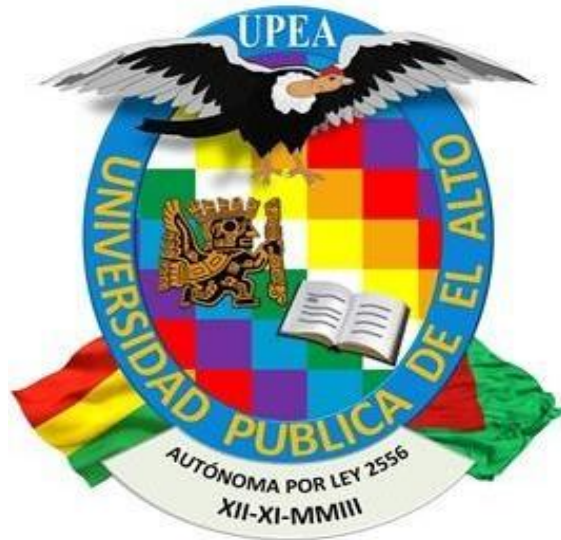


UNIVERSIDAD PÚBLICA DE EL ALTO

CARRERA INGENIERÍA DE SISTEMAS



PROYECTO DE GRADO

“SISTEMA WEB PARA EL SEGUIMIENTO Y CONTROL DE SOCIOS PROPIETARIOS DE MOVILIDADES”

CASO: SINDICATO SEÑOR DE SAN ROQUE

Para optar el título de Licenciatura en Ingeniería de Sistemas

MENCIÓN: GESTIÓN Y PRODUCCION

Postulante: Franz Guillermo Quispe Apaza

Tutor Metodológico: Ing. Marisol Arguedas Balladares

Tutor Revisor: M. Sc. Zara Yujra Cama

Tutor Especialista: Lic. Margarita Bernarda López Mariaca

EL ALTO – BOLIVIA

2022

(TALLER DE GRADO II)
DECLARACION JURADA DE
AUTENTICIDAD Y RESPONSABILIDAD

Yo Franz Guillermo Quispe Apaza estudiante con C.I. 6987939LP mediante la presente **declaro** de manera pública que la propuesta del **TRABAJO DE GRADO** titulada "**SISTEMA WEB PARA EL SEGUIMIENTO Y CONTROL DE SOCIOS PROPIETARIOS DE MOVILIDADES CASO: SINDICATO "SEÑOR DE SAN ROQUE"**" 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 cuestionarios que se desprenden de su lectura.

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

De identificarse falsificación, plagio, fraude, o que el **TRABAJO DE GRADO** haya sido publicado anteriormente; asumo las consecuencias y sanciones que de mi acción se deriven, responsabilizándome por todas as 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, 15 de noviembre del 2022



Estudiante: Franz Guillermo Quispe Apaza

C.I. 6987939LP

ingfgqa2020@gmail.com

DEDICATORIA

Dedico este trabajo principalmente a Dios, por haberme dado la vida y permitirme el haber llegado hasta este momento tan importante de mi formación profesional. A mi madre Querida Modesta Apaza Cocarico, que es el pilar más importante de mi vida, de haberme dado la vida, siempre confió en mí y nunca me abandonó a lo largo de mi carrera. A mi padre Marcelino Quispe Quispe, por ser el segundo pilar más importante y por demostrarme todo su apoyo incondicional.

A mi hijo Dylan Matías Quispe por ser la luz que ilumina mi camino durante todo este trayecto de mi trabajo y al amor de mi vida Rosa Antonia Quispe Quea, que siempre ha estado conmigo brindándome todo su cariño y su apoyo en cada momento de mi trabajo final.

AGRADECIMIENTOS:

El presente trabajo agradezco a Dios por ser mi guía y acompañarme en el transcurso de mi vida, brindándome paciencia y sabiduría para culminar con éxito mis metas propuestas. A mis padres por ser el pilar fundamental y haberme apoyado incondicionalmente en todo momento.

A mi tutora metodológico Ing. Marisol Arguedas Balladares por su conocimiento, apoyo, confianza, y motivación que me brindo en este proyecto de grado.

A mi tutor especialista Lic. Margarita Bernarda López Mariaca por brindarme sus orientaciones, compartir su conocimiento y las sugerencias durante el desarrollo del presente proyecto.

A mi tutor revisor M. Sc. Zara Yujra Cama por su disponibilidad de tiempo, sus acertadas y orientadas observaciones brindadas en la realización del proyecto.

Agradezco a todos los docentes por su conocimiento y apoyo, motivaron a desarrollarme como persona y profesional en la Universidad Pública de El Alto, por acogerme en sus aulas durante los años de estudio así también a la carrera Ingeniería de Sistemas y mis compañeras (os) de estudio por su apoyo incondicional.

RESUMEN

El presente proyecto de grado que es el desarrollo de un Sistema Web para dar el Seguimiento y Control de Socios Propietarios de Movilidades en el Sindicato Señor de San Roque, tiene los siguientes capítulos:

Capítulo I Marco Preliminar

Este primer capítulo describe el aspecto general del proyecto de grado como: la introducción, antecedentes, planteamiento del problema, objetivos, justificación, límites y alcances de esta forma dejar en claro los resultados las cuales se desea al concluir el proyecto de grado.

Capítulo II Marco Teórico

En este segundo capítulo proporcionara los fundamentos teóricos acerca de la metodología, métodos y herramientas empleados para el desarrollo del presente proyecto de grado, recopilando la información para el sustento de estudio que se realiza en el desarrollo del sistema.

Capítulo III Marco Práctico

En este tercer capítulo se realiza el análisis y diseño del nuevo sistema donde se va desarrollar de acuerdo al requerimiento de la Institución tomando en cuenta la metodología a seguir de acuerdo a las funcionalidades requeridas.

Capítulo IV Métricas de Calidad Estimación de Costos y Seguridad

En este cuarto capítulo se aplican las métricas de calidad de software para comprobar que el mismo es aceptable y cumple con los requisitos de la ISO 9126 al producto final y así también se cotizara los costos del producto en base a COCOMO, para determinar el costo final del software.

Capítulo V, Conclusiones y Recomendaciones

Finalmente en este último capítulo se mencionan las conclusiones que se formularon al concluir el proyecto de grado, satisfaciendo las necesidades y los requerimientos de la Institución, logrando cumplir con los objetivos planteados desarrollando un sistema eficiente, confiable para el usuario con los resultados obtenidos durante el desarrollo del proyecto, así también se da a conocer las recomendaciones correspondientes a futuro.

ABSTRACT

This degree project, which is the development of a Web System to monitor and control Mobility Owner Partners in the Señor de San Roque Union, has the following chapters:

Chapter I Preliminary Framework

This first chapter describes the general aspect of the degree project as: the introduction, background, problem statement, objectives, justification, limits and scope in this way to make clear the results which are desired at the end of the project. degree.

Chapter II Theoretical Framework

In this second chapter, it will provide the theoretical foundations about the methodology, methods and tools used for the development of this degree project, collecting the information for the support of the study that is carried out in the development of the system.

Chapter III Practical Framework

In this third chapter, the analysis and design of the new system is carried out, where it will be developed according to the Institution's requirement, taking into account the methodology to be followed according to the required functionalities.

Chapter IV Quality Metrics Estimation of Costs and Security

In this fourth chapter, the software quality metrics are applied to verify that it is acceptable and meets the requirements of ISO 9126 to the final product and thus the costs of the product will also be quoted based on COCOMO, to determine the final cost. of the software.

Chapter VI, Conclusions and Recommendations

Finally, in this last chapter, the conclusions that were formulated at the end of the degree project are mentioned, satisfying the needs and requirements of the Institution, managing to meet the objectives set by developing an efficient, reliable system for the user of the results obtained during the study. development of the project, thus the corresponding recommendations for the future are also made known.

INDICE GENERAL

	Pág.
1. CAPÍTULO I: MARCO PRELIMINAR	
1.1. Introducción	1
1.2. Antecedentes	1
1.2.1. Antecedentes de la Institución.....	1
1.3. Planteamiento del Problema	7
1.3.1. Problema Principal	7
1.3.2. Problemas Secundarios	7
1.4. Objetivos	8
1.4.1. Objetivo General	8
1.4.2. Objetivos Específicos	8
1.5. Justificaciones	8
1.5.1. Justificación Técnica	8
1.5.2. Justificación Económica.....	8
1.5.3. Justificación Social.....	9
1.6. Metodologías.....	9
1.6.1. Metodología UWE	9
1.6.2. Métrica de Calidad	10
1.6.3. Modelo de Costos COCOMO	10
1.7. Herramientas.....	10
1.7.1. Lenguajes de Programación	10
1.7.2. Servidor Web.....	12
1.7.3. Diseño de Interfaces	12

1.7.4.	Editores.....	13
1.7.5.	Base de Datos	13
1.8.	Límites y Alcances.....	14
1.8.1.	Limites.....	14
1.8.2.	Alcances	14
1.9.	Aportes.....	15

2. CAPÍTULO II: MARCO TEÓRICO

2.1.	Introducción	16
2.2.	Conceptos Básicos	16
2.2.1.	Sistema	16
2.2.2.	Sistema de Información.....	17
2.2.2.1.	Actividades de un Sistema de Información.....	17
2.2.2.2.	Ciclo de Vida de los Sistemas de Información	19
2.2.3.	Control.....	20
2.3.	Requerimientos del Sistema.....	20
2.3.1.	Requerimientos Funcionales	21
2.3.2.	Requerimientos no Funcionales	21
2.4.	Metodología UWE (UML-BASED WEB ENGINEERING).....	21
2.4.1.	Introducción	21
2.4.2.	Características de la Metodología UWE	22
2.4.3.	Actividades del Modelado UWE.....	23
2.4.4.	Fases de la Metodología UWE.....	23
2.4.5.	Modelos de la Metodología UWE.....	24
2.4.5.1.	Modelo de Casos de Uso.....	25

2.4.5.2.	Modelo Conceptual	26
2.4.5.3.	Modelo de Navegación	28
2.4.5.4.	Modelo de Procesos	30
2.4.5.5.	Modelo de Presentación	33
2.5.	Normas ISO 9126	35
2.5.1.	Características	36
2.5.2.	Factores de Calidad	39
2.5.3.	Ventajas y Desventajas Normas ISO 9126	39
2.5.4.	Formulas Normas ISO 9126.....	40
2.5.5.1.	Funcionalidad.....	40
2.5.5.2.	Confiabilidad.....	41
2.5.5.3.	Usabilidad	42
2.5.5.4.	Mantenibilidad	42
2.5.5.5.	Portabilidad	43
2.6.	Modelo de Estimación de Costo de Software	44
2.6.1.	COCOMO	44
2.7.	Seguridad	49
2.7.1.	Encriptación y Contraseñas en PHP	49
2.8.	Herramientas	56
2.8.1.	Gestor de Base de Datos MariaDB	56
2.8.2.	Lenguaje de Programación PHP	56
2.8.3.	Servidor Web Apache	58
2.8.4.	Bootstrap	58
2.8.5.	JQuery	59

2.8.6.	Json.....	59
2.8.7.	Html5.....	60
2.8.8.	Css3	62
2.9.	Pruebas de Software.....	66
2.9.1.	Pruebas de Caja Blanca.....	63
2.9.2.	Pruebas de Caja Negra	63
2.9.3.	Pruebas de Funcionalidad.....	63

3. CAPÍTULO III: MARCO APLICATIVO

3.1.	Introducción	64
3.2.	Ingeniería Web Basada UML (UWE).....	64
3.2.1.	Obtención de Requerimientos	64
3.2.1.1.	Definición de Actores	65
3.2.2.	Requerimientos del Sistema.....	66
3.2.2.1.	Requerimientos Funcionales.....	66
3.2.2.2.	Requerimientos no Funcionales.....	67
3.3.	Fase de Diseño del Sistema.....	68
3.3.1.	Diagrama de Caso de Uso General	68
3.3.2.	Diagrama de Caso de Uso Administración del Sistema.....	69
3.3.3.	Diagrama de Caso de Uso Personal Administrativo del Sistema.....	70
3.3.4.	Diagrama de Caso de Uso Jefe de Grupo.....	71
3.3.5.	Diagrama de Caso de Uso Asignación de Vehículos	72
3.3.6.	Diagrama de Caso de Uso Reportes	73
3.4.	Diagrama de Clases.....	74
3.5.	Diagrama de Contenido	76

3.6.	Diagrama de Navegación.....	77
3.7.	Diagrama de Presentación.....	78
3.7.1.	Modelo de Presentación Login.....	78
3.7.2.	Modelo de Presentación Administrador.....	79
3.7.3.	Modelo de Presentación Jefe de Grupo.....	80
3.7.4.	Modelo de Presentación Conductor.....	80
3.7.5.	Modelo de Presentación Agente.....	81
3.8.	Fase de Codificación del Software.....	81
3.9.	Fase de Pruebas.....	82
3.10.	Fase de Implementación del Sistema.....	89
3.10.1.	Interfaz Inicio de Sesión.....	89
3.10.2.	Interfaz Administrador.....	90
3.10.3.	Interfaz Administrativo.....	90
3.10.4.	Interfaz Agente.....	91
3.10.5.	Interfaz Jefe de Grupo.....	91
3.10.6.	Interfaz Conductor.....	92
3.11.	Fase de Mantenimiento.....	93
3.12.	Métodos de Prueba del Software.....	93
3.12.1.	Pruebas de Caja Blanca.....	93
3.12.2.	Pruebas de Caja Negra.....	95
3.12.3.	Pruebas de Funcionalidad.....	96
4.	CAPÍTULO IV: METRICAS DE CALIDAD, ESTIMACION DE COSTOS Y SEGURIDAD	
4.1.	Introducción.....	98

