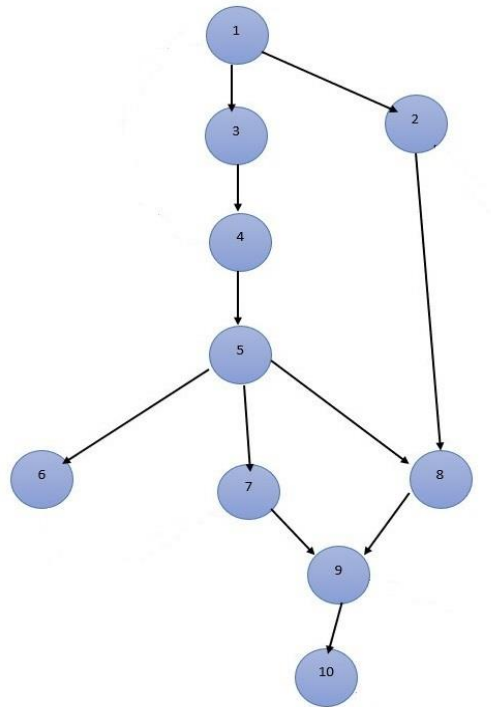
1. Inicio de servicio.
2. Registro de servicio.
3. Alta de servicio.
4. Modificación de servicio.
5. Cambio de partes.
6. Cambio de técnico.
7. Historial de servicio.
8. Asignar técnico.
9. Estado de servicio
10. Fin de sistemas

A continuación podemos observar el flujo grama de datos generado para los nodos ya mencionados



**Figura 4.7**

*Flujograma prueba de caja blanca*



Nota Nodos obtenidos para la prueba de caja blanca

Donde:

a = transiciones entre nodos,

n = número de nodos en el grafo.

Dado que en la prueba se tiene 10 nodos, la fórmula es:

$$V(G)=a-n+2$$

Remplazamos los valores:

$$V(G)=10-10+2=2$$

Entonces, según la complejidad ciclomática, se necesita realizar  $V(G)+1$  pruebas para cubrir todos los caminos independientes. En este caso, serían:

$$2+1=3 \text{ pruebas.}$$

#### 4.5.5 Prueba de caja negra

La evaluación base del sistema tomó en cuenta aspectos de funcionalidad, operación, aceptación de insumos, resultados, etc. La evaluación se realiza según los módulos para comparar los resultados entregados por el sistema o mediante informes o interfaces de resultados.

**Tabla 4.12**

Prueba de autenticación

.CN-01	Prueba de autenticación de usuarios
Propósito	Autenticar el inicio de sesión de un usuario por medio de roles asignados por el administrador del sistema.
Prerrequisitos	El usuario debe tener una cuenta en el sistema
Datos de entrada	Datos de usuario y contraseña registrados en la base de datos
Resultados esperados	El usuario accede correctamente al sistema, en caso de colocar credenciales no validas impedir el acceso.
Resultado obtenido	El usuario accede correctamente al sistema, y a sus módulos correspondientes
Resultado prueba	Correcto.

Nota Tabla de prueba de autenticación

**Tabla 4.13**

Prueba de registro de servicios

CN-2	Registro de servicios
------	-----------------------

Propósito	Registrar correctamente los servicios, equipos y otros datos para el correcto trabajo del personal.
Prerrequisito	El usuario debe ser un administrador o encargado del personal técnico
Datos de entrada	Se debe llenar los campos necesarios para iniciar el servicio
Resultados esperados	El sistema debe generar el servicio correctamente para el técnico encargado de dicho servicio.
Resultado obtenido	Los servicios son registrados correctamente
Resultado prueba	Correcto.

Nota Tabla de pruebas de registro de servicios

.

**Tabla 4.14**

Prueba de alta de servicios

.CN-3	Alta de servicios
Propósito	El técnico debe dar de alta los servicios que le corresponden.
Prerrequisito	El usuario debe ser un técnico
Datos de entrada	Para dar de alta un servicio es necesario llenar el motivo del alta del sistema.
Resultados esperados	El sistema da de alta el servicio y este pasa al historial, además de salir de dashboard

---

Resultado obtenido	El servicio pasa al historial con los datos solicitados
Resultado prueba	Correcto.

Nota Pruebas de altas para los servicios del sistema

#### 4.5.6 Prueba de estrés

Es una prueba que mide el comportamiento del sistema bajo un determinado requisito de conexión simultánea en el presente sistema utilizaremos un script.

Para realizar esta prueba se ha creado una tabla llamada TEST que vamos a utilizar en el la prueba para cargar datos, se cargaron datos simultáneamente para probar la cantidad de usuarios que podrían conectarse al sistema dándonos un resultado aproximado de 100 usuarios conectados al mismo tiempo el cual veremos a continuación.

#### Figura 4.8

*Prueba de estrés*

```
1. SQL_9iR2> CREATE TABLE test (id NUMBER);
2. SQL_9iR2> BEGIN
3.   FOR x IN 1 .. 100 LOOP
4.     INSERT INTO test VALUES (x);
5.   END LOOP;
6. END;
7. /

Procedimiento PL/SQL terminado correctamente.

Transcurrido: 00:00:01.25;
```

Nota Creación de una tabla de test

Ahora veremos el resultado del script para ejecutado sobre esta tabla.

Figura 4.9

Resultados prueba de estrés

```
1. TEST_ID TEST_DATE          COMMENTS
2. *****
3.      37 16/11/2023 08:46      100 conexiones concurrentes.
4.
5. Ingresá el TEST_ID: 37
6.
7. El archivo fue generado satisfactoriamente!
8.
9. -----
10. -----
11. ----- INFORMACION SOBRE EL STRESS_TEST -----
12. -----
13. -----
14. ID: 37 - FECHA: 16/11/2023 08:46 - COMENTARIOS: 100 conexiones concurrentes.
15. -----
16. -----
17. ----- RESULTADO INDIVIDUAL DEL STRESS_TEST (para cada una de las s
18. -----
19. -----
20. -----
21. SID: 248 - SPID: 11065
22. -----
23. -----
24. TOTAL_TIME_IN_SEC: 1.51 - TOTAL_TIME_IN_MIN: .03 - TOTAL_TIME_IN_HO: 0
25. -----
26. - EVENTOS -----
```

Nota Ejecución de script de test de concurrencia

Se concluyó que el presente sistema tiene un límite de 100 conexiones concurrentes.



# CAPÍTULO V

## CONCLUSIONES Y RECOMENDACIONES

## 5. CONCLUSIONES Y RECOMENDACIONES

### 5.1 Conclusiones

Después de haber desarrollado el SISTEMA PARA LA GESTIÓN DE CLIENTES MEDIANTE EL USO DE MICROSERVICIOS, A TRAVÉS DEL MODELO DE NEGOCIO CRM para la empresa E.P.S S.R.L.

Se llegó a las siguientes conclusiones:

- Se llegó a la conclusión de que las arquitecturas propuestas, así como las herramientas utilizadas fueron de total eficiencia para la culminación e implementación del presente proyecto.
- Aplicando el modelo de negocio CRM al sistema se pueden obtener datos precisos y relevantes sobre los clientes lo cual facilita la comunicación del gerente de ventas con dichos clientes.
- Se ha diseñado e implementado una base de datos en Oracle que permite ingresar y almacenar de manera rápida la información de cada trabajo realizado por los técnicos de la empresa. Esto ha ayudado a ahorrar tiempo a los empleados y mejorar los trabajos realizados.
- Durante la implementación del sistema se notó que la arquitectura de microservicios tuvo un gran impacto en la concurrencia de los datos agilizando todas las tareas de registro y edición.
- Con la implementación del módulo de servicios, se ha logrado mejorar el almacenamiento de la información de los clientes y sus equipos en espera de ser reparados, es posible ingresar y actualizar la información de manera eficiente.
- El módulo de seguimiento permite a los técnicos saber cuánto trabajo tienen pendiente, así como los datos del cliente fechas de entrega entre otros, lo cual permite que optimicen su trabajo.

- La implementación de la barra de búsqueda ha mejorado el acceso a la información requerida, es posible realizar búsquedas rápidas y precisas dentro del sistema.
- La generación de informes ejecutivos para los clientes ha mejorado la experiencia de atención hacia los clientes.
- La automatización y digitalización de los reportes e informes de los técnicos a optimizado el tiempo de trabajo de estos, con la eliminación de los procesos manuales las actividades diarias se agilizaron permitiendo un mejor aprovechamiento del tiempo.

En conclusión, el proyecto “SISTEMA PARA LA GESTIÓN DE CLIENTES MEDIANTE EL USO DE MICROSERVICIOS, A TRAVÉS DEL MODELO DE NEGOCIO CRM” ha cumplido con los objetivos planteados y ha brindado mejoras significativas en el trabajo diario de la empresa E.P.S S.R.L.

La implementación de los módulos adecuados, la optimización de los procesos de trabajo de los técnicos y la digitalización de las tareas han contribuido a mejorar la productividad de los empleados.