4.2.	Calidad de Software	98
4.2.1.	Funcionalidad	98
4.2.2.	Confiabilidad	101
4.2.3.	Usabilidad.....	102
4.2.4.	Mantenibilidad.....	104
4.2.5.	Portabilidad.....	104
4.2.6.	Resultado Final	105
4.3.	Estimación de Costos	105
4.3.1.	Método de Estimación COCOMO	106
4.4.	Seguridad del Sistema	109
4.4.1.	Uso del Password	109
4.4.2.	Encriptación de Datos.....	110
5.	CAPÍTULO V: CONCLUSIONES Y RECOMENDACIONES	
5.1.	Conclusiones	111
5.2.	Recomendaciones.....	111
	Bibliografía.....	112

ANEXOS

A. ÁRBOL DE PROBLEMAS

B. ÁRBOL DE OBJETIVOS

C. DOCUMENTACION DE LA
INSTITUCIÓN

D. MANUAL DE USUARIO

E. AVALES

INDICE DE FIGURAS

Figura 2.1. Grafico General de un Sistema	16
Figura 2.2. Fases de la Metodología UWE	24
Figura 2.3. Modelo de la Metodología UWE.....	24
Figura 2.4. Modelo de Casos de Uso	25
Figura 2.5. Modelo del Contenido	27
Figura 2.6. Estereotipos de Estructura de Navegación	28
Figura 2.7. Modelo de Navegación	29
Figura 2.8. Modelo de Procesos.....	31
Figura 2.9. Modelo de Proceso: Inicio	32
Figura 2.10. Modelo de Proceso: Buscar	32
Figura 2.11. Modelo de Presentación de Inicio	34
Figura 2.12. Página de Modelo de Presentación de Búsqueda	35
Figura 2.13. Encriptación de Contraseña MD5.....	56
Figura 3.1. Diagrama de Caso de Uso General	68
Figura 3.2. Diagrama de Caso de Uso: Administración del Sistema.....	69
Figura 3.3. Diagrama de Caso de Uso: Personal Administrativo	70
Figura 3.4. Diagrama de Caso de Uso: Jefe de Grupo.....	71
Figura 3.5. Diagrama de Casos de Uso: Asignación de Vehículos.....	72

Figura 3.6. Diagrama de Caso de Uso: Reportes	73
Figura 3.7. Diagrama de Clases	74
Figura 3.8. Diagrama de Flujo de Datos	75
Figura 3.9. Modelo de Contenido: Dato Relacional	76
Figura 3.10. Diseño de Navegación	77
Figura 3.11. Modelo de Presentación Login	78
Figura 3.12. Modelo de Presentación Administrador	79
Figura 3.13. Modelo de presentación Jefe de Grupo	80
Figura 3.14. Modelo de Presentación Conductor.....	80
Figura 3.15. Modelo de Presentación Agente	81
Figura 3.16. Código Fuente.....	81
Figura 3.17. Módulo de Usuarios: Administrador, Administrativo, Agente	82
Figura 3.18. Modulo Conductores	82
Figura 3.19. Diseño Conductor: Datos como Persona	83
Figura 3.20. Diseño Conductor: Datos como Conductor.....	83
Figura 3.21. Diseño Conductor: Datos como Usuario	84
Figura 3.22. Diseño Conductor: Archivos Subidos	84
Figura 3.23. Reporte Conductor: Memorándum de Trabajo.....	85
Figura 3.24. Reporte Conductor.....	85
Figura 3.25. Modulo Aportes	86

Figura 3.26. Módulo de Asistencia de Conductores Mediante Placas	86
Figura 3.27. Módulo Vehículos	87
Figura 3.28. Modulo Grupo	87
Figura 3.29. Modulo Línea	88
Figura 3.30. Modulo Ruta	88
Figura 3.31. Logo del Sistema	89
Figura 3.32. Interfaz Inicio de Sesión Login	89
Figura 3.33. Interfaz Administrador del Sistema	90
Figura 3.34. Interfaz Administrativo.....	90
Figura 3.35. Interfaz Agente	91
Figura 3.36. Interfaz Jefe de Grupo	91
Figura 3.37. Interfaz Conductor.....	92
Figura 3.38. Pruebas de Caja Blanca	94
Figura 3.39. Pruebas de Caja Negra Iniciar Sesión.....	96
Figura 4.1. Backup de la Base de Datos	110

INDICE DE TABLAS

Tabla 1.1. Cantidad de Movilidades del Sindicato Señor de San Roque.....	2
Tabla 1.2. Cantidad de Movilidades, Grupos y Líneas del Sindicato Señor de San Roque.	3
Tabla 1.3. Total de Conductores en la Institución del Sindicato Señor de San Roque.....	4
Tabla 2.1. Características de Calidad.....	36
Tabla 2.2. Coeficiente del Modelo Básico.....	46
Tabla 2.3. Coeficiente del Modelo Intermedio COCOMO.....	47
Tabla 3.1. Obtención de Requerimientos.....	64
Tabla 3.2. Definición de Actores	65
Tabla 3.3. Requerimientos Funcionales.....	66
Tabla 3.4. Requerimientos no Funcionales.....	67
Tabla 3.5. Descripción de Caso de Uso: Administrador del Sistema	69
Tabla 3.6. Descripción de Caso de Uso: Personal Administrativo	70
Tabla 3.7. Descripción de Caso de Uso: Jefe de Grupo.....	71
Tabla 3.8. Descripción de Casos de Uso: Asignación de Vehículos	72
Tabla 3.9. Descripción de Casos de Uso: Reportes	73
Tabla 3.10. Valores de Límites de Inicio de Sesión.....	96
Tabla 3.11. Pruebas de Caja Negra Inicio de Sesión	96
Tabla 3.12. Pruebas de Funcionalidad	97
Tabla 4.1. Número de Entrada de Usuario.....	98

Tabla 4.2. Calculo de Funcionalidad de Punto de Función	99
Tabla 4.3. Factores de Complejidad.....	100
Tabla 4.4. Escala de Valores de Preguntas	102
Tabla 4.5. Preguntas para Determinar la Usabilidad del Sistema	103
Tabla 4.6. Resultado Total	105
Tabla 4.7. Coeficiente de a, b, c, d COCOMO	106
Tabla 4.8. Valores de Costo COCOMO DETALLADO	107
Tabla 4.9. Costo de Elaboración de Proyecto	109
Tabla 4.10. Costo Total del Proyecto.....	109

CAPÍTULO I

MARCO PRELIMINAR

1.1.Introducción

Las nuevas tendencias tecnológicas ofrece un sin fin de herramientas tecnológicas, para que nuestra sociedad este cada día más conectada electrónicamente, facilitar a las personas e incrementar su eficacia en el desarrollo de las actividades creando condiciones de gran competencia y dinamismo en todos los ámbitos del desarrollo del país.

Dentro de estas herramientas tecnológicas existen un conjunto de aplicaciones, en la cual hace falta implementar en la Institución del Sindicato Señor de San Roque, para que el personal administrativo tenga un mejor ámbito laboral, y así los Jefes de grupo de la Institución puedan ingresar al sistema y recabar su memorándum de trabajo de los Conductores.

El proyecto de investigación tiene el propósito de elaborar un Sistema Web para el Seguimiento y Control de Socios Propietarios de Movilidades en el Sindicato Señor de san Roque, para mejorar su ámbito social y laboral, con el fin de automatizar los controles de la información respecto a cada operación realizado, además incrementara la productividad del Sindicato y evitara la duplicidad de información, también se centralizara la información en una base de datos realizados, para que así lo requieran a futuro el cual podrán obtener informes y las estadísticas cuando sea necesario.

El presente proyecto estará basado en la metodología UWE, debido que se adapta a cualquier tipo de proyecto y resolución de problemas, para la base de datos se usara MySQL, como lenguaje de programación PHP y APACHE como servidor, estas herramientas se eligen por medio de comparaciones para lograr las especificaciones más adecuadas.

1.2.Antecedentes

1.2.1.Antecedentes de la Institución

La Institución del Sindicato Señor de San Soque, actualmente se encuentra ubicado en el distrito 7 de la ciudad del alto, Zona San Roque Calle bolívar Nro. 876, se ha fundado el 04 de agosto del 2001 con resolución suprema Nro. 24462.

En 2001 los vecinos de esta zona sintiendo la necesidad de falta de movilidad en la

zona urbana impulsaron a los vecinos y sectores para la creación de la nueva Institución Señor de San Roque, para que se haga un mejor servicio público a los vecinos ya que así se crearon poco a poco diferentes líneas y cada uno con diferentes rutas para las zona urbana.

Misión

La Institución del Sindicato Señor de San Roque presta el servicio de transporte público de movilidades con diferentes líneas a distintitos sectores de la zona. En nuestro esfuerzo continuo por conseguir ser líderes y ofrecer un servicio de transporte competitivo, el objetivo fundamental es satisfacer al máximo las necesidades de movilidad de nuestras personas o usuarios, ofreciendo un servicio de alta calidad.

Visión

La Institución del Sindicato Señor de san Roque pretende ser líder dentro del ámbito de movilidad de nuestra población, tanto en servicios ofertados como en calidad. Desarrollaremos distintas medidas con el fin de mejorar la imagen que la sociedad mantiene respecto a nuestra Institución.

Tabla 1.1.

Cantidad de Movilidades del Sindicato Señor de San Roque.

Movilidades	Cantidad
Minibús	1546
Taxi	136
Trufis	118
Total	1800

Nota. En la siguiente tabla 1.1. Vemos la cantidad de Movilidades: Minibús, Taxi, Trufis.

Tabla 1.2.*Cantidad de Movilidades, Grupos y Líneas del Sindicato Señor de San Roque.*

N	Grupos	Líneas	Cantidad	Tipo
1	A pioneros	698 negro	180	Minibús
2	B leones	737 negro	150	Minibús
3	C intocables	490 negro	80	Minibús
4	D escorpión	768 negro	120	Minibús
5	E correccaminos	737 rojo	112	Minibús
6	F andino	768 rojo	70	Minibús
7	G geniales	698 rojo	70	Minibús
8	H académicos	470 rojo	67	Minibús
9	I dinámicos	468 negro	120	Minibús
10	J tigres	470 rojo	113	Minibús
11	K lobos	402 negro	90	Minibús
12	L vencedores	468 rojo	124	Minibús
13	N diablos rojos	402 rojo	136	Minibús
14	P ejecutivos	490 rojo	114	Minibús
15	Carrys Illimani	100 negro	16	Trufis
16	Carrys escorpión	143 rojo	49	Trufis
17	Carrys Halcones	100 rojo	53	Trufis
18	Rey de reyes		28	Taxi
19	Fenix del norte		30	Taxi
20	Aguilas reales		28	Taxi
21	Aguilas por siempre		23	Taxi
22	Aguilas		27	Taxi
Total			1800	

Nota. En la siguiente tabla 1.2. Vemos los Grupos con sus respectivas líneas y también la cantidad de Movilidades: Minibús, Taxi, Trufis.

Tabla 1.3.

Total de Socios Propietarios de la Institución del Sindicato Señor de San Roque.

Conductores	Cantidad
Propietarios	1800
Asalariados	156
Total	1856

Nota. En la presente tabla 1.3. Vemos la cantidad de Conductores.

Antecedentes a Fines de Trabajo

Luego de realizar un relevamiento de información y una revisión bibliográfica entre algunos trabajos afines al presente tema, se tiene las siguientes bibliografías:

Antecedentes Internacionales

"SISTEMA DE GESTIÓN WEB PARA EL CONTROL ADMINISTRATIVO DE SOCIOS Y UNIDADES DE TRANSPORTE COOPERATIVA DE TAXIS TAXIBUCAY S. A." , Con el objetivo de una implementación de una aplicación web por medio de herramientas de programación como PHP y MySQL, para controlar los procesos administrativos de la cooperativa de taxis TAXIBUCAY S.A. del cantón Bucay, La metodología que se aplicó en esta propuesta tecnológica fue el modelo en cascada, este modelo es un proceso de desarrollo secuencial, en el cual la elaboración del software se realizó como un conjunto de etapas que se ejecutan una tras otra. Se la conoce como cascada por las posiciones que tienen las diferentes fases que componen el proyecto, ubicadas una encima de la otra y continuando el flujo de ejecución de arriba hacia abajo simulando una cascada, Con los datos obtenidos, se procedió a planificar el proyecto en base a la metodología de diseño de software conocida como modelo en cascada, la misma que permite ir realizando las actividades por fases. Se inició con la fase del análisis, la que permitió analizar todos los problemas

y establecer los objetivos de solución, luego se procedió a diseñar el modelo del sistema con la construcción de la base de datos y los diagramas de procesos, a continuación se procedió a codificar el sistema, armando cada uno de los módulos que forman parte del programa, finalmente se realizaron las pruebas para dejar a punto el software y proceder a la implementación. (Scotland Muñoz, 2020)

“DESARROLLO E IMPLEMENTACION DE UN SISTEMA DE SEGUIMIENTO Y CONTROL DEL SERVICIO VEHICULAR EN LA ESTACION LA JUANITA, BOGOTA”, Con el Objetivo de Desarrollar un sistema de seguimiento y control para la optimización de los procesos del servicio de mantenimiento vehicular en la estación la juanita, Bogotá, La metodología de investigación a utilizar es no experimental porque no manipula variables independientes, es decir, se usa la observación de situaciones existentes, En conclusión el sistema que se elaboró ayuda a cumplir con los parámetros del diagnóstico del estado actual del proceso en el que se encuentra el vehículo, pues este le permite saber al cliente en qué fase se encuentra del proceso. Adicionalmente, este permite realizar un análisis a través del proceso diagnosticado, este se genera por medio del sistema de seguimiento y control en el servicio de mantenimiento vehicular en la estación. Además, se crean unas estrategias que permiten optimizar el servicio de mantenimiento vehicular que se da por medio del modelo, puesto que este se da a partir de los resultados obtenidos de la investigación. Para finalizar, el desarrollo de este sistema de seguimiento y control ayuda a que esta área tenga un proceso del servicio de mantenimiento vehicular de la empresa y/o taller de Yavegas, sin embargo, este no soluciona el problema de raíz, si no que permite regular y control el funcionamiento de esta área. (Ramirez Cortes, 2022)