## **5.2 Recomendaciones**

- Se recomienda mantener la base de datos actualizada realizando un seguimiento constante de los trabajos realizados, esto garantizará la generación de informes precisos.
- Brindar una capacitación a todos los empleados nuevos en el uso del sistema, esto les permitirá aprovechar el sistema y realizar un mejor seguimiento de sus tareas.
- Realizar un monitoreo del desempeño del sistema y llevar a cabo mejoras en el sistema en caso de identificar alguna nueva tarea.
- Establecer medidas de seguridad adecuadas para proteger la información almacenada en la base de datos esto incluye el no permitir a personal no autorizado tener contacto con la base de datos.



- Evaluar la posibilidad de integrar el sistema con otras herramientas y sistemas utilizados en la empresa.
- Utilizar los datos del sistema para realizar análisis y mejorar los procesos empresariales.
- Mantenerse al día con las actualizaciones de todas las herramientas usadas en el desarrollo del presente sistema garantizando la compatibilidad y la eficiencia del sistema a largo plazo.

## Bibliografía

- Abenza, P. P. (2015). *Comenzando a programar con JAVA*. Universidad Miguel Hernández.
- Babelcube Inc. (2017). *Cómo construir Microservicios: Los diez principales trucos para modelar, integrar y desplegar microservicios*. Babelcube Inc.
- Cali, A. V. (2015). *Estrategias para el uso de un CRM*. IT Campus Academy.
- Carrion, J. (2017). *Diferencia entre dato información y conocimiento*.
- Chavez, J. (2020). ¿Qué es un Sistema informático? Componentes, características y ejemplos. *CEUPE Magazine*.
- Corcuera, P. (2017). En *Introducción a la tecnología Java*. Universidad de Cantabria.
- Corcuera, P. (2017). *Introducción a la tecnología Java*. Universidad de Cantabria.
- Debrauwer, L. &. (2016). *UML 2.5: iniciación, ejemplos y ejercicios corregidos*. Ediciones ENI.
- Dimes, T. (2015). *Conceptos Básicos de Scrum: Desarrollo de software Agile y manejo de proyectos Agile*. Babelcube Inc.
- Durango, A. (2015). *Diseño Web con CSS: 2ª Edición*. Campus Academy.
- Folgueiras Bertomeu, P. (2016). La entrevista.
- Gluon. (28 de Febrero de 2023). Scene Builder: <https://gluonhq.com/products/scene-builder/>
- Gómez, A. L. (2010). *Un modelo de estimacion de proyectos de software*.
- González, Y. D. (2012). Patrón Modelo-Vista-Controlador. *Telemática*, 47-57.
- Gracia, L. (7 de Febrero de 2022). *Modelos de estimación: un poco sobre COCOMO II*. <https://unpocodejava.com/2012/02/07/modelos-de-estimacion-un-poco-sobre-cocomo-ii/>
- Hans-Erik. (26 de Enero de 2023). *UML Toolkit*. <http://profesores.fi-b.unam.mx/carlos/aydoo/uml.html>
- Hernández, U. (2 de Febrero de 2023). <https://codigofacilito.com/articulos/mvc-model-view-controller-explicado>
- Hervella, J. F. (2015). *Java Struts Framework*. CI de Madrid, Ed.
- Heurtel, O. (2015). *Oracle 12C administracion*. Ediciones ENI.

Hiberus. (4 de Marzo de 2022). *Hiberus Blog*. <https://www.hiberus.com/crecemos-contigo/de-una-arquitectura-tradicional-a-microservicios/>

IBM. (2 de Diciembre de 2022). *Cifrado de contraseñas de ID de usuario de herramientas de servicio*. <https://www.ibm.com/docs/es/i/7.5?topic=ids-encryption-service-tools-user-id-passwords>

ISO. (2 de Febrero de 2023). *Iso standars*. ISO/IEC 27001:2022 Information security management systems: <https://www.iso.org/standard/27001>

ISO 25000. (4 de Enero de 2023). *Normas Iso 25000*. <https://iso25000.com/index.php/normas-iso-25000/iso-25040>

Jet Brains. (6 de Marzo de 2023). *IntelliJ IDEA – the Leading Java and Kotlin IDE*. <https://www.jetbrains.com/idea/>

LemonTech. (2 de Agosto de 2021). *Sistemas de seguimiento, control y monitoreo de proyectos legales*. <https://blog.lemontech.com/sistemas-de-seguimiento/>

López, D. &. (2017). *Arquitectura de Software basada en Microservicios para Desarrollo de Aplicaciones Web*.

Mesa, A. R. (19 de Diciembre de 2022). *Qué es un Sprint de Scrum*. <https://openwebinars.net/blog/que-es-un-sprint-scrum/>

Microsoft. (2 de Junio de 2023). <https://learn.microsoft.com/es-es/azure/architecture/guide/architecture-styles/microservices>

Mundo Creativo. (23 de Octubre de 2022). *Roles de usuario*. [https://mundo-creativo.fandom.com/es/wiki/2.1\\_Roles\\_de\\_usuario](https://mundo-creativo.fandom.com/es/wiki/2.1_Roles_de_usuario)

Oracle. (9 de Marzo de 2023). Oracle Database 23C: <https://www.oracle.com/es/database/>

Pajaro, G. (2020). *Protocolo individual 1 seguridad de software*. Universidad de Cartagena.

Pontis, S. (2011). Qué es el diseño de información. *Foro Alfa*.

Ramírez, M. R. (2019). Metodología SCRUM y desarrollo de Repositorio Digital. *Ibérica De Sistemas e Tecnologias De Informação*, 1062-1072.

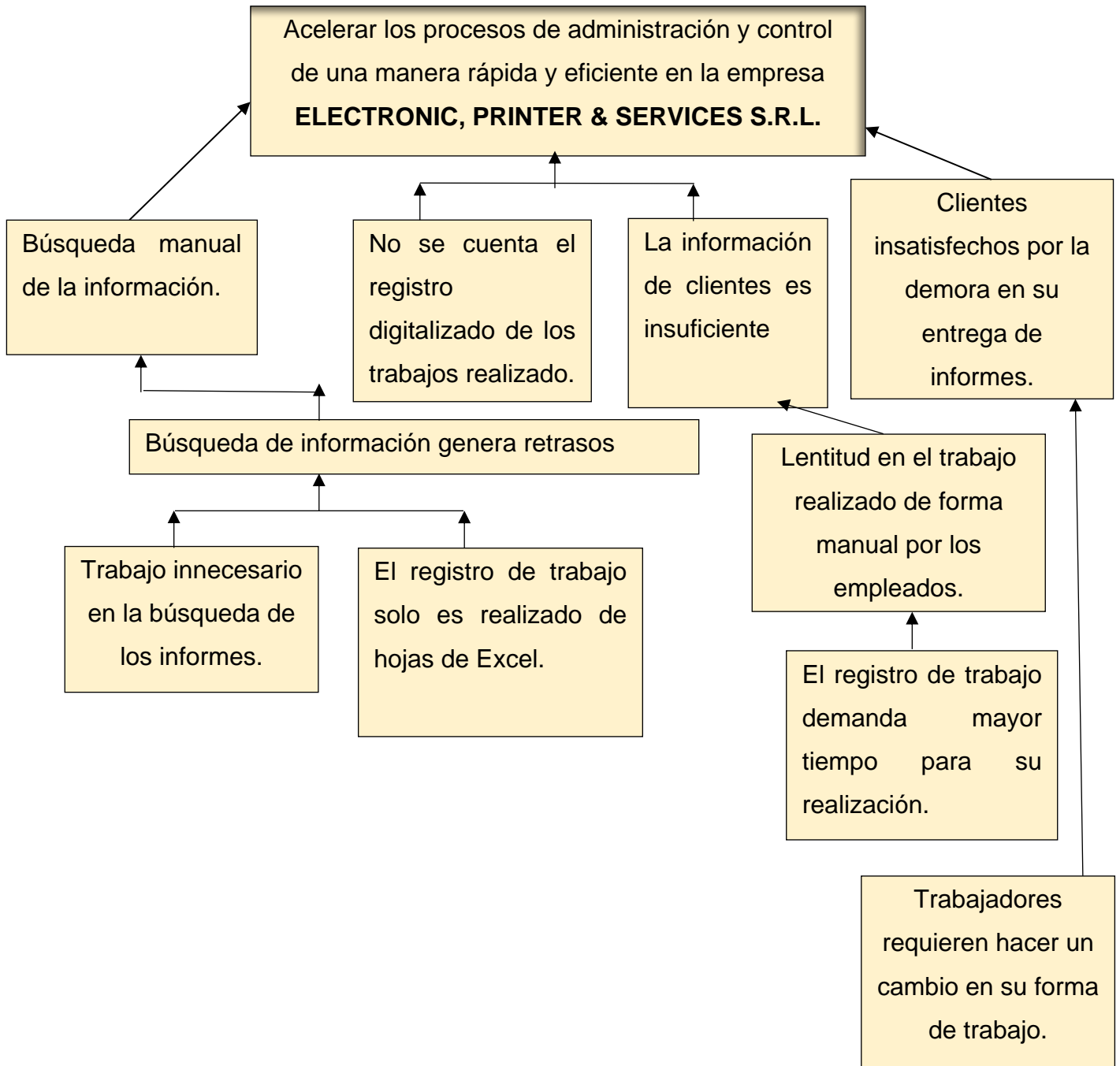
RD Station. (1 de junio de 2023). *RD Station*. <https://www.rdstation.com/blog/es/sistema->



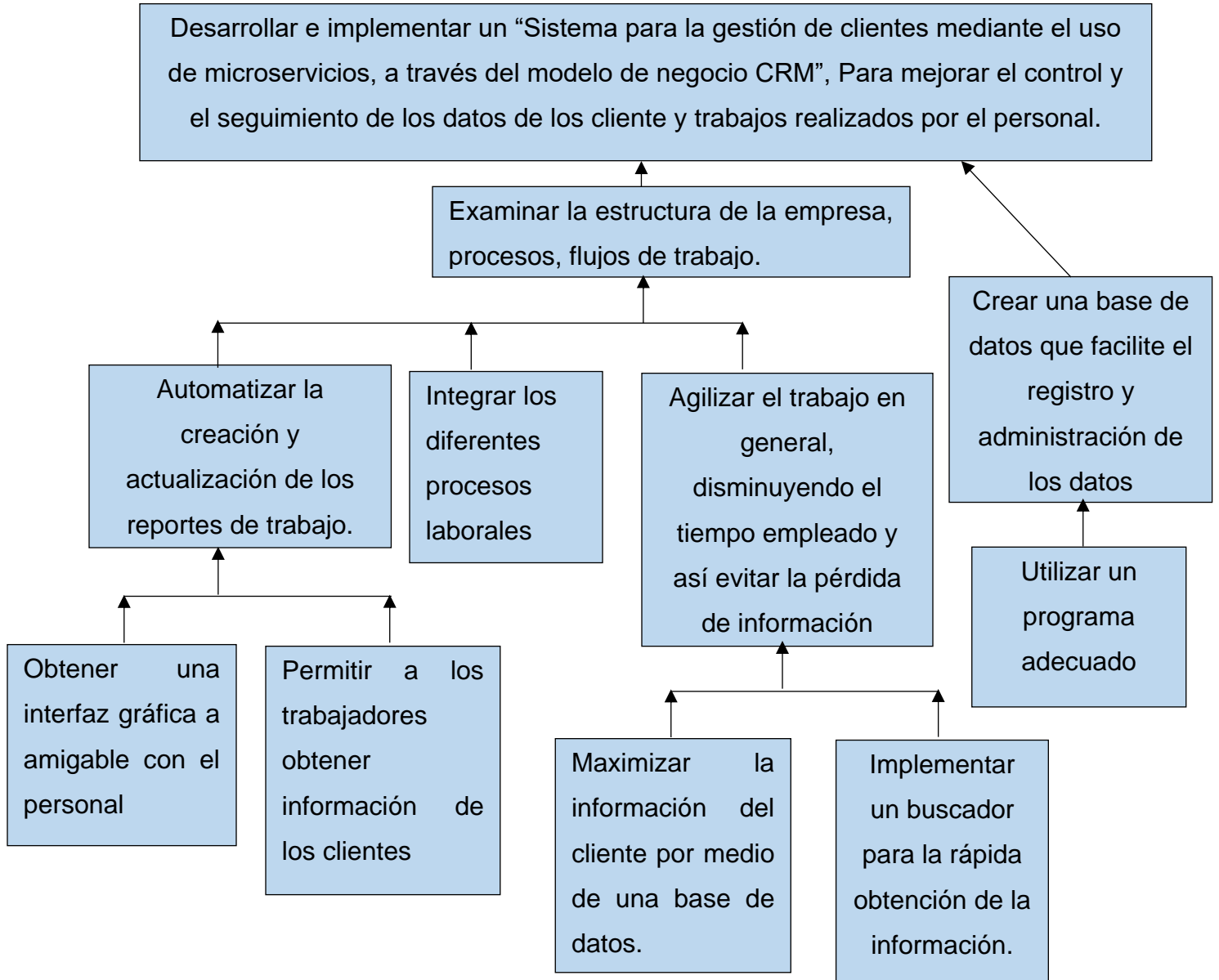


# ANEXOS

## ÁRBOL DE PROBLEMAS



## ÁRBOL DE OBJETIVOS









# ANEXO A DOCUMENTACION

El Alto, Noviembre 2023

Señor:  
Ing. William Roque Roque  
**DIRECTOR DE CARRERA**  
**INGENIERÍA DE SISTEMAS**  
Presente. –

## REF. AVAL DE CONFORMIDAD

Distinguido director de carrera:

Mediante la presente tengo a bien comunicarle mi conformidad del Trabajo de Grado:

**TITULO:** "SISTEMA PARA LA GESTIÓN DE CLIENTES MEDIANTE EL USO DE MICROSERVICIOS, A TRAVÉS DEL MODELO DE NEGOCIO CRM"

**CASO:** ELECTRONIC, PRINTER & SERVICES S.R.L.

**MODALIDAD:** PROYECTO DE GRADO

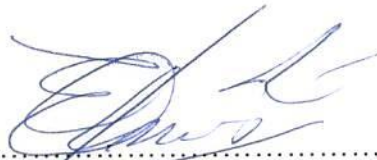
**UNIVERSITARIO:** RICARDO FRANCO SANCHEZ QUISPE

**REGISTRO UNIVERSITARIO:** 14007839

**CÉDULA DE IDENTIDAD:** 9204244 LP.

Para su defensa pública y evaluación correspondiente a la materia de Taller de Grado II, de acuerdo al reglamento vigente de la Carrera de Ingeniería de sistemas de la Universidad Pública de El Alto.

Atentamente,



.....  
M.Sc. Ing. Enrique Flores Baltazar  
TUTOR METODOLÓGICO  
TALLER DE GRADO II

El Alto, noviembre de 2023

Señor:  
M.Sc. Ing. Enrique Flores Baltazar  
**TUTOR METODOLÓGICO**  
**TALLER DE GRADO II**  
Presente. –

**REF. AVAL DE CONFORMIDAD**

Distinguido tutor metodológico:

Mediante la presente tengo a bien comunicarle mi conformidad del Trabajo de Grado:

**TITULO:** “SISTEMA PARA LA GESTIÓN DE CLIENTES MEDIANTE EL USO DE  
MICROSERVICIOS, A TRAVÉS DEL MODELO DE NEGOCIO CRM”  
**CASO:** ELECTRONIC, PRINTER & SERVICES S.R.L.  
**MODALIDAD:** PROYECTO DE GRADO  
**UNIVERSITARIO:** RICARDO FRANCO SANCHEZ QUISPE  
**REGISTRO UNIVERSITARIO:** 14007839  
**CÉDULA DE IDENTIDAD:** 9204244 LP.

Para su defensa pública y evaluación correspondiente a la materia de Taller de Grado II, de acuerdo al reglamento vigente de la Carrera de Ingeniería de Sistemas de la Universidad Pública de El Alto.

Atentamente.

  
.....  
Ing. Santos Chillo Espinoza  
**TUTOR REVISOR**

El Alto, noviembre de 2023

Señor:  
M.Sc. Ing. Enrique Flores Baltazar  
**TUTOR METODOLÓGICO**  
**TALLER DE GRADO II**  
Presente. –

**REF. AVAL DE CONFORMIDAD**

Distinguido tutor metodológico:

Mediante la presente tengo a bien comunicarle mi conformidad del Trabajo de Grado:

**TITULO:** “SISTEMA PARA LA GESTIÓN DE CLIENTES MEDIANTE EL USO DE MICROSERVICIOS, A TRAVÉS DEL MODELO DE NEGOCIO CRM”

**CASO:** ELECTRONIC, PRINTER & SERVICES S.R.L.

**MODALIDAD:** PROYECTO DE GRADO

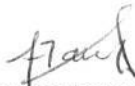
**UNIVERSITARIO:** RICARDO FRANCO SANCHEZ QUISPE

**REGISTRO UNIVERSITARIO:** 14007839

**CÉDULA DE IDENTIDAD:** 9204244 LP.

Para su defensa pública y evaluación correspondiente a la materia de Taller de Grado II, de acuerdo al reglamento vigente de la Carrera de Ingeniería de sistemas de la Universidad Pública de El Alto.

Atentamente.



.....  
Lic. Maria Magdalena Aguilar Guanto  
**TUTOR ESPECIALISTA**

La Paz, Noviembre de 2023

**Señor:**  
**M. SC. LIC. ING. ENRIQUE FLORES BALTAZAR**  
**TUTOR METODOLÓGICO**  
**TALLER DE MODALIDAD DE GRADUACIÓN**  
Presente. –

### REF. AVAL DE CONFORMIDAD

Distinguido tutor metodológico:

Mediante la presente tengo a bien de comunicarle la conformidad del trabajo de Grado:

**TÍTULO:** SISTEMA PARA LA GESTION DE CLIENTES MEDIANTE EL USO DE MICROSERVICIOS, A TRAVEZ DEL MODELO DE NEGOCIO CRM.

**CASO:** ELECTRONIC, PRINTER & SERVICES S.R.L.

**MODALIDAD:** PROYECTO DE GRADO

**ESTUDIANTE:** RICARDO FRANCO SANCHEZ QUISPE

**CÉDULA DE IDENTIDAD:** 9204244 L.P.

De tal forma cabe recalcar que el **SISTEMA** satisface los requerimientos de la institución, de esta forma se dio cumplimiento de los objetos del presente proyecto.

El **SISTEMA** fue **IMPLEMENTADO** satisfactoriamente en la institución.