“IMPLEMENTACION DE UN SISTEMA WEB PARA EL CONTROL DE UNIDADES DE UNA EMPRESA DE TRANSPORTE URBANO”, Como objetivo es Desarrollar un sistema web para el control de unidades que permita mejorar la gestión de transportes, también en este estudio se desarrolla el método RUP, el cual puede asegurar la calidad de software, especialmente es de gran apoyo para que el personal de programación

se enfoque en satisfacer los requerimientos del sistema. Se concluyó que aplicando RUP como método para la programación y UML como lenguaje de modelado, y continuando con las pautas de seguridad del sistema OWASP llegamos a la conclusión de que El diseño e implementación de un sistema web para el control de unidades permite mejorar la gestión de transportes. (Tenazoa Cuba, 2022)

Antecedentes Nacionales

“MODELO DE REALIDAD AUMENTADA PARA EL CONTROL DE TRAMOS DEL TRANSPORTE PÚBLICO BASADO EN ANDROID Y CÓDIGO QR” CASO: SINDICATO “SIMON BOLIVAR”, su objetivo es Desarrollar un modelo de realidad aumentada para el control de tramos del transporte público para dispositivo móvil, que permita al usuario conocer el recorrido de un vehículo del transporte público mediante código QR, verificando la localización del usuario en la ruta Es un método para el desarrollo de Aplicaciones Móviles donde se tienen prácticamente los mismos problemas de desarrollos de software. El objetivo de este método es conseguir ciclos de desarrollo muy rápidos en equipos muy pequeños. Mobile - D está basado en metodologías conocidas: XP, Crystal y RUP se compone de distintas fases, a la conclusión por esta razón que se llevó a cabo la investigación del caso, aplicando el Modelo de Realidad Aumentada para el Control de tramos del transporte público dimos soluciones alternativas y modernas para un mejor servicio de autotransporte en el caso particular del Sindicato “Simón Bolívar”. (Cruz Sirpa, 2020)

“SISTEMA WEB DE CONTROL Y SEGUIMIENTO DE SERVICIOS Y GESTION DE CLIENTES PARA LA EMPRESA CONSULTORA AYS S.R.L., Su Objetivo tiene a Desarrollar un sistema web de control y seguimiento de servicios y gestión de clientes para La Empresa Consultora “Contadores Públicos & Auditores AyS S.R.L.”, El presente proyecto claramente será una referencia para el desarrollo de proyectos orientados al desarrollo web pues en el sistema se aplicará la metodología Kanban el cual tiene la cualidad de ser flexible, especialmente en la entrada de tareas, así como en el seguimiento de estas, la priorización, la supervisión del equipo de trabajo y los

informes de dedicación. En cuanto al modelado y análisis del sistema se empleará UML basado en Ingeniería Web (UWE). Habiendo culminado el desarrollo y la implementación del Sistema de para la empresa Consultora AyS S.R.L., donde se aplicó todo el conocimiento de metodologías de análisis y diseño de software, todo en base a los requerimientos planteados por el beneficiado se puede concluir que se cumplió con los objetivos planteados. (Mamani Condori, 2020)

1.3.Planteamiento del Problema

1.3.1.Problema Principal

Luego de haberse realizado un relevamiento de información se ha realizado un Análisis de los problemas existentes en la institución, por lo cual se ha identificado que la forma en que se realiza el procesamiento de información en cuanto a: Afiliaciones de Socios Propietarios, asalariados, Asignación de Grupos, se realizan de manera automática en aplicaciones ofimáticas por lo que se dificulta la generación de consultas e informes requeridos en el Sindicato Señor de san Roque de forma rápida, oportuna y confiable.

1.3.2.Problemas Secundarios

- No se tiene un conocimiento preciso sobre los fundamentos teóricos relacionados a: Metodología UWE, COCOMO y Seguridad de Encriptación.
- Se requiere realizar un diagnóstico de los procesos realizados de: Afiliaciones y su Seguimiento en el Sindicato Señor de san Roque.
- Se hace necesario realizar mejoras en los procesos de afiliaciones y su seguimiento.

Formulación del Problema

¿Cómo lograr el Seguimiento y Control de Socios Propietarios de Movilidades en el Sindicato Señor de San Roque, que le permita realizar un trabajo eficiente y prestar un adecuado servicio a la Institución?

1.4.Objetivos

1.4.1.Objetivo General

Desarrollar un Sistema de Información Web para el Seguimiento y Control de Socios Propietarios de Movilidades en el Sindicato Señor de san Roque, para obtener información confiable y oportuna para coadyuvar a una mejor toma de decisiones a los directivos.

1.4.2.Objetivos Específicos

- Fundamentar los conocimientos teóricos relacionados a la: Metodología UWE, COCOMO, Seguridad de Encriptación.
- Realizar un diagnóstico de los procesos de afiliación y seguimiento de socios propietarios del Sindicato Señor de San Roque.
- Implementar un módulo de conductores para tener información de los socios propietarios en los diferentes grupos, módulo de vehículos de acuerdo al requerimiento de la Institución, módulo de grupos en función a la cantidad de registrados para su respectivo control en la Institución.

1.5.Justificaciones

1.5.1.Justificación Técnica

Se justifica técnicamente, porque la institución cuenta con recursos tecnológicos para el desarrollo del Sistema de Información Web para el Seguimiento y Control de Socios Propietarios de Movilidades en el Sindicato Señor de San Roque, ya que el presente proyecto contará con una nueva herramienta de información que será de forma gratuito, aplicando métodos, metodologías y técnicas para su desarrollo ofreciendo información de forma rápida, oportuna y confiable.

1.5.2.Justificación Económica

El presente proyecto se justifica económicamente porque reducirá los materiales de escritorio puesto que el registro será digital así mismo, ya que será accesible para los afiliados así también el monto que corresponde a los aportes, multas y otros serán

correctamente controlados y administrados por la Institución.

1.5.3. Justificación Social

El personal de la Institución será beneficiado con el Sistema Web para obtener información requerida de forma rápida y segura, sobre todo ayudará a los afiliados de la Institución desde cualquier lugar, brindándole un manejo amigable y confiable.

Los beneficiados del sistema serán:

- La Institución, ya que tendrán un mejor control en los socios propietarios y en qué grupo se encuentra sus vehículos.
- Los Usuarios, la cual tendrán acceso a visualizar y obtener información de su vehículo.

1.6. Metodologías

1.6.1. Metodología UWE

UWE es una metodología basada en el Proceso Unificado y UML para el desarrollo de aplicaciones Web, cubre todo el ciclo de vida de las aplicaciones Web. Su proceso de desarrollo se base en tres frases principales: la fase de captura de requisitos, la fase de análisis y diseño y la fase de la implementación.

La metodología UWE define vistas especiales representadas gráficamente por diagramas en UML, tales como el modelo de navegación y el modelo de presentación.

Los principales artefactos que produce el método de diseño de UWE son los siguientes:

- Un Modelo de Requerimientos que captura los requerimientos del sistema.
- Un Modelo de contenido.
- Un Modelo de Usuario.
- Un Modelo de Navegación que comprende la estructura de la navegación.
- Un Modelo de Presentación que abarca modelos estáticos y dinámicos (modelo de estructura de la presentación, modelo del flujo de la presentación, modelo de interface abstracta de usuario, y modelo de ciclo de vida del objeto). (Olsina, 2008)

1.6.2.Métrica de Calidad

Las Métricas de Calidad proporcionan una indicación de cómo se ajusta el software, a los requerimientos implícitos y explícitos del cliente.

Modelo de Calidad Establecido por el Estándar ISO 9126

La ISO/IEC9126 o ISO 9126 (estándar internacional para la evaluación de la calidad de producto de software) publicado en 1992, fue sustituido por el proyecto SQuaRE, ISO 25000:2005, el cual sigue los mismos conceptos con el modelo de calidad de producto propuesto por McCall. El modelo de calidad establecido en la primera parte del estándar, ISO 9126-1, clasifica la calidad del software en un conjunto estructurado de características y sus características de la siguiente manera: funcionalidad, confiabilidad, usabilidad, eficiencia, mantenibilidad y portabilidad.

1.6.3.Modelo de Costos COCOMO

El Modelo Constructivo de Costos, es un modelo matemático de base empírica utilizado para estimación de costos de software. Incluye tres submodelos, cada uno ofrece un nivel de detalle y aproximación, cada vez mayor, a medida que avanza el proceso de desarrollo del software: básico, intermedio y detallado.

1.7.Herramientas

Las herramientas para el desarrollo Sistema Web para el Seguimiento y Control de Socios Propietarios de Movilidades para este trabajo son las siguientes:

1.7.1.Lenguajes de Programación

- **PHP:** El PHP es un lenguaje de scripting de código abierto, del lado del servidor, con programación HTML integrada que se utiliza para crear páginas web dinámicas. Las ventajas de PHP son su flexibilidad y su alta compatibilidad con otras bases de datos. Además, PHP es considerado como un lenguaje fácil de aprender.

Las funciones de PHP se relacionan con los scripts del lado del servidor. PHP puede realizar cualquier tarea que cualquier programa CGI (Common Gateway

Interface) puede hacer y maneja el intercambio de datos entre el servidor y el software.

Tres funciones principales distinguen en particular el PHP:

Programación del lado del servidor: esto requiere tres componentes, que son un analizador PHP, un navegador web y un servidor web. Este último está conectado con una instalación PHP. El navegador web se utiliza para acceder al programa PHP. El navegador muestra páginas web que se almacenan en el servidor y se generan con PHP. La programación del lado del servidor es ideal para webs de prueba antes de su publicación en la web.

Programación a través de la línea de comandos: Los scripts PHP pueden ser creados sin un navegador o servidor. En este caso, sólo necesitarás un analizador PHP. Estos scripts son adecuados para las tareas regulares que se llevan a cabo en una web.

Escribir aplicaciones de escritorio: esta función no se utiliza con tanta frecuencia como la programación de páginas web. Sin embargo, el PHP es adecuado para el desarrollo de aplicaciones de escritorio debido a su complejidad, que también puede ser utilizado a través de múltiples plataformas.

- **JQUERY:** JQuery es una librería de JavaScript (JavaScript es un lenguaje de programación muy usado en desarrollo web). Esta librería de código abierto, simplifica la tarea de programar en JavaScript y permite agregar interactividad a un sitio web sin tener conocimientos del lenguaje.

Todos los plugins de JQuery necesitan la librería de JQuery para funcionar. La librería es un archivo .js que se puede descargar desde el sitio oficial: <https://jquery.com/> colocar en una carpeta js y luego vincular con una etiqueta <script>, por ejemplo:

```
<Script src="js/jquery-3.2.1.min.js"></Script>
```

Ese vínculo conviene colocarlo dentro del body y hacia el final del HTML para no demorar la carga de los contenidos. (Chuburu, 2020)

1.7.2.Servidor Web

Es un programa especialmente diseñado para transferir datos de hipertexto, es decir, páginas web con todos sus elementos (textos, widgets, banners, etc.). Estos servidores web utilizan el protocolo Http://. Es un servidor web HTTP de código abierto para la creación de páginas y servicios web, es un servidor multiplataforma, una de las ventajas más grandes 16 de Apache es que es un servidor web multiplataforma, gratuito muy robusto y que destaca por su seguridad y rendimiento.

1.7.3.Diseño de Interfaces

- **HTML:** HTML5 se creó con el objetivo de que el proceso para escribir el código sea mucho más simple y lógico. Y es que, como ya mencionamos, la sintaxis de HTML5 es en el ámbito multimedia. La idea es que se pueda visualizar el contenido multimedia variado que se encuentra en Internet.

Las etiquetas son una herramienta fundamental para que todos los navegadores puedan traducir los códigos correctamente y que el usuario pueda ver las imágenes, texto, párrafo y todos los elementos de la página web, correctamente. Imagina que los navegadores con traductores de las etiquetas y con HTML5 se agregaron nuevas que ayudan a ahorrar el uso de productos que utilizaban para complementar y hacer tareas simples con el HTML.

Para que te hagas una idea de lo que esto significa, te ponemos un ejemplo. Un diseño nuevo para una página web, estará reflejado en etiquetas, como por ejemplo: <header>, <footer>, <nav>, <section>,<article>, estás reemplazarán las necesidades de tener una etiqueta para cada parte de la página. Se propone es tener etiquetas destinada para cada función. (Marcago, 2022)

- **CSS:** Antes de comenzar, debes tener claro un concepto clave: una página web es realmente un documento de texto. En dicho documento se **escribe** código HTML, con el que sé que crea el contenido de una web. Por otro lado, existe el código **CSS**, que unido al código HTML permite darle forma, color, posición (y otras características visuales) a una página.

En resumen, se trata de un idioma como podría ser el inglés o el alemán, que los

navegadores web como Chrome o Firefox conocen y pueden entender. Nuestro objetivo como diseñadores y programadores web es precisamente ese: aprender el idioma.

Las siglas **CSS** (Cascading Style Sheets) significan «Hojas de estilo en cascada» y parten de un concepto simple pero muy potente: aplicar **estilos** (colores, formas, márgenes, etc...) a uno o varios documentos (generalmente documentos HTML, páginas webs) de forma masiva.

Se le denomina estilos en cascada porque se aplican de arriba a abajo (siguiendo un patrón denominado herencia que trataremos más adelante) y en el caso de existir ambigüedad, se siguen una serie de normas para resolverla.

- La idea de CSS es la de utilizar el concepto de separación de presentación y contenido, intentando que los documentos HTML incluyan sólo información y datos, relativos al significado de la información a transmitir el contenido, y todos los aspectos relacionados con el estilo diseño, colores, formas.

1.7.4. Editores

Sublime Text es un editor de Texto para escribir código en casi cualquier formato de archivo. Está especialmente pensado para escribir sin distracciones. Esto quiere decir que visualmente ofrece un entorno oscuro donde las líneas de código que escribas resaltarán para que puedas centrarte exclusivamente en ellas.

Sublime Text es un editor de texto sofisticado para código, marcado y prosa. Tiene una interfaz de programación de aplicaciones Python (API). Es compatible con muchos lenguajes de programación y marcado, y su funcionalidad puede ser extendida por los usuarios con complementos, generalmente creados por la comunidad y mantenidos bajo licencias de software libre. (stack)

1.7.5. Base de Datos

MariaDB es un sistema de gestión de base de datos MySQL. El propósito de este sistema es proveer capacidades similares y extendidas en relación a MySQL, con un foco especial en mantener el software de forma libre a través de la licencia GNU GPL.

Con aplicaciones muy amplias, puede ser usado por equipos de desarrollo de software.
(Salas, s.f.)

1.8.Límites y Alcances

1.8.1.Limites

El proyecto se limita en los siguientes aspectos:

- El Sistema solo gestiona el flujo de información de la Institución.
- El Sistema no provee notificaciones o no contempla algunas funciones como es: brindar actividades o fechas recordatorios que se realiza en la Institución.
- El Sistema no podrá registrar Entrada y Salida del personal administrativo que trabaja en la Institución.
- El Sistema no realiza las planillas de pago en la Institución donde trabajan el personal administrativo.

1.8.2.Alcances

Los alcances del Sistema de manera general son lo siguiente:

- Módulo de Usuarios.- El Sistema realiza el registro de usuarios al personal autorizado con todos los datos personales que solicita la Institución.
- Módulo de Conductores.- El Sistema realiza el registro de socios propietarios con los requisitos que solicita la Institución.
- Módulo de Vehículos.- El Sistema realiza el registro de vehículos con los requerimientos correspondientes.
- Módulo de Grupo.- EL Sistema asigna diferentes grupos de manera fácil y confiable.
- Módulo de Línea.- El Sistema realiza el registro de cada línea.
- Módulo de Aportes y Multas.- los administradores realizan las multas correspondientes al afiliado y los aportes realizan a los nuevos afiliados para el ingreso a la Institución.

1.9.Aportes

El presente proyecto obtendrá un gran aporte para la Institución lo cual es proveer una herramienta factible, que permita un mejor control en las operaciones, lo cual implica que la información será ordenada en una base de datos que se lograra estar correctamente guardada y respaldada, así mismo esta herramienta protegerá la información generada.

Se facilita el trabajo de proyecto al General de la Institución brindando información y generar reporte, ayudando a facilitar la búsqueda de información.

Así mismo se logrará la sistematización del proceso aplicable y eficiente para mejorar el proceso en la toma de decisiones para la Institución del Sindicato Señor de San Roque.

CAPÍTULO II

MARCO TEÓRICO

2.1.Introducción

Realizar un Sistema de Información Web para el Seguimiento y Control de Socios Propietarios de Movilidades, ya que en todo desarrollo de software es de mucha importancia el seguir alguna especificación que permita a los desarrolladores el tener una disciplina que haga que todas las etapas del desarrollo del sistema.

El Presente proyecto en el capítulo se ven conceptos más relevantes sobre las metodologías, métodos y herramientas utilizadas para el desarrollo del sistema. Se aplicara con la metodología UWE que está basada en el Proceso Unificado y UML para el desarrollo web que cubre todo el ciclo de vida de las aplicaciones Web.

2.2.Conceptos Básicos

2.2.1.Sistema

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

Cada sistema puede ser estudiado con el objetivo de comprender el funcionamiento del mismo, descubrir sus límites/fronteras visibles y/o no visibles, entender el objetivo del mismo y cómo interactúa con otros sistemas externos. (Alegsa.com.ar, 2018)

Figura 2.1.

Grafico General de un Sistema.



Nota. Se puede observar el Grafico general del sistema entra y salida.

2.2.2.Sistema de Información

Se llama sistema de información (SI) a un conjunto de datos y elementos que interaccionan entre sí y que tienen un fin específico que, en general, tiene que ver con satisfacer una necesidad.

Los sistemas de información deben ser eficientes y de fácil ejecución porque procesan y almacenan una gran cantidad de datos. Estos datos son cargados, de forma automática o manual, en un soporte físico o digital y son usados para producir información que es útil para la ejecución de determinada actividad u objetivo. Toda empresa u organización, sea en el área de la medicina, los negocios, la química o las ciencias sociales, utiliza un sistema de información.

El término sistema de información suele usarse como sinónimo de sistema de información informático, sin embargo, un sistema de información es “todo aquello que tenga diferentes componentes que funcionen para la recolección y almacenamiento de datos (utilice o no un programa informático)”. (Uriarte, 2021)

2.2.2.1.Actividad de un Sistema de Información

Sistema de información. Conjunto organizado de elementos, que pueden ser personas, datos, actividades o recursos materiales en general. Estos elementos interactúan entre sí para procesar información y distribuirla de manera adecuada en función de los objetivos de una organización.

Un sistema de información realiza cuatro funciones básicas:

- Entrada.
- Procesamiento.
- Almacenamiento.
- Salida de información.

Entrada de la Información

Es el proceso mediante el cual el Sistema de Información toma los datos que requiere para procesar la misma. Las entradas pueden ser manuales o automáticas. Las manuales son aquellas que se proporcionan en forma directa por el usuario, mientras

que las automáticas son datos o información que provienen o son tomados de otros sistemas o módulos. Esto último se denomina interfaces automáticas. Las unidades típicas de entrada de datos a las computadoras son las terminales, las cintas magnéticas, las unidades de diskette, los códigos de barras, los escáner, la voz, los monitores sensibles al tacto, el teclado y el mouse, entre otras.

Procesamiento de Información

Es la capacidad del Sistema de Información para efectuar cálculos de acuerdo con una secuencia de operaciones preestablecida. Estos cálculos pueden efectuarse con datos introducidos recientemente en el sistema o bien con datos que están almacenados. Esta característica de los sistemas permite la transformación de datos fuente en información que puede ser utilizada para la toma de decisiones, lo que hace posible, entre otras cosas, que un tomador de decisiones genere una proyección financiera a partir de los datos que contiene un estado de resultados o un balance general de un año base.

Almacenamiento de Información

El almacenamiento es una de las actividades o capacidades más importantes que tiene una computadora, ya que a través de esta propiedad el sistema puede recordar la información guardada en la sección o proceso anterior. Esta información suele ser almacenada en estructuras de información denominadas archivos. La unidad típica de almacenamiento son los discos magnéticos o discos duros, los discos flexibles o disquete y los discos compactos CD-ROM.

Salida de Información

La salida es la capacidad de un Sistema de Información para sacar la información procesada o bien datos de entrada al exterior. Las unidades típicas de salida son las impresoras, terminales, diskettes, cintas magnéticas, la voz, los graficadores y los plotters, entre otros. Es importante aclarar que la salida de un Sistema de Información puede constituir la entrada a otro sistema o módulo. En este caso, también existe una interface automática de salida. (Alegsa, 2018).

2.2.2.2.Ciclo de Vida de los Sistemas de Información

El método de ciclo de vida para el desarrollo de sistemas es el conjunto de actividades que los analistas, diseñadores y usuarios realizan para desarrollar e implantar un sistema de información.

El método del ciclo de vida para el desarrollo de sistemas consta de 6 fases:

1). Investigación Preliminar: La solicitud para recibir ayuda de un sistema de información puede originarse por varias razones: sin importar cuales sean estas, el proceso se inicia siempre con la petición de una persona.

2). Determinación de los Requerimientos del Sistema: El aspecto fundamental del análisis de sistemas es comprender todas las facetas importantes de la parte de la empresa que se encuentra bajo estudio. Los analistas, al trabajar con los empleados y administradores, deben estudiar los procesos de una empresa para dar respuesta a las siguientes preguntas clave:

3). Diseño del Sistema: El diseño de un sistema de información produce los detalles que establecen la forma en la que el sistema cumplirá con los requerimientos identificados durante la fase de análisis. Los especialistas en sistemas se refieren, con frecuencia, a esta etapa como diseño lógico en contraste con la del desarrollo del software, a la que denominan diseño físico.

4). Desarrollo del Software: Los encargados de desarrollar software pueden instalar software comprobando a terceros o escribir programas diseñados a la medida del solicitante. La elección depende del costo de cada alternativa, del tiempo disponible para escribir el software y de la disponibilidad de los programadores.

Por lo general, los programadores que trabajan en las grandes organizaciones pertenecen a un grupo permanente de profesionales.

5). Prueba de Sistemas: Durante la prueba de sistemas, el sistema se emplea de manera experimental para asegurarse de que el software no tenga fallas, es decir, que funciona de acuerdo con las especificaciones y en la forma en que los usuarios esperan que lo haga.

Se alimentan como entradas conjunto de datos de prueba para su procesamiento y después se examinan los resultados.

6). Implantación y Evaluación: La implantación es el proceso de verificar e instalar nuevo equipo, entrenar a los usuarios, instalar la aplicación y construir todos los archivos de datos necesarios para utilizarla. Una vez instaladas, las aplicaciones se emplean durante muchos años. Sin embargo, las organizaciones y los usuarios cambian con el paso del tiempo, incluso el ambiente es diferente con el paso de las semanas y los meses. (James, 1992)

2.2.3.Control

Control es medir y corregir las actividades de subordinación para asegurarse que los eventos se ajusten a los planes. El control es el procesamiento de verificar el 21 desempeño de distintas áreas o funciones de organización. Administrar significa ejercer la dirección de un proyecto, que se encamina hacia un objetivo, por lo tanto, consiste en planearlo y organizarlo en pasos simultáneos y /o sucesivos, que son necesarios supervisar, para que se desarrollen en el rumbo adecuado hacia el fin. Esa supervisión, es lo que llamamos control administrativo. (Rivera, 2018)

2.3.Requerimientos del Sistema

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

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

2.3.1.Requerimientos Funcionales

Los requisitos funcionales son declaraciones de los servicios que prestará el sistema, en la forma en que reaccionará a determinados insumos. Cuando hablamos de las entradas, no necesariamente hablamos sólo de las entradas de los usuarios. Pueden ser interacciones con otros sistemas, respuestas automáticas, procesos predefinidos. En algunos casos, los requisitos funcionales de los sistemas también establecen explícitamente lo que el sistema no debe hacer.

Es importante recordar esto: un RF puede ser también una declaración negativa. Siempre y cuando el resultado de su comportamiento sea una respuesta funcional al usuario o a otro sistema, es correcto. Y más aún, no sólo es correcto, sino que es necesario definirlo. Y eso nos lleva al siguiente punto.

2.3.2.Requerimientos no Funcionales

Se trata de requisitos que no se refieren directamente a las funciones específicas suministradas por el sistema características de usuario, sino a las propiedades del sistema: rendimiento, seguridad, disponibilidad. En palabras más sencillas, no hablan de “lo que” hace el sistema, sino de “cómo” lo hace. Alternativamente, definen restricciones del sistema tales como la capacidad de los dispositivos de entrada/salida y la representación de los datos utilizados en la interfaz del sistema.

Los requisitos no funcionales se originan en la necesidad del usuario, debido a restricciones presupuestarias, políticas organizacionales, la necesidad de interoperabilidad con otros sistemas de software o hardware, o factores externos tales como regulaciones de seguridad, políticas de privacidad, entre otros. (Blog, 2018)

2.4.Metodología UWE (UML-BASED WEB ENGINEERING)

2.4.1.Introducción

UWE es un proceso del desarrollo para aplicaciones Web enfocado sobre el diseño sistemático, la personalización y la generación semiautomática de escenarios que guíen el proceso de desarrollo de una aplicación Web. UWE describe una metodología de diseño sistemática, basada en las técnicas de UML, la notación de UML y los

mecanismos de extensión de UML.

UWE es una propuesta basada en el proceso unificado y UML pero adaptados a la web. En requisitos separa las fases de captura, definición y validación. Hace además una clasificación y un tratamiento especial dependiendo del carácter de cada requisito. En el marco de UWE es necesario la definición de un perfil UML (extensión) basado en estereotipos con este perfil se logra la asociación de una semántica distinta a los diagramas del UML puro, con el propósito de acoplar el UML a un dominio específico, en este caso, las aplicaciones Web. Entre los principales modelos de UWE podemos citar: el modelo lógico-conceptual, modelo navegacional, modelo de presentación, visualización de Escenarios Web y la interacción temporal.

UWE define vistas especiales representadas gráficamente por diagramas en UML. Además UWE no limita el número de vistas posibles de una aplicación, UML proporciona mecanismos de extensión basados en estereotipos. Estos mecanismos de extensión son los que UWE utiliza para definir estereotipos que son lo que finalmente se utilizarán en las vistas especiales para el modelado de aplicaciones Web. UWE está especializada en la especificación de aplicaciones adaptativas, y por tanto hace especial hincapié en características de personalización, como es la definición de un modelo de usuario o una etapa de definición de características adaptativas de la navegación en función de las preferencias, conocimiento o tareas de usuario.

Además de estar considerado como una extensión del estándar UML, también se basa en otros estándares como por ejemplo: XMI como modelo de intercambio de formato, MOF para la meta-modelado, los principios de modelado de MDA, el modelo de transformación del lenguaje QVT y XML. (Roque, 2013)

2.4.2. Características de la Metodología UWE

Se basa en las características principales siguientes:

- Métodos definidos: pasos definidos para la construcción de cada modelo.
- Notación Estándar: el uso de la metodología UML para todos los modelos.