En cuanto certifico, en honor a la verdad para fines consiguientes de interesado para su defensa pública y evaluación correspondiente ala materia de Taller de Grado II, de acuerdo al reglamento vigente de la Carrera de Ingeniería de Sistemas de la universidad Pública de El Alto

Atentamente.



Victor J. Sanchez C.  
REPRESENTANTE LEGAL  
 **Electronic Printer  
& Services S.R.L.**





**ANEXO B**  
**MANUAL DE USUARIO**

# MANUAL DE USUARIO

**“SISTEMA PARA LA GESTIÓN DE CLIENTES MEDIANTE EL USO  
DE MICROSERVICIOS, A TRAVÉS DEL MODELO DE NEGOCIO  
CRM”**



La Paz - Bolivia

2023

## Tabla de contenido

1 Ingreso al Sistema.....	1
2 Seguridad del sistema .....	1
3 Salir del Sistema .....	1
4 Desconectarse del Sistema.....	1
5 Opciones .....	2
6 Opciones Principales de Modulo .....	2
7 Barra lateral del modulo .....	2
8 Detalle de Botones .....	3
9 Columnas .....	3
10 Consulta de Datos.....	3
11 Detalle de Pantallas de alertas.....	4
12 Módulo de usuarios .....	5
13 Cambio de Contraseña .....	5
14 Resetear Contraseña .....	5
15 Administración de Usuarios.....	6
16 Bloqueo de Usuarios .....	6
17 Desbloquear Usuarios.....	6
18 Roles .....	7
19 Administración de Roles.....	7
20 Asignación de Roles a Usuarios.....	8
21 Clientes .....	8
22 Seguimiento de servicios.....	9
23 Reportes.....	9
24 Dashboard.....	10



## Tabla de figuras

Figura 1 Pantalla de inicio del sistema.....	1
Figura 2 Menú del sistema.....	2
Figura 3 Barra lateral.....	2
Figura 4 Columna de datos.....	3
Figura 5 Barra de búsqueda.....	3
Figura 6 Alerta correcto.....	4
Figura 7 Alerta de advertencia.....	4
Figura 8 Alerta de problema.....	4
Figura 9 Lista de errores.....	4
Figura 10 Ventana de cambio de contraseña.....	5
Figura 11 Ventana de reseteo de contraseña.....	5
Figura 12 Ventana de nuevos usuarios.....	6
Figura 13 Ventana bloqueo de usuarios.....	6
Figura 14 Ventana de roles.....	7
Figura 15 Ventana administración de roles.....	7
Figura 16 Ventana registro de nuevos clientes.....	8
Figura 17 Ventana de servicios.....	8
Figura 18 Ventana de reportes.....	9
Figura 19 Ventana de dashboard.....	10
Figura 20 Servicios pendientes.....	10

## Manual de Usuario

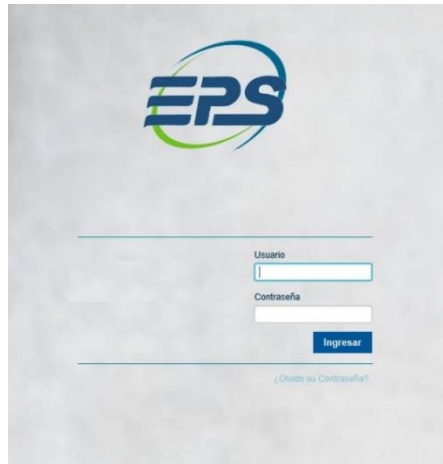
### 1 Ingreso al Sistema.

Para Ingresar al sistema se debe ingresar los datos de Usuario y Contraseña asignados al momento de alta de su usuario en el sistema y hacer clic en el botón de

Ingresar .

#### Figura 1

Pantalla de inicio del sistema



Pantalla de acceso al sistema.

### 2 Seguridad del sistema

La contraseña de acceso debe contener caracteres alfabéticos, números y caracteres especiales.

### 3 Salir del Sistema

Para salir del sistema encontrara el icono  en la parte superior derecha del sistema.

### 4 Desconectarse del Sistema

Para desconectarse del sistema el usuario debe hacer clic en



## 5 Opciones



Atrás, nos lleva a la anterior opción de menú del módulo seleccionado.



Botón Home, nos lleva al menú de Módulos del sistema.



Botón salir, desconecta al usuario del sistema.

## 6 Opciones Principales de Modulo

Una vez seleccionado un módulo, el sistema despliega las opciones principales del módulo.

### Figura 2

Menú del sistema



Módulos del sistema

## 7 Barra lateral del modulo

Después de seleccionar una opción de la pantalla de Opciones Principales, el sistema despliega una lista de opciones secundarias a la izquierda de la pantalla. Para seleccionar una opción se debe hacer clic en la misma.

### Figura 3

Barra lateral



Opciones del menú lateral

## 8 Detalle de Botones

Registro de equipo: Registra un nuevo equipo.

Reporte de equipo: Saca un listado de los equipos.

Registro Marcas de Equipos: Registra las marcas de los equipos

Registro Modelos de equipos: Registra modelos de equipos

Registro tipos de equipo: Registra el tipo de equipo (computadora, impresora, etc.).

## 9 Columnas

Para ordenar cualquier tabla se debe hacer clic en la columna a ordenar.

**Figura 4**

Columna de datos

ID	Formulario	Fe Servicio	Fe Estimada	Tecnico
23	9938555	2021-01-01	2021-01-01	JOSE SANCHEZ
22	66969	2021-01-01	2021-01-01	VICTOR SANCHEZ CALLIZ
21	55665	2021-03-01	2021-03-01	JOSE SANCHEZ
20	669988	2021-03-01	2021-03-01	JAIME TOLA
19	33333	2021-03-01	2021-03-01	JOSE SANCHEZ
18	2222	2021-03-01	2021-03-01	JOSE SANCHEZ
17	FOR 2222	2021-01-01	2021-01-01	JAIME TOLA
16	FOR / 1555	2021-03-01	2021-03-01	JAIME TOLA

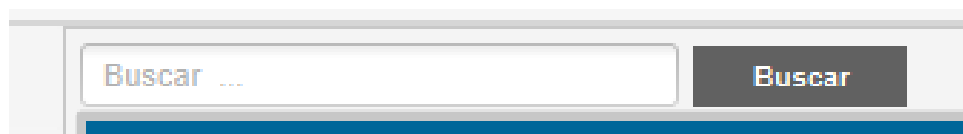
Columnas

## 10 Consulta de Datos

Para realizar búsquedas en las tablas se debe introducir el dato buscado en el buscador (este buscara por cualquier columna que la tabla muestre) y presionar la tecla Intro

**Figura 5**

Barra de búsqueda



Una barra de búsqueda con un campo de texto que contiene el texto "Buscar ..." y un botón "Buscar" a la derecha.

Buscar datos

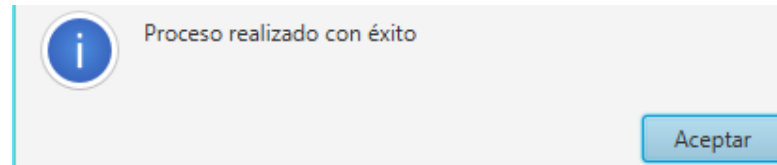
## 11 Detalle de Pantallas de alertas

En el sistema se despliegan diferentes pantallas de con mensajes de errores, alertas e Información.

Indica que el proceso se ejecutó sin problemas

### Figura 6

Alerta de éxito

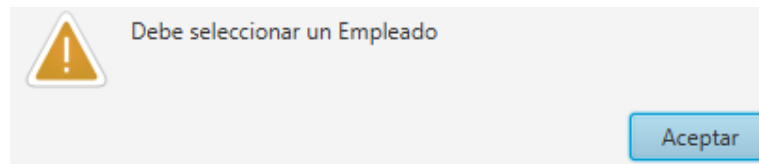


Alerta de éxito

Mensaje de alerta: Esta pantalla se presenta para indicar que existe falta de información, completar datos, etc.

### Figura 7

Alerta de advertencia

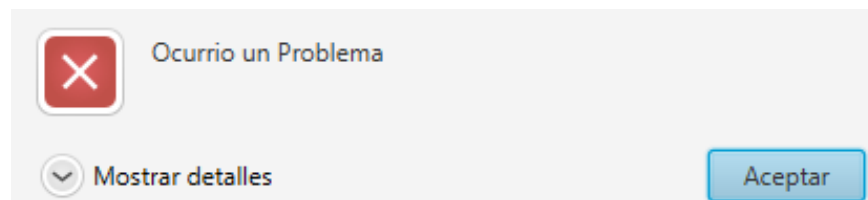


Alerta advertencia

Indica que existe algún error en el proceso.

### Figura 8

Alerta de problema

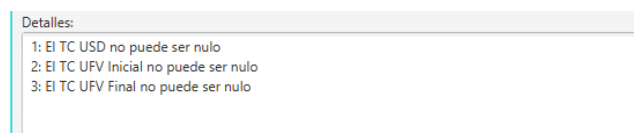


Alerta de Error

Existen otro tipo de errores donde nos muestra el error

### Figura 9

Lista de errores



Listado de errores

## 12 Módulo de usuarios

Para ingresar al módulo de Usuarios se debe hacer clic en el icono de usuarios.