Especificación de restricciones: recomendables de manera escrita, para que la exactitud en cada modelo aumente.

2.4.3.Actividades del Modelado UWE

Las actividades base de modelado de UWE son el análisis de requerimientos, el modelo conceptual, el modelo navegacional y el modelo de presentación. A estos modelos se pueden sumar otros modelos como lo son el modelo de interacción y la visualización de Escenarios Web.

2.4.4.Fases de la Metodología UWE

A)Fase de Análisis y Especificación de Requisitos: En simple palabras y básicamente, durante esta fase, se adquieren, reúnen y especifican las características funcionales y no funcionales que deberá cumplir la aplicación web.

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

- Diagrama de Casos de Usos
- Diagrama Conceptual
- Diagrama Físico
- Diagrama de Clases
- Modelo Navegacional
- Modelo de Presentación

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

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

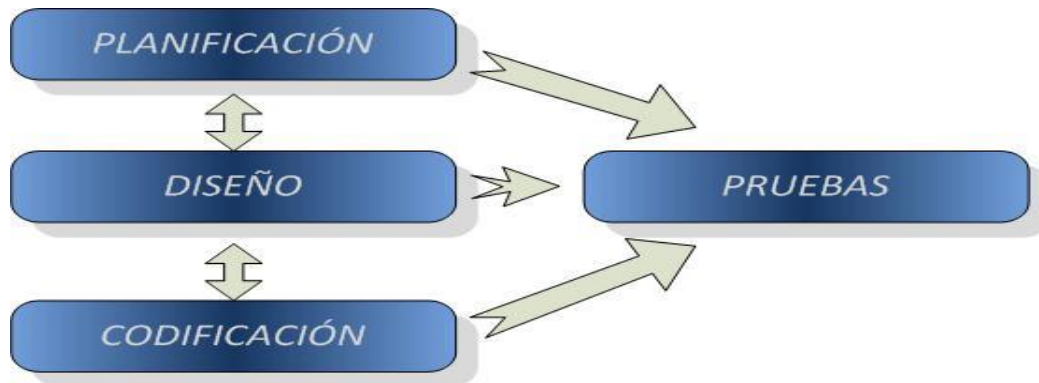
E)Fase de Implementación: es el proceso por el cual los programas desarrollados son transferidos apropiadamente al computador destino, inicializados y eventualmente configurados; todo ello con el propósito de ser ya utilizados por el usuario final.

F)Fase de Mantenimiento: es el proceso de control, mejora y optimización del software ya desarrollado e instalado, que también incluye depuración de errores y

defectos que puedan haberse filtrado de la fase de pruebas de control.

Figura 2.2.

Fases de la Metodología UWE.



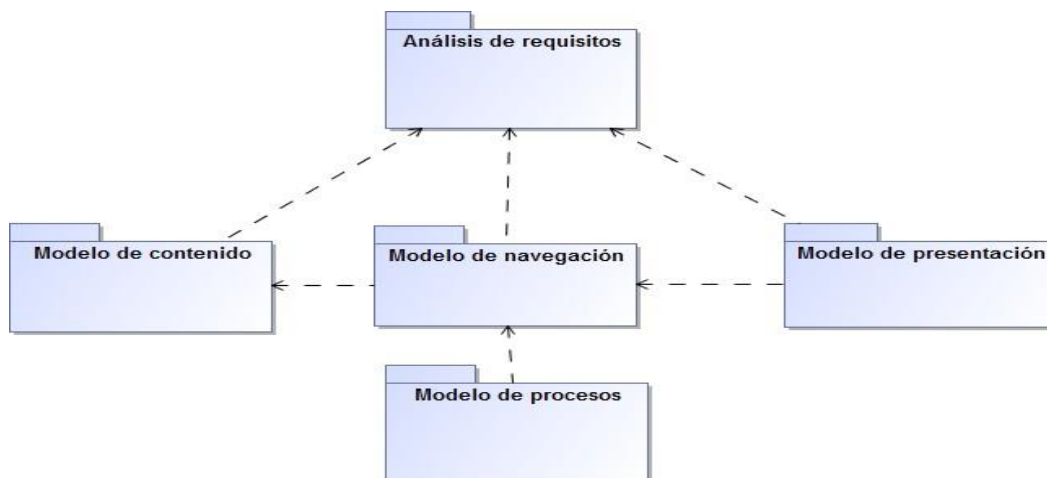
Nota. Se puede observar las Fases de la Metodología UWE.

2.4.5. Modelos de la Metodología UWE

El análisis de requisitos en UWE se modela con casos de uso. Está conformado por los elementos actor y caso de uso. En este sentido, los actores se utilizan para modelar los usuarios de la aplicación Web.

Figura 2.3.

Modelo de la Metodología UWE.



Nota. Como se puede Observar en la Figura 2.3. Tenemos el Modelo de la Metodología UWE.

2.4.5.1. Modelo de Casos de Uso

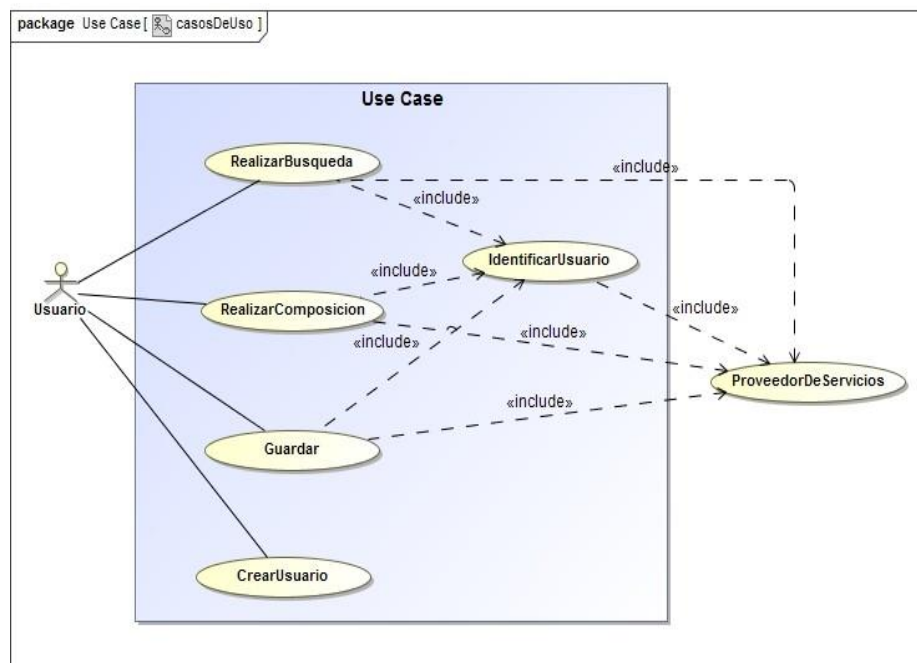
Una de las primeras actividades en la construcción de aplicaciones Web es la identificación de los requisitos, y en UWE se especifican mediante el modelo de requerimientos, que involucra el modelado de casos de uso con UML.

El diagrama de casos de uso está conformado por los elementos actor y caso de uso. Los actores se utilizan para modelar los usuarios de la aplicación Web que para este caso de estudio son los diferentes tipos de usuarios (Chofer, Jefe de grupo, Agente, Administrador) que pueden interactuar con el mismo.

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

Figura 2.4.

Modelo de Casos de Uso.



Nota. Como se puede Observar en la Figura 2.4. Tenemos el Modelo de Casos de Uso para las iteraciones de cada usuario.

El caso de uso "Realizar Búsqueda" es del estereotipo explorar («browsing»). Modela la búsqueda de los objetos de aprendizaje por medio de las características de los objetos y de los usuarios para que el sistema pueda proporcionar una recomendación personalizada.

El caso de uso "Realizar Composición" es del estereotipo procesar («processing»). Según la lista final seleccionada por el usuario, ejecuta el proceso de composición para conformar un nuevo objeto de mayor nivel de instrucción añadiendo cambios a los metadatos si el usuario así lo decide.

El caso de uso "Identificar Usuario" es del estereotipo explorar («browsing»). Ejecuta el proceso de inicio de sesión el cual verifica si el usuario proporcionado existe en el sistema.

El caso de uso "Guardar" es del estereotipo procesar («processing»). Ejecuta la conversión del objeto compuesto al estándar IEEE-LOM y almacena el objeto compuesto en la computadora o en el repositorio para su posterior uso.

El caso de uso "Crear Usuario" es el estereotipo procesar («processing»). Registra los datos de un nuevo usuario que se agrega al sistema, lo que facilita información de su perfil y mejora la personalización de los resultados.

El nivel de detalle y la formalidad de la especificación de requerimientos dependen de los riesgos del proyecto y de la complejidad de la aplicación Web a construir. A menudo una especificación basada solamente en casos de uso no es suficiente.

Siguiendo el principio de usar UML para la especificación hasta donde sea posible, es factible emplear diagramas de actividades en esta fase. Para cada caso de uso descrito para actividades no triviales se puede construir al menos un diagrama de actividad por cada flujo principal de tareas realizadas en orden. Esto con el fin de describir la funcionalidad indicada por el caso de uso correspondiente.

2.4.5.2. Modelo Conceptual

El objetivo del modelo de contenido es proporcionar una especificación visual de la información en el dominio relevante para la aplicación Web.

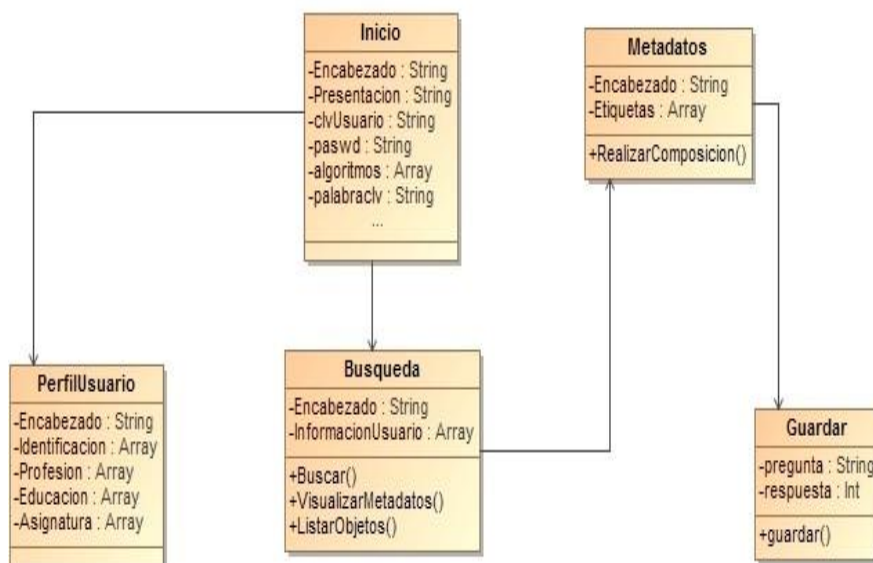
Este es un diagrama UML normal de clases, por ello se debe pensar en las clases que

son necesarias para el caso de estudio presentado.

Se presenta el diagrama de clases para el modelo de contenido. En particular, la información de los usuarios es modelada por la clase "Perfil Usuario" donde se almacenan las propiedades que describen a los diferentes tipos de usuarios.

Figura 2.5.

Modelo del Contenido.



Nota. Como se puede Observar en la Figura 2.5. Tenemos el Modelo del contenido para proporcionar información visual de la aplicación.

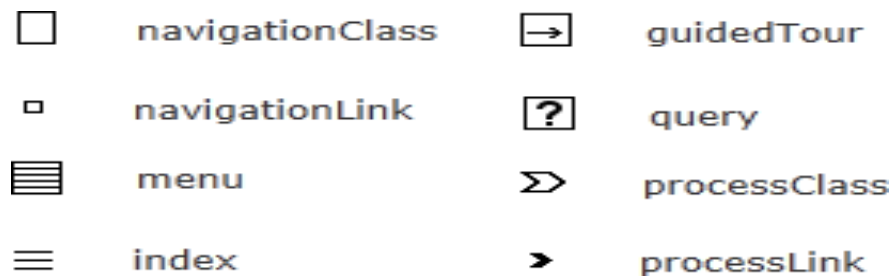
En la clase "Inicio" se modela el inicio de la aplicación web, se almacenan las credenciales y propiedades que sirven para identificar al usuario que quiere iniciar sesión. La clase "Búsqueda" modela la información que el usuario proporciona para realizar una consulta y los métodos que se ejecutan para generar la lista de recomendación, la selección de los objetos y la recuperación de los mismos con sus metadatos. La clase "metadatos" modela las características devueltas por los objetos de aprendizaje que el usuario ha seleccionado y el método de realizar la composición con la selección y los metadatos proporcionados. La clase "guardar" modela las características de almacenamiento del nuevo objeto compuesto.

2.4.5.3. Modelo de Navegación

En una aplicación para la Web es útil saber cómo están enlazadas las páginas. Ello significa que se requiere un diagrama de navegación con nodos y enlaces. Este diagrama se modela con base en el análisis de los requisitos y el modelo de contenido. UWE provee diferentes estereotipos para el modelado de navegación, Se presentan los usados en este caso de estudio y seguidamente se da una descripción de cada uno de ellos.

Figura 2.6.

Estereotipos de Estructura de Navegación.



Nota. Como se puede Observar en la Figura 2.6. Tenemos la estructura de navegación de estereotipos.