## 13 Cambio de Contraseña

Esta opción nos permite realizar el cambio de contraseña en el sistema

### Figura 10

Ventana de cambio de contraseña



The screenshot shows a web form titled "Cambio de Contraseña del Usuario". It contains three input fields: "Contraseña Anterior", "Nueva Contraseña", and "Confirme nueva Contraseña". A "Guardar" button is located at the bottom right of the form.

Cambio de Contraseña

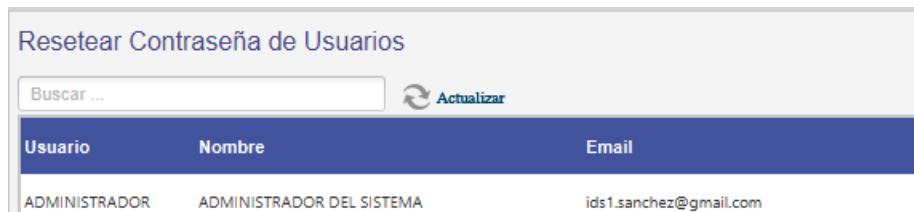
Debe tomar en cuenta que la contraseña debe tener al menos 6 caracteres los cuales deben contener letras (mayúsculas y minúsculas), números y caracteres especiales (como ser |, \$, #, etc.)

## 14 Resetear Contraseña

Esta opción sirve para reestablecer la contraseña de un usuario. Se debe seleccionar el usuario y hacer clic en el botón de guardar.

### Figura 11

Ventana de reseteo de contraseña



The screenshot shows a web form titled "Resetear Contraseña de Usuarios". It features a search input field labeled "Buscar ..." and an "Actualizar" button. Below the search field is a table with the following data:

Usuario	Nombre	Email
ADMINISTRADOR	ADMINISTRADOR DEL SISTEMA	ids1.sanchez@gmail.com

Reseteo de contraseña

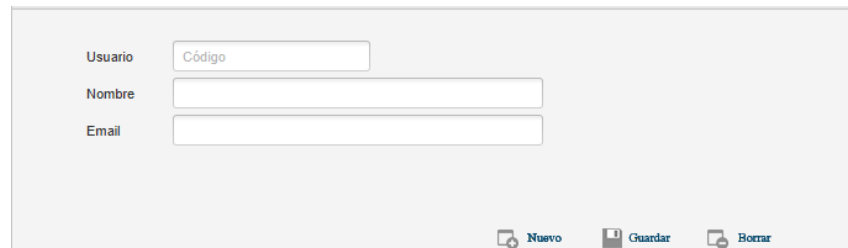
El sistema cuenta con dos opciones para la generación de la nueva contraseña:

## 15 Administración de Usuarios

Mediante esta opción podemos dar de alta, baja y modificación de datos de los usuarios en el sistema

### Figura 12

Ventana de nuevos usuarios



Formulario para registrar nuevos usuarios. Campos: Usuario (Código), Nombre, Email. Botones: Nuevo, Guardar, Borrar.

Registro de nuevos usuarios del Sistema

## 16 Bloqueo de Usuarios

Por medio de esta opción se realiza el bloqueo definitivo de ingreso al sistema a los usuarios que ya no se encuentran en la empresa.

### Figura 13

Ventana bloqueo de usuarios



Usuario	Nombre	Email
ADMINISTRADOR	ADMINISTRADOR DEL SISTEMA	ids1.sanchez@gmail.com

Bloqueo de usuarios

## 17 Desbloquear Usuarios

Esta opción permite habilitar nuevamente a los usuarios para ingresar al sistema

## 18 Roles

Esta opción nos permite crear roles de acceso al sistema, es decir perfiles con los cuales se puede restringir el acceso de los usuarios a las opciones de menús, sub-menús, etc.

### Figura 14

Ventana de roles

Roles	Descripción
1000	SUPER USUARIO
1001	SUPERVISOR
1002	AUXILIAR

Rol: Generacion Automática  
Descripción:

Administración de roles del sistema

## 19 Administración de Roles

Esta opción nos permite realizar la asignación de Roles a las Opciones de Menú, es muy importante considerar que esta asignación al ROL permitirá el acceso a las opciones de menú.

### Figura 15

Ventana administración de roles


Modulo: ADMINISTRACIÓN DEL CAPITAL HUMANO  
Rol: AUXILIAR

Opciones de Menú	Asignar al Rol
Mensajería	<input type="checkbox"/>
Registro de Operaciones	<input type="checkbox"/>
Reporte de Operaciones	<input type="checkbox"/>
Programación de Operaciones	<input type="checkbox"/>
Reversión de Planillas	<input type="checkbox"/>
Cierre Definitivo de Planillas	<input type="checkbox"/>

Opciones de Menú Asignadas al Rol	Quitar del Rol
Menú de RRHH	<input type="checkbox"/>
Registro de Empleados	<input type="checkbox"/>
Operaciones	<input type="checkbox"/>
Registro de Procesos	<input type="checkbox"/>
Registro Form 110	<input type="checkbox"/>
Generación de Planillas	<input type="checkbox"/>

Administración de Roles



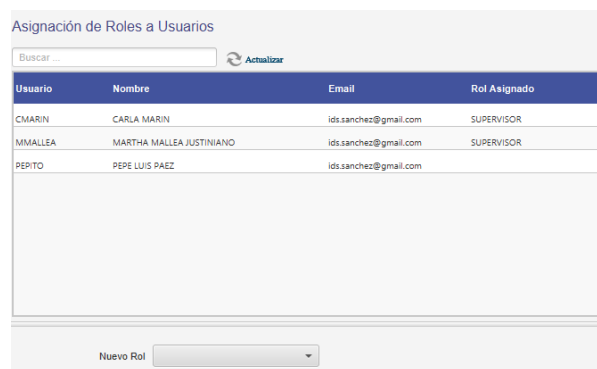
Para asignar y/o eliminar las opciones del menú, seleccione la opción y presione o haga clic el botón de guardar 

## 20 Asignación de Roles a Usuarios

Esta opción nos permite la Asignación y Cambio de Roles a Usuarios del sistema. Para realizar el proceso de asignación, primero debe seleccionar a un usuario y posteriormente asignar el rol correspondiente.

### Figura 16

Ventana asignación de roles



Usuario	Nombre	Email	Rol Asignado
CMARIN	CARLA MARIN	ids.sanchez@gmail.com	SUPERVISOR
MMALLEA	MARTHA MALLEA JUSTINIANO	ids.sanchez@gmail.com	SUPERVISOR
PEPI TO	PEPE LUIS PAEZ	ids.sanchez@gmail.com	

Asignación de Roles a Usuarios del Sistema

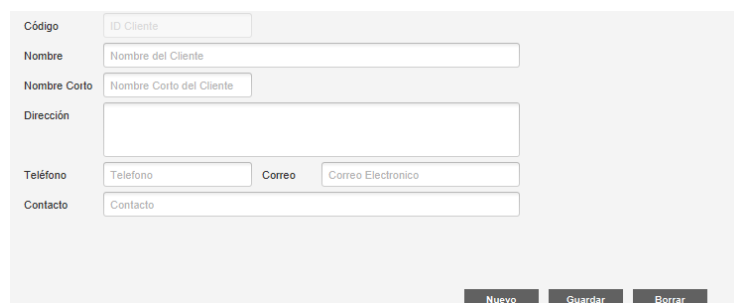
Para realizar el cambio de rol, debe seleccionar el usuario, después seleccionar el nuevo rol a ser asignado y guardar los cambios.

## 21 Clientes

En este módulo se puede registrar, editar y eliminar los clientes de la empresa, estos clientes pueden tener alguna sucursal

### Figura 16

Ventana registro de nuevos clientes



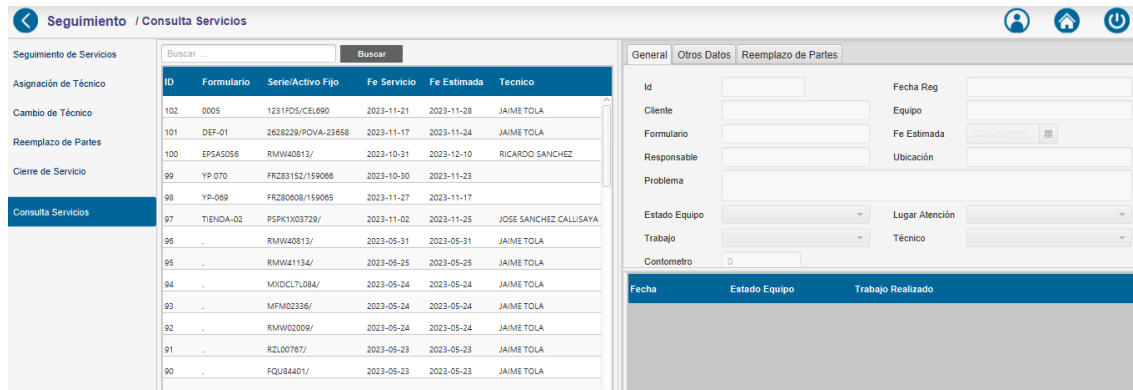
Formulario módulo de clientes

## 22 Seguimiento de servicios

En este módulo podemos realizar el seguimiento de los servicios previamente registrados, este módulo mostrara los servicios relacionados a su usuario.