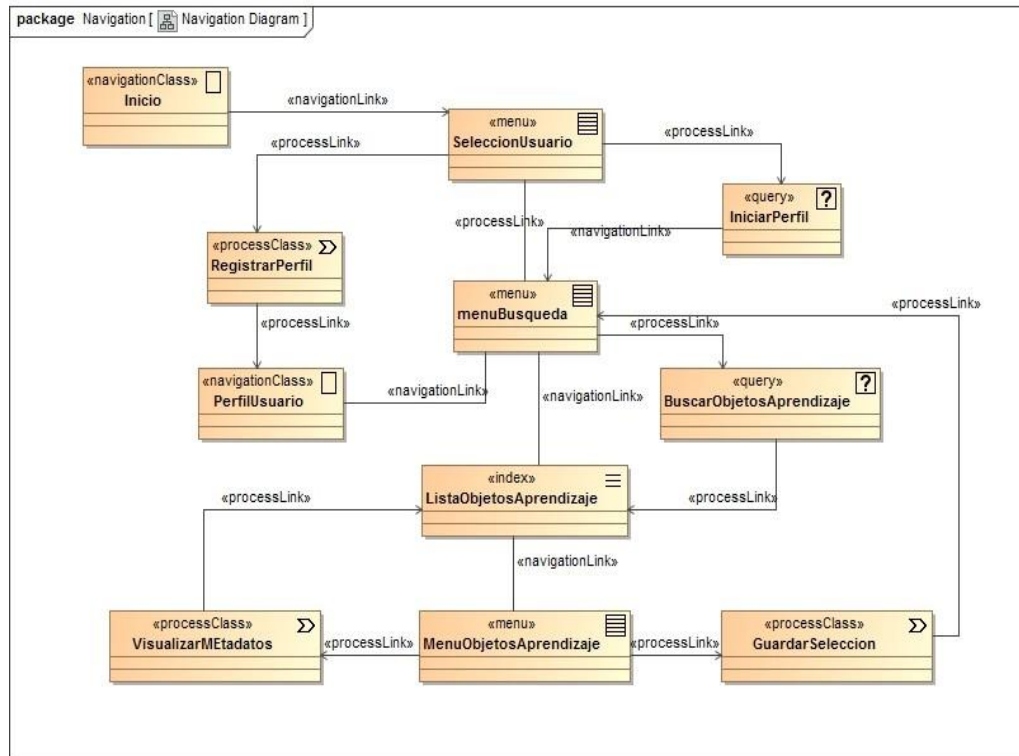
Las clases de navegación («navigationClass») representan nodos navegables de la estructura de hipertexto; los enlaces de navegación («navigationLink») muestran vínculos directos entre las clases de navegación; las rutas alternativas de navegación son manejadas por menú («menu»). Los accesos se utilizan para llegar a múltiples instancias de una clase de navegación («index» o «guidedTour») o para seleccionar los elementos («query»). Las clases de procesos («processClass») forman los puntos de entrada y salida de los procesos de negocio en este modelado y la vinculación entre sí y a las clases de navegación se modela por enlaces de procesos («processLink»).

Las clases de navegación "Inicio y PerfilUsuario" representan nodos navegables de la estructura de hipertexto y se consideran relevantes para la navegación.

Los enlaces de navegación "navigationLink" y "processLink" muestran vínculos directos entre las clases de navegación y representan posibles pasos a seguir por el usuario y, por lo tanto, estos vínculos tienen que ser dirigidos.

Figura 2.7.

Modelo de Navegación.



Nota. Como se puede Observar en la Figura 2.7. Tenemos el Modelo de Navegación para tener diferentes iteraciones que representa el menú.

La navegación por diferentes alternativas es representada por las clases «menú» ("SeleccionUsuario, MenuBusqueda y MenuObjetosAprendizaje") que se añaden a cada clase de navegación que tiene más de una asociación saliente.

Las primitivas de acceso «índice» como es "ListaObjetosAprendizaje" se utilizan para llegar a múltiples instancias de una clase de navegación o para seleccionar los elementos con los tipos «query» como "IniciarPerfil y BuscarObjetosAprendizaje", este tipo de clase se debe agregar entre dos clases de navegación cada vez que la multiplicidad de la meta final de su asociación de enlace sea mayor que 1. Las entradas y salidas de las clases "RegistrarPerfil,

VisualizarMetadatos y GuardarSeleccion" son modeladas por las clases «process».

Es así que desde la página de Inicio un usuario puede, por medio de "SeleccionUsuario", tener una representación personalizada según sea su tipo de

usuario con el que accede al sistema. Puede optar por usar "IniciarPerfil" para consultar si existe su clave de usuario proporcionada, o por "registrarPerfil" que inicia el proceso de registro del nuevo usuario. El usuario que ingresa a la aplicación proporciona palabras clave para "BuscarObjetosAprendizaje" que arroja una "ListaObjetosAprendizaje" para la selección por parte del usuario. De los objetos que son seleccionados en un "MenuObjetosAprendizaje", el usuario puede "VisualizarMetadatos" de los objetos que son candidatos a conformar un nuevo Objeto de Aprendizaje de nivel superior de complejidad para "GuardarSeleccion".

2.4.5.4. Modelo de Procesos

La estructura de navegación puede ser extendida mediante clases de procesos que representan la entrada y la salida de procesos de negocio. El modelo del proceso representa el aspecto que tienen las acciones de las clases de proceso. En este modelo se tienen dos tipos de modelos:

Modelo de estructura del proceso, que describe las relaciones entre las diferentes clases de proceso, y Modelo de flujo del proceso, que especifica las actividades conectadas con cada « processClass». A continuación se describen cada uno de ellos:

Modelo de estructura del proceso. Es representado por un diagrama de clases donde se describen las relaciones entre las diferentes clases de proceso. Presenta la aplicación del modelo para el caso de estudio analizado.

Modelo del flujo del proceso. Siguiendo el principio de la utilización de UML se han refinado los requisitos con los diagramas de actividad UML. Los diagramas de actividades incluyen actividades, actores responsables de estas actividades (opcional) y elementos de flujo de control. Ellos pueden ser enriquecidos con flujos de objetos que muestran objetos relevantes para la entrada o salida de esas actividades.

Estos diagramas representan el flujo del proceso, describiendo el comportamiento de una clase de proceso. Se ilustra el diagrama de actividad para el proceso

"Inicio". El diagrama muestra que al generar la página de inicio el usuario puede optar por dos opciones:

Proporcionar su clave de usuario y contraseña si es un usuario registrado, activar el

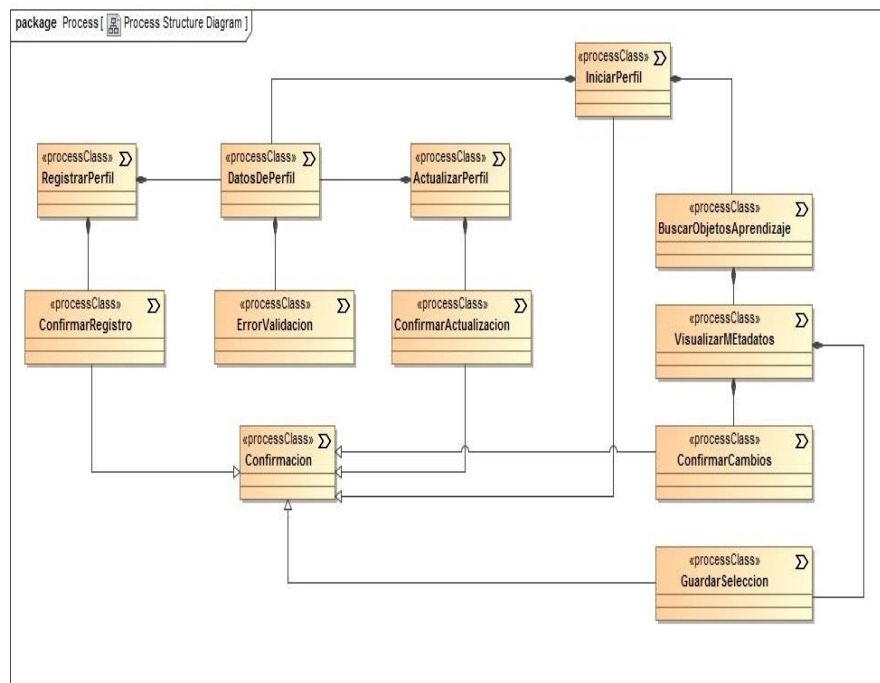
botón para registrarse como nuevo usuario.

En el caso de la primera opción, el sistema debe validar al usuario proporcionando el acceso a la búsqueda de objetos a aquellos usuarios que sean confirmados como válidos o mostrando un mensaje de error para el caso contrario. En la segunda opción, se debe activar el proceso de registro para capturar el perfil del nuevo usuario.

Se ilustra el diagrama de actividad para el proceso "Buscar". El diagrama muestra que se activa con el botón buscar y el usuario proporciona las palabras clave para iniciar la búsqueda. La aplicación regresa una lista de objetos de aprendizaje candidatos a ser seleccionados por el usuario. Si existe la información se recupera la misma desde el repositorio, en caso contrario se regresa a la página de búsqueda. Si la información listada es de interés para el usuario, este selecciona la misma, en caso contrario cambia sus parámetros de búsqueda.

Figura 2.8.

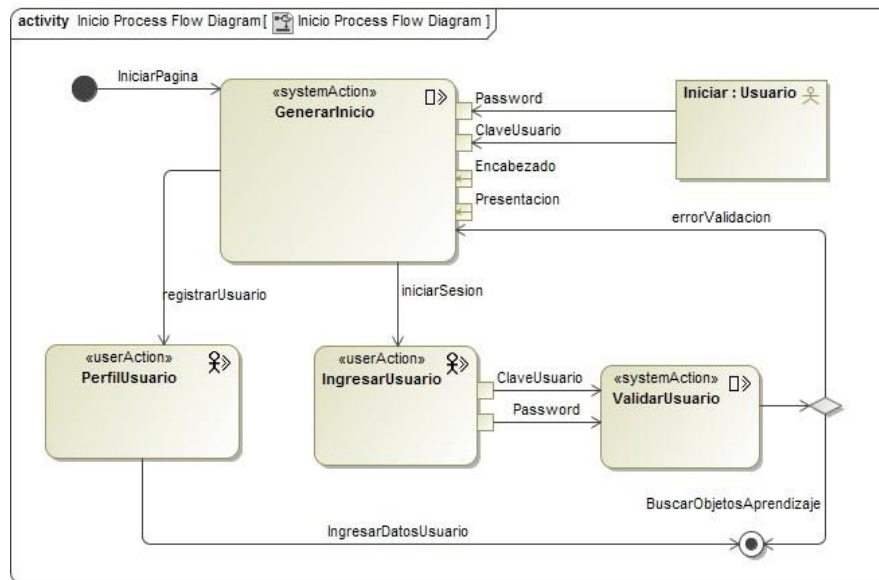
Modelo de Procesos.



Nota. Como se puede Observar en la Figura 2.8. Tenemos el Modelo de Proceso para el usuario.

Figura 2.9.

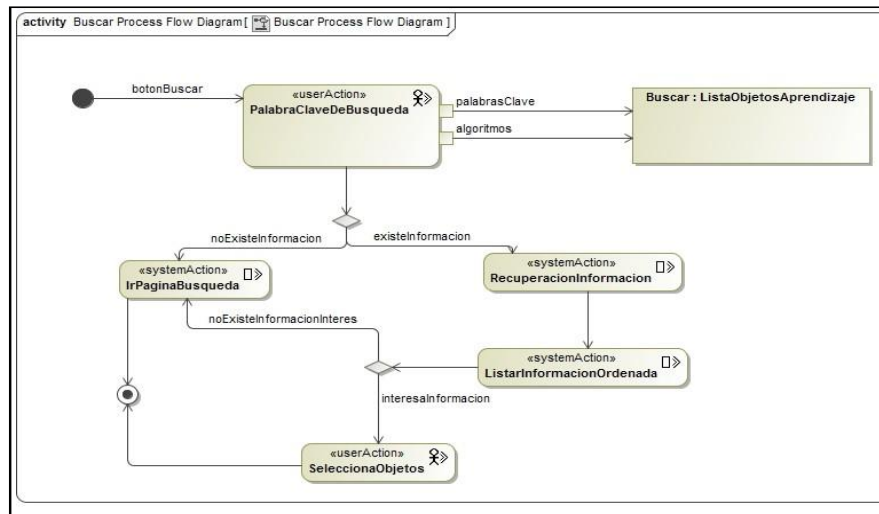
Flujo de Proceso: Inicio.



Nota. Como se puede Observar en la Figura 2.9. Tenemos el Modelo de Procesos: Inicio que representa el inicio del usuario.

Figura 2.10.

Flujo de Proceso: Buscar.



Nota. Como se puede Observar en la Figura 2.10. Tenemos el Modelo de Procesos: Buscar, que representada la búsqueda del usuario.