Figura 17

Ventana de servicios



The screenshot shows a web application interface for service tracking. On the left, there is a sidebar with navigation options: 'Seguimiento de Servicios', 'Asignación de Técnico', 'Cambio de Técnico', 'Reemplazo de Partes', 'Cierre de Servicio', and 'Consulta Servicios'. The main area displays a table of services with columns for ID, Formulario, Serie/Activo Fijo, Fe Servicio, Fe Estimada, and Técnico. The table contains 18 rows of data. On the right, there is a detailed view for a selected service, including fields for Id, Cliente, Formulario, Responsable, Problema, Estado Equipo, Trabajo, and Contometro. Below these fields is a table with columns for Fecha, Estado Equipo, and Trabajo Realizado.

ID	Formulario	Serie/Activo Fijo	Fe Servicio	Fe Estimada	Técnico
102	0005	1231FDS/CEL690	2023-11-21	2023-11-28	JAIME TOLA
101	DEF-01	2628229/POVA-22658	2023-11-17	2023-11-24	JAIME TOLA
100	EPSA5056	RMW40813/	2023-10-31	2023-12-10	RICARDO SANCHEZ
99	YP 070	FRZ83152/159066	2023-10-30	2023-11-23	
98	YP-069	FRZ80608/159065	2023-11-27	2023-11-17	
97	TIENDA-02	PSPK1X03729/	2023-11-02	2023-11-25	JOSE SANCHEZ CALLISAYA
96	.	RMW40813/	2023-05-31	2023-05-31	JAIME TOLA
95	.	RMW41134/	2023-05-25	2023-05-25	JAIME TOLA
94	.	MXDCL7,084/	2023-05-24	2023-05-24	JAIME TOLA
93	.	MPM02336/	2023-05-24	2023-05-24	JAIME TOLA
92	.	RMW02009/	2023-05-24	2023-05-24	JAIME TOLA
91	.	RZL00797/	2023-05-23	2023-05-23	JAIME TOLA
90	.	FQU84401/	2023-05-23	2023-05-23	JAIME TOLA
89	.	MJ706633/	2023-06-21	2023-06-21	JAIME TOLA

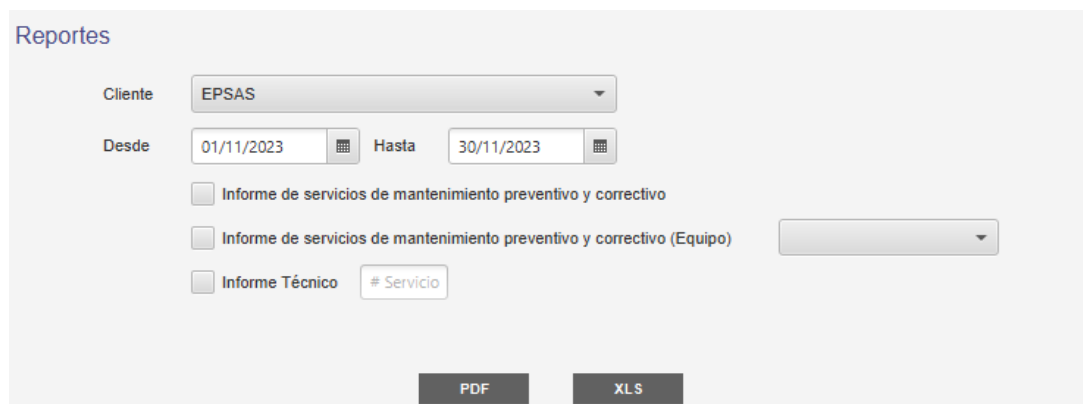
Panel de servicios

## 23 Reportes

En el módulo de reportes podemos generar los reportes necesarios, primero se debe seleccionar un cliente de la lista, luego seleccionar el rango de fechas requerido para el reporte para finalmente tachar el tipo de reporte que necesita.

Figura 18

Ventana de reportes



The screenshot shows a web application interface for generating reports. The title is 'Reportes'. There are three main sections: 'Cliente' with a dropdown menu showing 'EPSAS'; 'Desde' and 'Hasta' with date pickers showing '01/11/2023' and '30/11/2023' respectively; and 'Informe de servicios de mantenimiento preventivo y correctivo' with a dropdown menu. Below these are two radio buttons: 'Informe de servicios de mantenimiento preventivo y correctivo (Equipo)' and 'Informe Técnico' with a text input field containing '# Servicio'. At the bottom, there are two buttons: 'PDF' and 'XLS'.

Generando reportes

## 24 Dashboard

En la pantalla del dashboard podemos observar los trabajos pendientes, veremos campos como el técnico asignado al servicio, la fecha de finalización el lugar y tipo de servicio que se debe realizar.

**Figura 19**

Ventana de dashboard

Servicios Pendientes											
Y.P.F.B	FOTOCOPIADORA	CANON	IR 2525	Edificio corporativo			NO SACA COPIAS	27/11/202	17/11/202	●	
								3	3		
Y.P.F.B	FOTOCOPIADORA	CANON	IR 2525	Edificio corporativo			BANDEJA DE FOTOCOPIADORA NO JALA PAPEL	30/10/202	23/11/202	●	
								3	3		
MIN_DEF	CELULAR	TECNO	POVA 2	CENTRAL	DEFENSA	1	JAIME TOLA	NO ENCIENDE	17/11/202	24/11/202	●
									3	3	
TIENDA EPS	DATASHOW	EPSON	POWERLITE S12 PLUS	OFICINA EPS			JOSE SANCHEZ CALLISAYA	FOCO DAÑADO TIENE UNA MANCHA AMARILLA	02/11/202	25/11/202	●
									3	3	
MIN AGUA	CELULAR	TECNO	POVA 5	central	DESARROLLO	2	JAIME TOLA	NO ENCIENDE	21/11/202	28/11/202	●
									3	3	

Ventana de Dashboard

Así mismo se tiene 3 colores para los servicios

**Figura 20**

Servicios pendientes

30/10/202	23/11/202	●
3	3	
17/11/202	24/11/202	●
3	3	
02/11/202	25/11/202	●
3	3	
21/11/202	28/11/202	●
3	3	

Alertas del dashboard

- El color rojo nos indica que el servicio está llegando a su fecha de entrega
- El color amarillo nos indica que el servicio acabara en una semana o menos
- El color verde nos indica que se está a tiempo para acabar el servicio



**ANEXO C**  
**MANUAL TÉCNICO**

# MANUAL TÉCNICO

**“SISTEMA PARA LA GESTIÓN DE CLIENTES MEDIANTE EL USO  
DE MICROSERVICIOS, A TRAVÉS DEL MODELO DE NEGOCIO  
CRM”**



La Paz - Bolivia

2023

## Tabla de contenido

1	Requerimientos técnicos .....	1
1.1	Requisitos mínimos: .....	1
2	Instalación de programas .....	1
2.1	Oracle.....	1
2.2	JDK java .....	6
3	Instalando el sistema.....	7
3.2	Copia del sistema.....	7
3.2	Archivos de configuración .....	7
3.3	Creando acceso directo .....	8

## Tabla de ilustraciones

Ilustración 1 Descargando Oracle .....	1
Ilustración 2 Instalación de Oracle .....	2
Ilustración 3 Términos de Oracle.....	2
Ilustración 4 Ruta de instalación de Oracle .....	3
Ilustración 5 Credenciales de Oracle.....	3
Ilustración 6 Ventana de instalación de Oracle .....	4
Ilustración 7 Acceso a firewall .....	4
Ilustración 8 Ejecutando Oracle.....	5
Ilustración 9 Backup de la base de datos .....	5
Ilustración 10 Descargando JDK.....	6
Ilustración 11 Ejecutando instalador.....	6
Ilustración 12 Instalando JDK.....	7
Ilustración 13 Carpeta del sistema .....	7
Ilustración 14 Archivos de configuración .....	7
Ilustración 15 Ejecutable del sistema .....	8
Ilustración 16 Acceso directo al sistema .....	8
Ilustración 17 Pantalla de Inicio de sesión .....	8

# 1 Requerimientos técnicos

Para el correcto funcionamiento del sistema se recomienda tener las siguientes características en el equipo que actuara como servidor principal:

## 1.1 Requisitos mínimos:

- Memoria ram de 8 GB
- Disco duro de 250 GB
- Windows 8 o superior
- Procesador Intel Core I3

# 2 Instalación de programas

## 2.1 Oracle

Primeramente, nos descargamos el Oracle de la página:

<https://www.oracle.com/es/database/technologies/>

### Ilustración 1

Descargando Oracle



Pantalla de descarga de JDK



Una vez descargado instalamos mediante el setup del programa.