2.4.5.5. Modelo de Presentación

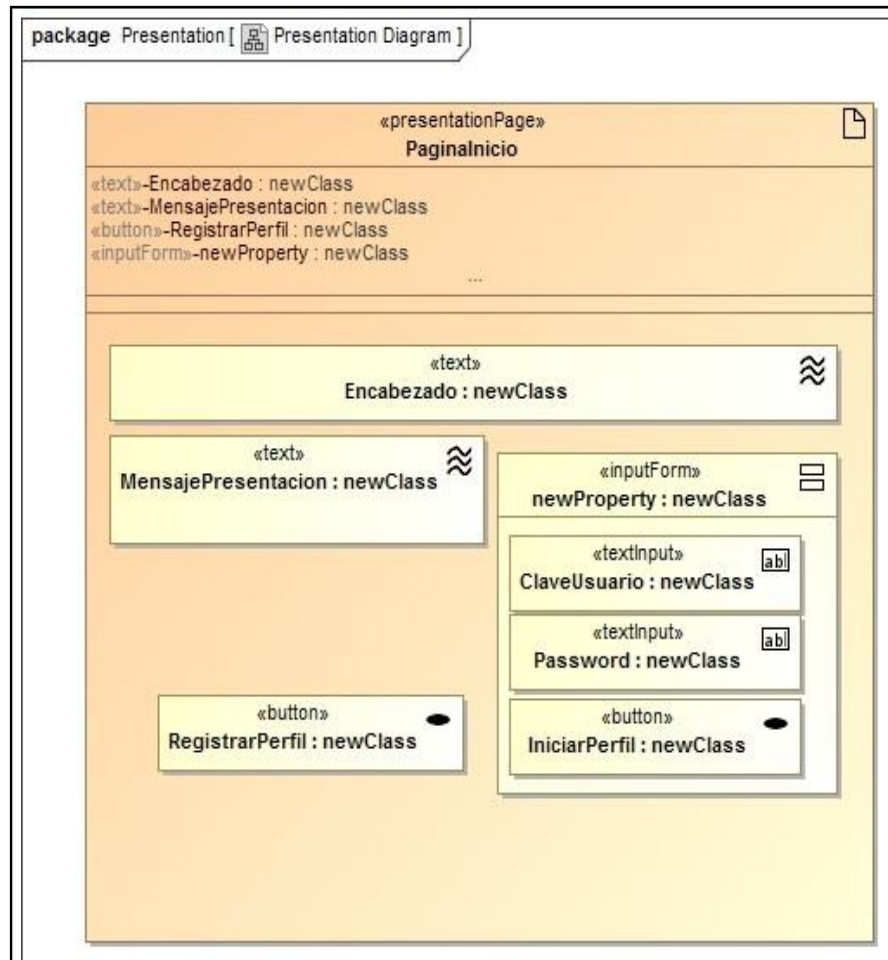
El modelo de presentación ofrece una visión abstracta de la interfaz de usuario de una aplicación Web. Se basa en el modelo de navegación y en los aspectos concretos de la interfaz de usuario (IU). Describe la estructura básica de la IU, es decir, ¿qué elementos de interfaz de usuario (por ejemplo, texto, imágenes, enlaces, formularios) se utilizan para presentar los nodos de navegación? Su ventaja es que es independiente de las técnicas actuales que se utilizan para implementar un sitio Web, lo que permite a las partes interesadas discutir la conveniencia de la presentación antes de que realmente se aplique.

Una clase de presentación está compuesta de elementos de IU como texto («text»), enlaces («anchor»), botones («button»), imágenes («image»), formularios («form») y colecciones de enlaces («anchored collection»). La figura 2.11 muestra un ejemplo de la clase de presentación para la clase de navegación Inicio.

En la figura 2.11 se modela la página de presentación "PaginaInicio". Existe una representación de texto para el encabezado y un mensaje de presentación. Modela también un formulario de entrada para que el usuario introduzca clave y contraseña, así como los botones de "iniciarperfil" y "registrarPerfil". Usualmente la información de varios nodos de aplicación es presentada en una página Web, la cual es modelada por páginas en UWE, por ejemplo, se tiene una («presentationPage»). Las páginas de presentación también pueden contener grupos de presentación («presentationGroup»), grupos de presentación iterativos («iteratedPresentationGroup»), y presentaciones alternativas («presentationAlternative»), por ejemplo ajustar la interfaz al dispositivo utilizado para ejecutar la aplicación. Un grupo de presentación puede contener a si mismo grupos de presentación y clases de presentación.

Figura 2.11.

Modelo de Presentación Inicio.

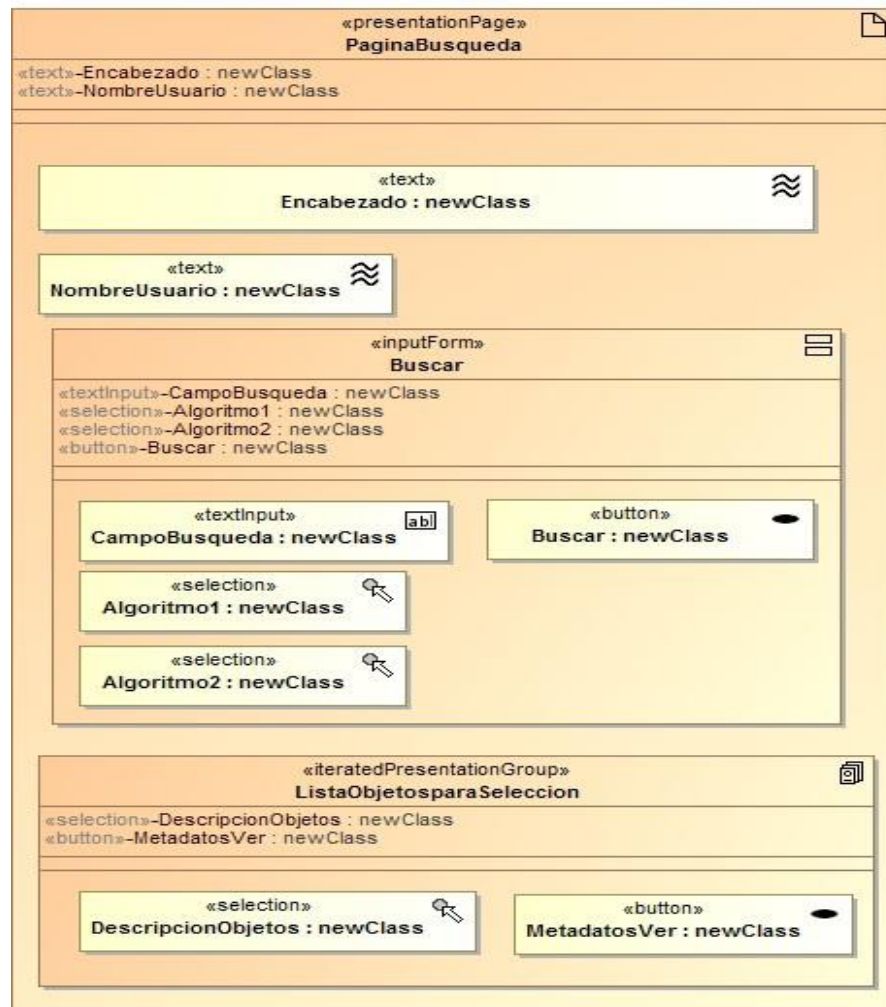


Nota. Como se puede Observar en la Figura 2.11. Tenemos el Modelo de Presentación Inicio, representa el inicio del sistema para los usuarios.

En la figura 2.12 se modela la página de presentación "paginaBusqueda" donde se representa como texto un encabezado y el nombre del usuario. Existe un formulario donde se puede introducir las palabras clave de búsqueda así como seleccionar los algoritmos que se pueden aplicar. Esta página de presentación contiene un grupo de presentación para modelar las listas de objetos candidatos a la composición y los botones de buscar y ver metadatos. (Nieves Guerrero, Ucan Pech, & Mendez Dominguez, 2014).

Figura 2.12.

Página de Modelo de Presentación de Búsqueda.



Nota. Como se puede Observar en la Figura 2.12. Tenemos el Modelo de Presentación Búsqueda: para la búsqueda de cada usuario.

2.5. Normas ISO 9126

Las métricas de calidad de software proporcionan de cómo se ajusta el software, a los requerimientos explícitos del cliente. Por lo cual se aplicara la norma ISO 9126 lo cual es un estándar internacional para la evaluación de software, que establece que cualquier componente de la calidad pueda ser descrito por las características de Funcionalidad, Confiabilidad, Mantenibilidad, Usabilidad y Portabilidad.

2.5.1. Características

La norma ISO-9126 fue desarrollada en un intento de identificar los atributos clave de calidad para el software. La tabla 1 muestra las preguntas centrales que atiende cada una de estas características. (Cruz Perez, 2009)

Tabla 2.1

Características de Calidad.

Características de Calidad	Preguntas
Funcionalidad	¿Las funciones y propiedades satisfacen las necesidades explícitas e implícitas?
Confiabilidad	¿Puede mantener el nivel de rendimiento, bajo ciertas condiciones y por cierto tiempo?
Usabilidad	¿El software es fácil de usar y de aprender?
Eficiencia	¿Es rápido y minimalista en cuanto al uso de recursos?
Mantenibilidad	¿Es fácil de modificar y verificar?
Portabilidad	¿Es fácil de transferir de un ambiente a otro?

Nota. Características de la norma ISO-9126.

El estándar ISO-9126 establece que cualquier componente de la calidad del software puede ser descrito en términos de seis atributos cada una de las cuales se detalla a través de un conjunto de sub-atributos que permiten analizar y profundizar en la evaluación de la calidad de productos de software. A continuación se explican estos atributos detalladamente:

Funcionalidad: Conjunto de atributos que se relacionan con la existencia de un conjunto de funciones y sus propiedades específicas. Las funciones son aquellas que satisfacen lo indicado o implica necesidades.

- **Idoneidad:** Se enfoca a evaluar si el software cuenta con un conjunto de funciones

apropiadas para efectuar las tareas que fueron especificadas en su definición.

- **Exactitud:** Permite evaluar si el software presenta resultados o efectos acordes a las necesidades para las cuales fue creado.
- **Interoperabilidad:** Permite evaluar la habilidad del software de interactuar con otros sistemas previamente especificados.
- **Seguridad:** Se refiere a la habilidad de prevenir el acceso no autorizado, ya sea accidental o premeditado, a los programas y datos.
- **Conformidad:** Evalúa si el software se adhiere a estándares, convenciones o regulaciones en leyes y prescripciones similares.

Confiabilidad: Conjunto de atributos relacionados con la capacidad del software de mantener su nivel de prestación bajo condiciones establecidas durante un período de tiempo establecido.

- **Madurez:** Permite medir la frecuencia de falla por errores en el software.
- **Recuperación:** Se refiere a la capacidad de restablecer el nivel de operación y recobrar los datos que hayan sido afectados directamente por una falla, así como al tiempo y el esfuerzo necesarios para lograrlo.
- **Tolerancia a fallos:** Se refiere a la habilidad de mantener un nivel específico de funcionamiento en caso de fallas del software o de cometer infracciones de su interfaz específica.

Usabilidad: Conjunto de atributos relacionados con el esfuerzo necesitado para el uso, y en la valoración individual de tal uso, por un establecido o implicado conjunto de usuarios.

- **Comprensión:** Se refiere al esfuerzo requerido por los usuarios para reconocer la estructura lógica del sistema y los conceptos relativos a la aplicación del software.
- **Facilidad de Aprender:** Establece atributos del software relativos al esfuerzo que los usuarios deben hacer para aprender a usar la aplicación.
- **Operatividad:** Agrupa los conceptos que evalúan la operación y el control del sistema.

Eficiencia: Conjunto de atributos relacionados con la relación entre el nivel de desempeño del software y la cantidad de recursos necesitados bajo condiciones establecidas.

- Comportamiento en el tiempo: Comportamiento en el tiempo: Atributos del software relativos a los tiempos de respuesta y de procesamiento de los datos.
- Comportamiento de recursos Comportamiento de recursos recursos: Atributos del software relativos a la cantidad de recursos usados y la duración de su uso en la realización de sus funciones.

Mantenibilidad: Conjunto de atributos relacionados con la facilidad de extender, modificar o corregir errores en un sistema software.

- Estabilidad: Capacidad del software de tener un desempeño normal a pesar de hacerse modificaciones.
- Facilidad de análisis: Relativo al esfuerzo necesario para diagnosticar las deficiencias o causas de fallas, o para identificar las partes que deberán ser modificadas.
- Facilidad de cambio: Capacidad del software para que alguna de sus partes pueda ser modificado.
- Facilidad de pruebas: Facilidad de pruebas: Capacidad del que tiene el software para que la modificación pueda ser válida.

Portabilidad: Conjunto de atributos relacionados con la capacidad de un sistema software para ser transferido desde una plataforma a otra.

- Adaptabilidad: Evalúa la oportunidad para adaptar el software a diferentes ambientes sin necesidad de aplicarle modificaciones.
- Facilidad de Instalación: Es el esfuerzo necesario para instalar el software en un ambiente determinado.
- Cumplimiento: Permite evaluar si el software se adhiere a estándares o convenciones relativas a portabilidad.
- Capacidad de reemplazo: Se refiere a la oportunidad y el esfuerzo usado en sustituir el software por otro producto con funciones similares.

2.5.2. Factores de Calidad

El estándar ISO 9126 ha sido desarrollado en un intento de identificar los atributos clave de calidad para el software. El estándar identifica seis atributos clave de calidad:

Funcionalidad. El grado en que el software satisface las necesidades indicadas por los siguientes subatributos: idoneidad, corrección, interoperatividad, conformidad y seguridad.

Confiabilidad. Cantidad de tiempo que el software está disponible para su uso. Está referido por los siguientes subatributos: madurez, tolerancia a fallos y facilidad de recuperación.

Usabilidad. Grado en que el software es fácil de usar. Viene reflejado por los siguientes subatributos: facilidad de comprensión, facilidad de aprendizaje y operatividad.

Eficiencia. Grado en que el software hace Óptimo el uso de los recursos del sistema. Está indicado por los siguientes subatributos: tiempo de uso y recursos utilizados.

Facilidad de Mantenimiento. La facilidad con que una modificación puede ser realizada. Está indicada por los siguientes subatributos: facilidad de análisis, facilidad de cambio, estabilidad y facilidad de prueba.

Portabilidad. La facilidad con que el software puede ser llevado de un entorno a otro. Está referido por los siguientes subatributos: facilidad de instalación, facilidad de ajuste, facilidad de adaptación al cambio. (Chuquimia)

2.5.3. Ventajas y Desventajas Normas ISO 9126

Ventajas

Es un modelo de corte internacional pero adaptado al caso colombiano y latinoamericano.

La terminología es clara y precisa lo que hace que sea más comprensible para todos los actores del proceso.