### **Ilustración 2**

*Instalación de Oracle*

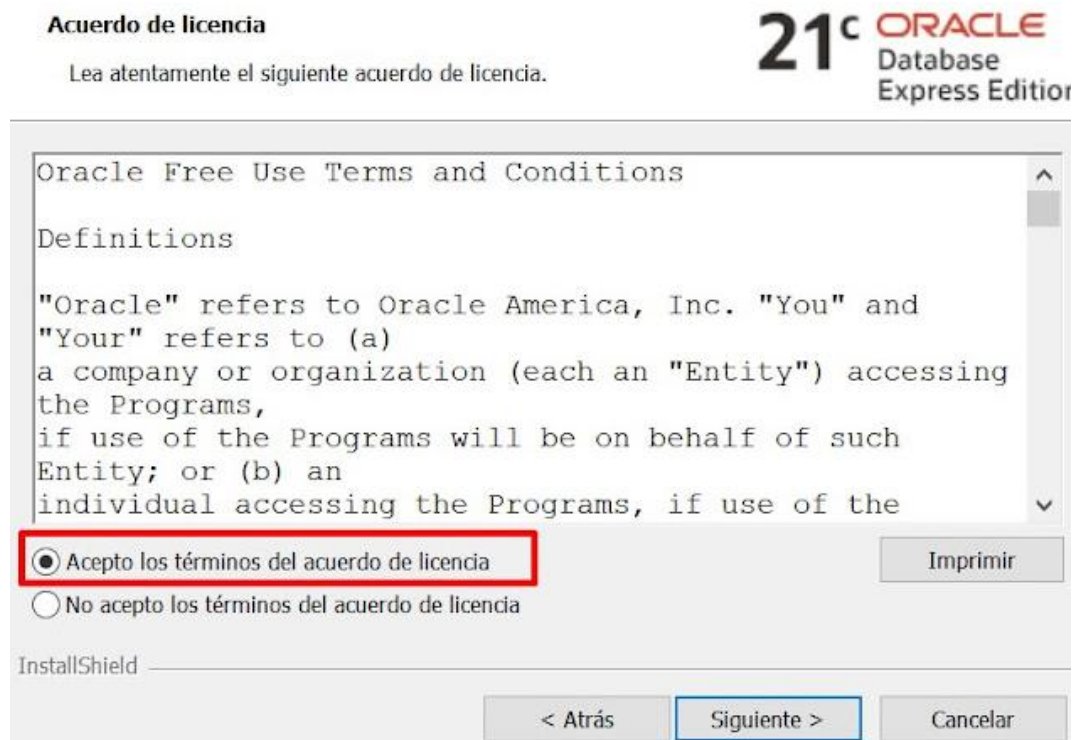


Instalador de Oracle

Aceptamos los términos y condiciones del programa

### **Ilustración 3**

*Términos de Oracle*

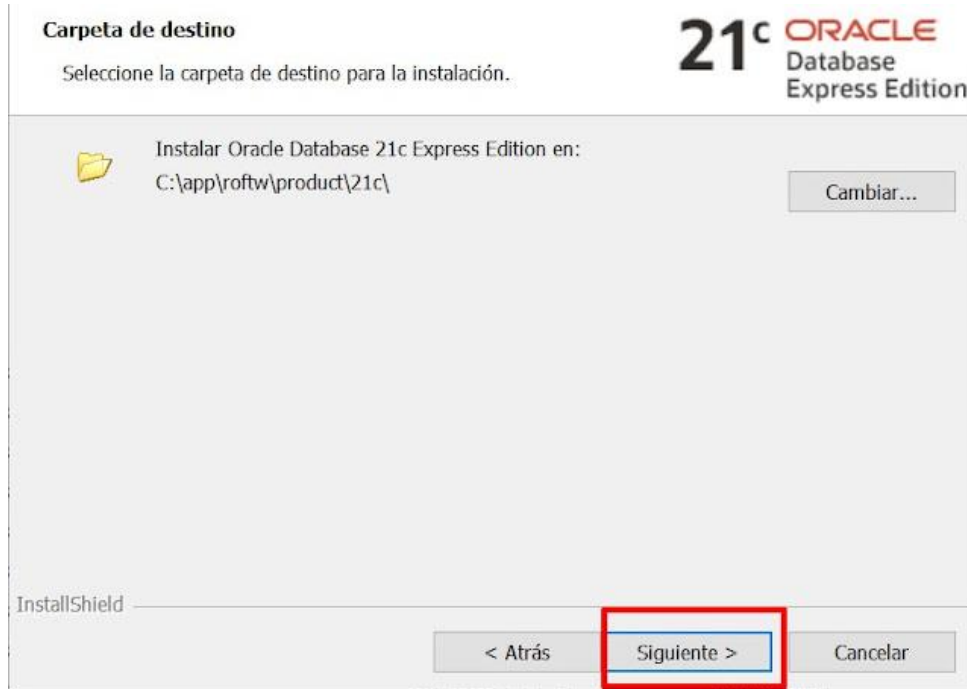


Términos y condiciones

Elegimos la ruta de instalación

#### Ilustración 4

Ruta de instalación de Oracle

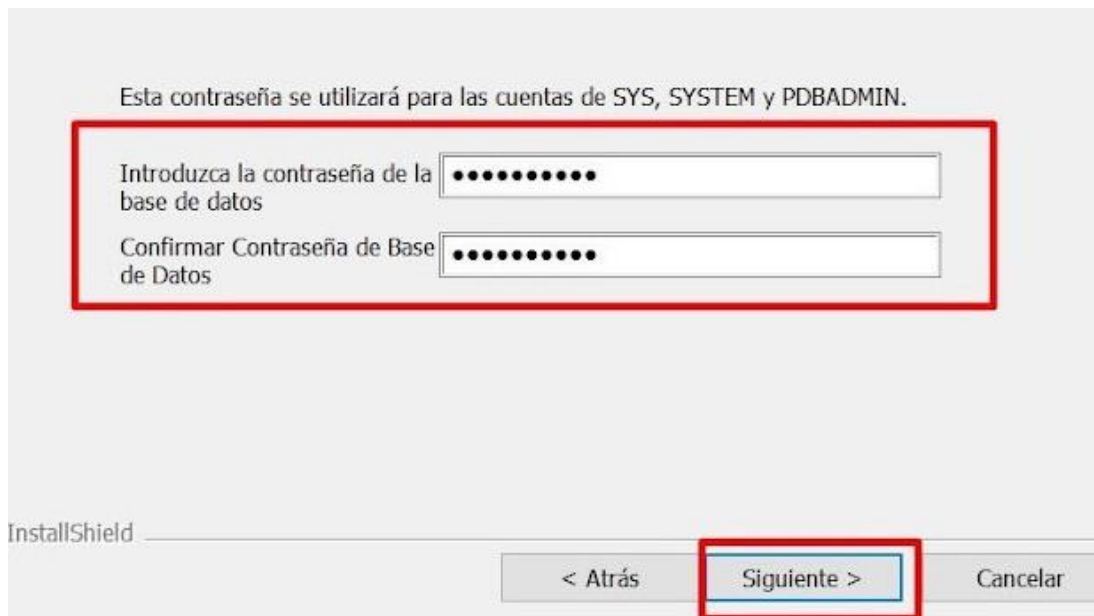


Instalado el Oracle

Creamos una contraseña y un usuario

#### Ilustración 5

Credenciales de Oracle

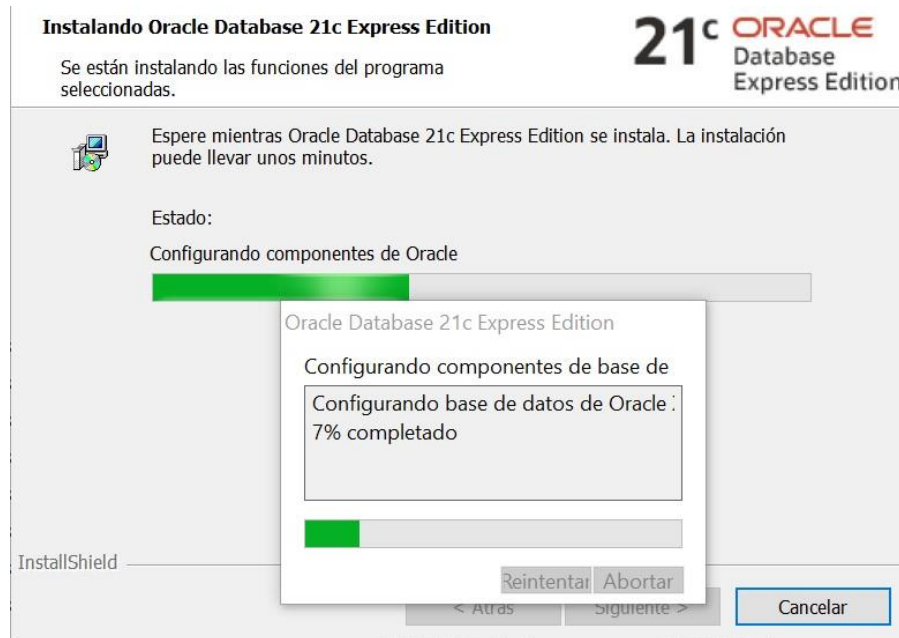


Credenciales Oracle

Esperamos a que termine la instalación

### Ilustración 6

Ventana de instalación de Oracle

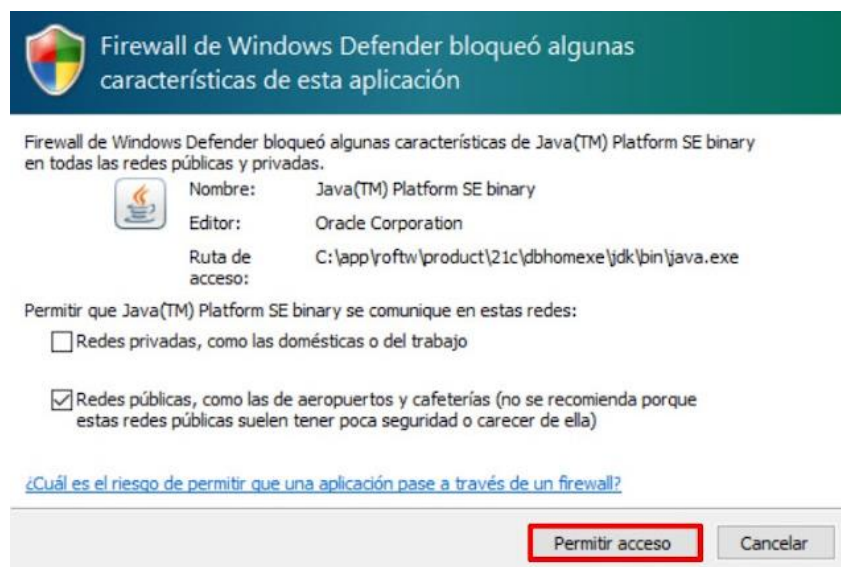


Instalando el programa

### Ilustración 7

Acceso a firewall

Si nos sale una ventana de alerta de Windows le permitimos el acceso

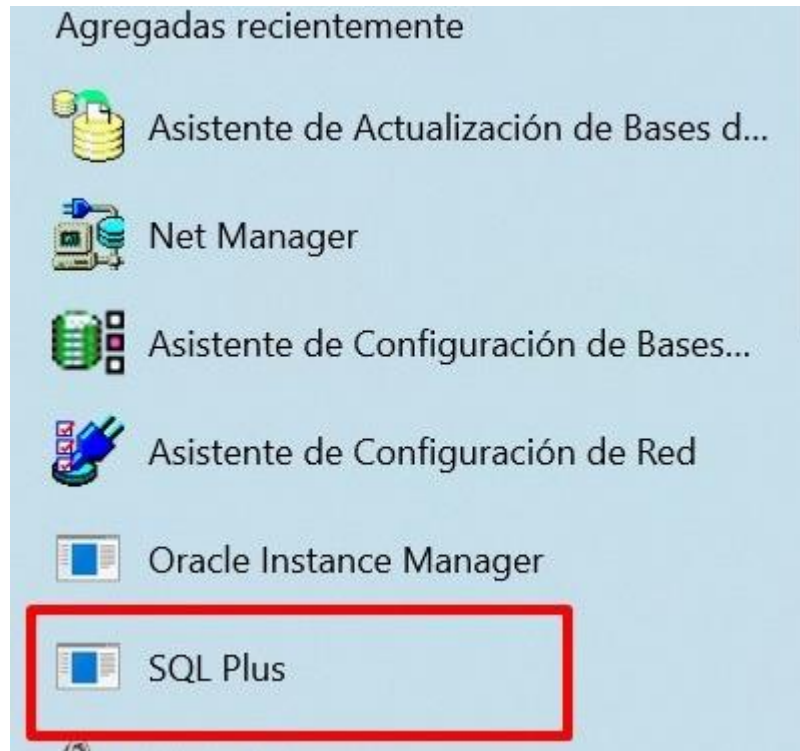


Firewall de Windows

Posteriormente entramos al programa y realizamos la importación de la base de datos

**Ilustración 8**

*Ejecutando Oracle*

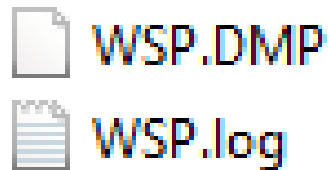


Ejecutable de la base de datos

La base de datos se encuentra en la carpeta DB

**Ilustración 9**

*Backup de la base de datos*



Archivos de la base de datos a importar

## 2.2 JDK java

Posteriormente nos vamos a la página:

<https://www.oracle.com/java/technologies/downloads/>

Descargamos el JDK que contiene lo último en java para que pueda correr el sistema.

### **Ilustración 10**

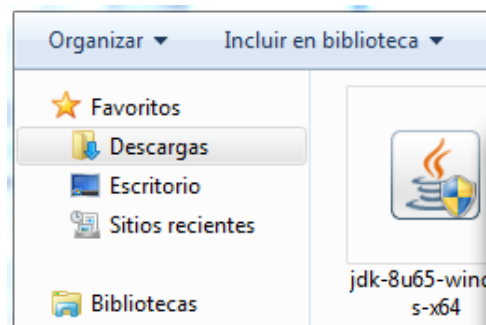
*Descargando JDK*



Página de descarga de java  
Ejecutamos el instalador

### **Ilustración 11**

*Ejecutando instalador*

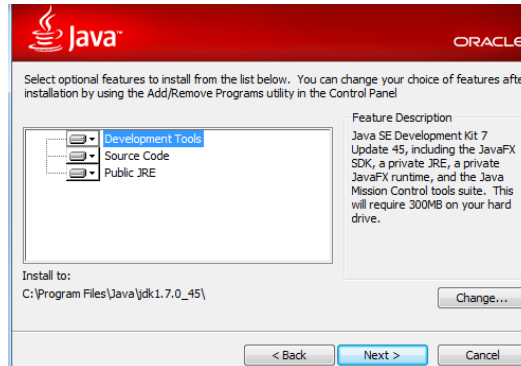


Instalador de java

Apretamos en Next y esperamos que termine la descarga

## Ilustración 12

Instalando JDK



Instalación de java

## 3 Instalando el sistema

### 3.2 Copia del sistema

El sistema estará almacenado en una carpeta denominada EPS (Electronic, Printer & Services)

## Ilustración 13

Carpeta del sistema

Archivos de programa (x86)	14/11/2023 20:02	Carpeta de archivos
Boot	13/01/2022 1:16	Carpeta de archivos
EPS	16/11/2023 20:48	Carpeta de archivos

Carpeta contenedora del sistema

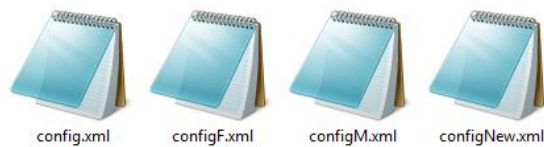
Debemos copiar esta carpeta al disco local "C"

### 3.2 Archivos de configuración

El siguiente paso es copiar los archivos de configuración a la carpeta "config"

## Ilustración 14

Archivos de configuración





Archivos de configuración de la base de datos

### 3.3 Creando acceso directo

En la carpeta principal tenemos el archivo EPS.jar que es el ejecutable del sistema, se recomienda crear un acceso directo el cual moveremos al escritorio

#### **Ilustración 15**

*Ejecutable del sistema*

 EPS.jar	16/11/2023 20:23	Executable Jar File	6.380 KB
 EPS	12/07/2023 16:35	Acceso directo	2 KB

Ejecutable del sistema

Ejecutamos el archivo jar e iniciamos sesión

#### **Ilustración 16** Acceso directo al sistema

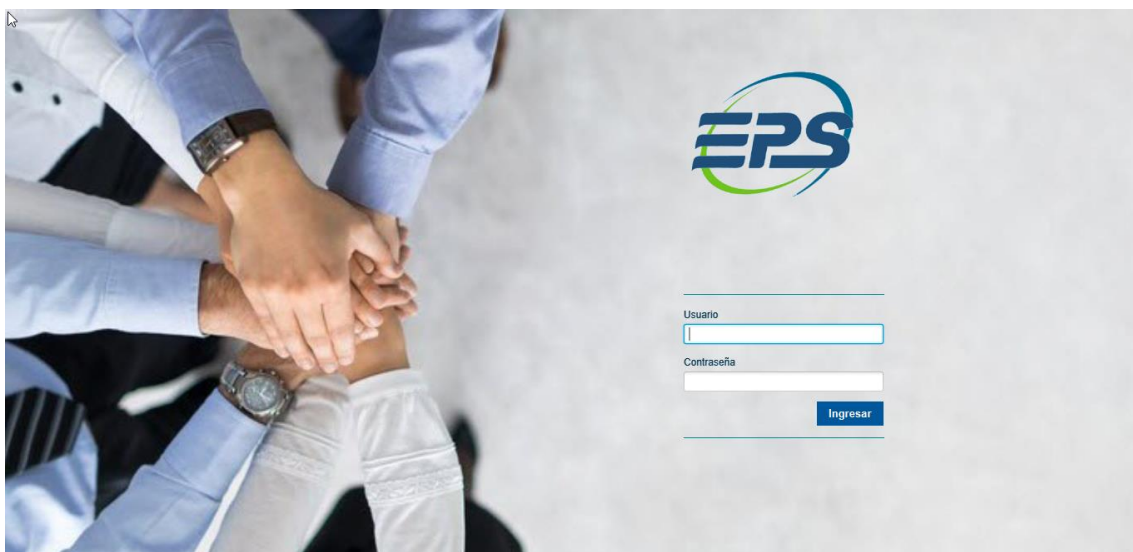


Acceso directo al sistema

Colocamos nuestras credenciales y podemos trabajar

#### **Ilustración 17**

*Pantalla de Inicio de sesión*



Pantalla de Inicio de sesión del sis