Involucra la utilización de la norma ISO.

Introduce un nuevo concepto que es la calidad del uso que tiene en cuenta lo más importante para la gestión de la calidad que es la opinión del usuario.

Se podría utilizar no para uno, si no para varios proyectos.

Desventajas

Como en el caso de todos los modelos implicados, se necesita de muchas métricas lo que requiere de mayor esfuerzo, tiempo y costo.

2.5.4. Formulas Normas ISO 9126

2.5.4.1. Funcionabilidad

Número de Entradas de Usuario: Cuenta con distintos entrada para cada usuario que proporciona la aplicación. Las entradas de los usuarios se diferencian mediante peticiones, las cuales se cuentan de forma separada.

Número de Salidas de Usuario: Cuenta con distintos salidas para cada usuario que proporciona la aplicación. Las salidas se refieren a informes. Planillas, mensajes de error.

Número de Petición de Usuario: Una petición está definida como una entrada interactiva que resultado de la generación de algún tipo de respuesta en forma de salida interactiva.

Número de Archivos de Usuario: Se cuenta cada archivo lógico maestro, tablas existentes en la base de datos.

Número de Interfaces Externas de Usuario: Se cuentan todas las interfaces muy legibles por la máquina.

Para calcular la Siguiete ecuación se usa la siguiente formula:

Ecuación 1.

$$PF = Cuenta\ total * (confiabilidad\ proyecto + error\ min * \sum Fi)$$

Ecuación 2.

$$Funcionalidad = \left(\frac{PF}{PF_{maximo}} \right)$$

Donde:

PF: Medida de Funcionalidad.

Máximo: Medida de Funcionalidad con su valor mínimo.

Cuenta Total: Es la suma de los siguientes datos: Número de entradas, número de salidas, número de peticiones, número de archivo y número de interfaces externas.

Confiabilidad del Proyecto: Varía del 1% al 100%.

Error min: Error mínimo aceptable de complejidad. $\sum Fi$: Son los valores de ajuste de complejidad, donde $(1 \leq i \leq 14)$.

2.5.4.2. Confiabilidad

Un conjunto de atributos relacionados con la capacidad del software de mantener su nivel de prestación bajo condiciones establecidas durante un período establecido.

- Madurez
- Recuperabilidad
- Tolerancia a fallos
- Cumplimiento de Fiabilidad

Se tienen en cuenta aspectos como la capacidad y facilidad de recuperación, la mitigación de fallos, cantidad de tiempo que el software está disponible para su uso y la tolerancia. Por lo que esto tiene en cuenta todo lo relacionado a los fallos que podría dar el producto de software.

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

Ecuacion3.

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

Donde:

f: Funcionalidad del sistema.

μ : Es la probabilidad de error que puede tener el sistema.

t: Tiempo de duración de gestión en el sistema.

Se observa el trabajo del sistema hasta que produce una falla en el instante T, el cual

se aproxima a una variable aleatoria continua.

$P(T \leq t) = F(t)$ Probabilidad de fallas.

$P(T \leq t) = 1 - F(t)$ Probabilidad del trabajo de tiempo en el que se ejecuta y se obtiene muestras sin fallas.

Para lo que consideramos un periodo de 20 días como tiempo de prueba donde se define que cada 10 ejecuciones se presenta 1 falla conociendo la funcionalidad de 93.6% del sistema.

2.5.4.3. Usabilidad

Un conjunto de atributos relacionados con el esfuerzo necesario para su uso, y en la valoración individual de tal uso, por un establecido o implicado conjunto de usuarios.

- Aprendizaje
- Comprensión
- Operatividad
- Atractividad

Mide el grado en que el software es fácil de usar, qué tan intuitivo es, el manejo que el usuario le da al sistema y si este presenta menús sencillos, lectura de textos ágil, cuenta con funciones de forma clara y puntual, entre otros.

Ecuación 4.

$$FU = \left[\sum \left(\frac{x_i}{n} \right) * 100 \right]$$

Donde:

X_i : Es la sumatoria de valores

n : Es el número de preguntas

2.5.4.4. Mantenibilidad

Conjunto de atributos relacionados con la facilidad de extender, modificar o corregir errores en un sistema software.

- Estabilidad
- Facilidad de Análisis

- Facilidad de Cambio
- Facilidad de Pruebas

La facilidad con que una modificación puede ser realizada, la capacidad para hacerle pruebas de rendimiento, regresión, accesibilidad, inspección de código y toda la ingeniería de requerimientos. Esto hace que el producto de software sea escalable ya que es posible hacerle mejoras constantes sin que este no tenga ningún problema al querer realizarle alguna modificación o incorporación.

Para hallar mantenibilidad del sistema se utiliza el índice de madurez de software (IMS):

Ecuación 5.

$$IMS = \frac{Mt - (Fc + Fa + FE)}{Mt}$$

Donde:

- Mt:** Numero de módulos total de la versión actual = 8
- Fc:** Numero de módulos de la versión actual que se cambiaron. = 2
- Fa:** Numero de módulos de la versión actual que se añadieron. = 0
- FE:** Numero de módulos de la versión anterior que se eliminaron en la versión = 0

2.5.4.5.Portabilidad

La facilidad con que el software puede ser llevado de un entorno a otro. Proporciona facilidad de instalación, facilidad de ajuste, facilidad de adaptación al cambio y otros aspectos que lo hacen un sistema que garantiza portabilidad.

El sistema actual está en plataforma de Windows ejecutable desde cualquier plataforma debido a su diseño adaptable el único requisito es que el dispositivo cuente con internet y un navegador.

El presente proyecto está diseñado en un entorno de acceso vía web mide la portabilidad en: lado del servidor y lado del cliente, la portabilidad del software se enfoca en tres aspectos:

- Hardware del servidor
- Sistema operativo del servidor

- Software del servidor

Fórmula para hallar la Portabilidad:

Ecuación 6.

$$P=1-EP/EI$$

Dónde:

- P = Portabilidad
- EP = Número de días para portar el sistema
- EI = Número de días para implementar el sistema
- EP=0.5 días
- EI= 2,5 días

2.6.Modelo de Estimación de Costo de Software

2.6.1.COCOMO

El Modelo Constructivo de Costos (o COCOMO, por su acrónimo del inglés Constructive Cost Model) (Boehm, 1981) afirma:

Es un modelo matemático de base empírica utilizado para estimación de costos de software. Incluye tres submodelos, cada uno ofrece un nivel de detalle y aproximación, cada vez mayor, a medida que avanza el proceso de desarrollo del software: básico, intermedio y detallado.

Y tiene las siguientes características generales:

Características generales

Pertenece a la categoría de modelos estimadores basados en estimaciones matemáticas. Está orientado a la magnitud del producto final, midiendo el "tamaño" del proyecto, en función de la cantidad de líneas de código, principalmente.

Se presentan tres niveles: básico, intermedio y detallado.

Modelos de estimación

Las ecuaciones que se utilizan en los tres modelos son:

En Persona – Mes

Ecuación 1.

$$E = a (kl^{(b)} * m^{(x)})$$

En Meses:

Ecuación 2.

$$Tdev = c (E^d)$$

En Personas:

Ecuación 3.

$$P = E / Tdev$$

Donde:

E: es el esfuerzo requerido por el proyecto, en persona-mes

Tdev: es el tiempo requerido por el proyecto, en meses

P: es el número de personas requerido por el proyecto

a, b, c y d son constantes con valores definidos en una tabla, según cada submodelo

Kl: es la cantidad de líneas de código, en miles.

m(X): Es un multiplicador que depende de 15 atributos.

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

A.Modo Orgánico: un pequeño grupo de programadores experimentados desarrollan software en un entorno familiar. El tamaño del software varía desde unos pocos miles de líneas (tamaño pequeño) a unas decenas de miles (medio).

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

C.Modo Rígido o Empotrado: el proyecto tiene fuertes restricciones, que pueden estar relacionadas con la funcionalidad y/o pueden ser técnicas. El problema a resolver es único y es difícil basarse en la experiencia, puesto que puede no haberla.

Modelo Básico

Se utiliza para obtener una primera aproximación rápida del esfuerzo, 2 y hace

uso de lasiguiente tabla de constantes para calcular distintos aspectos de costes:

Tabla 2.2

Coficiente del Modelo Básico.

MODO	A	B	C	D
Orgánico	2.40	1.05	2.50	0.38
Semi – Orgánico	3.00	1.12	2.50	0.35
Empotrado	3.60	1.20	2.50	0.33

Nota. Distintos aspectos de coste de COCOMO. (Boehm 1981)

Estos valores son para las fórmulas:

Personas necesarias por mes para llevar adelante el proyecto:

Ecuación 4.

$$MM = a * (KI^b)$$

Tiempo de desarrollo del proyecto:

Ecuación 5.

$$TDEV = c * (MM^d)$$

Personas necesarias para realizar el proyecto:

Ecuación 6.

$$CosteH = MM / TDEV$$

Costo total del proyecto:

Ecuación 7.

$$CosteM = CosteH * \text{Salario medio entre los programadores y analistas}$$

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

Modelo Intermedio

Este añade al modelo básico quince modificadores opcionales para tener en cuenta

en el entorno de trabajo, incrementando así la precisión de la estimación.

Para este ajuste, al resultado de la fórmula general se lo multiplica por el coeficiente surgido de aplicar los atributos que se decidan utilizar.

Los valores de las constantes a reemplazar en la fórmula son:

Tabla 2.3

Coficiente del Modelo Intermedio COCOMO.

MODO	A	B
Orgánico	3.20	1.05
Semi – Orgánico	3.00	1.12
Empotrado	2.80	1.20

Nota. Modelo intermedio de COCOMO. (Boehm 1981).

Se puede observar que los exponentes son los mismos que los del modelo básico, confirmando el papel que representa el tamaño; mientras que los coeficientes de los modos orgánico y rígido han cambiado, para mantener el equilibrio alrededor del semilibre con respecto al efecto multiplicador de los atributos de coste.

Atributos

Cada atributo se cuantifica para un entorno de proyecto. La escala es muy baja - bajo - nominal - alto - muy alto - extremadamente alto. Dependiendo de la calificación de cada atributo, se asigna un valor para usar de multiplicador en la fórmula. El significado de los atributos es el siguiente, según su tipo:

- **De Software**

RELY: garantía de funcionamiento requerida al software. Indica las posibles consecuencias para el usuario en el caso que existan defectos en el producto. Va desde la sola inconveniencia de corregir un fallo (muy bajo) hasta la posible pérdida de vidas humanas, extremadamente alto, software de alta criticidad.

DATA: tamaño de la base de datos en relación con el tamaño del programa. El valor del modificador se define por la relación: D/K , donde D corresponde al tamaño de la base de datos en bytes y K es el tamaño del programa en cantidad de líneas de código.

CPLX: representa la complejidad del producto.

- **De Hardware**

TIME: limitaciones en el porcentaje del uso de la CPU. **STOR:** limitaciones en el porcentaje del uso de la memoria. **VIRT:** volatilidad de la máquina virtual.

TURN: tiempo de respuesta requerido.

- **De Personal**

ACAP: calificación de los analistas.

AEXP: experiencia del personal en aplicaciones similares.

PCAP: calificación de los programadores.

VEXP: experiencia del personal en la máquina virtual.

LEXP: experiencia en el lenguaje de programación a usar.

- **De Proyecto**

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

Modelo Detallado

Presenta principalmente dos mejoras respecto al anterior:

Los factores correspondientes a los atributos son sensibles o dependientes de la fase sobre la que se realizan las estimaciones. Aspectos tales como la experiencia en la aplicación, utilización de herramientas de software, etc., tienen mayor influencia en unas fases que en otras, y además van variando de una etapa a otra. (Gómez, López, Migani, & Otazú, 2020)

Establece una jerarquía de tres niveles de productos, de forma que los aspectos que representan gran variación a bajo nivel, se consideran a nivel módulo, los que representan pocas variaciones, a nivel de subsistema; y los restantes son considerados a nivel sistema.

2.7.Seguridad

La seguridad por la que viaja la información a través del internet sin duda es muy importante, ya que nos da tranquilidad y confianza al saber que datos personales y confidenciales como los datos de la tarjeta de crédito, cuentas bancarias, contraseñas, entre otros tipos de información, están siendo cifradas y hay muy pocas, pero mínimas probabilidades de que sean descifradas por algún hacker y quede toda la data expuesta.

Entonces podemos decir que la encriptación es un proceso en donde uno o varios archivos son codificados a través de un algoritmo que modifica la información original y hace imposible su lectura a menos que cuentes con la autorización o mejor dicho con la llave correspondiente.

2.7.1. Encriptación y Contraseñas en PHP

Desde el principio PHP ha sido un lenguaje de programación para la construcción de sitios web. Esa idea permanece en el núcleo del lenguaje, y por eso es tan popular para la construcción de aplicaciones web. Cuando se creó en los años 90, el término aplicación web no existía aún, por lo que la protección de contraseñas para cuentas de usuarios no era algo en lo que estuviera centrado.

Han pasado muchos años desde entonces y actualmente es impensable una aplicación web que no proteja las cuentas de los usuarios con contraseñas. Es fundamental para cualquier programador hacer que estas contraseñas tengan una encriptación segura y eficiente. PHP 5.5 añadió una nueva librería llamada Hash de contraseñas para la encriptación de contraseñas, con funciones que facilitan la tarea y utilizan los últimos métodos más eficaces.

1. La importancia de los hashes seguros

Siempre hay que guardar las contraseñas encriptadas mediante un algoritmo de encriptación como el algoritmo hashing para hacer imposible a alguien que acceda a una base de datos conseguir averiguar la contraseña. Esto no es sólo para proteger a los usuarios frente a algún atacante sino también frente a los propios empleados de la aplicación.

Mucha gente utiliza las mismas contraseñas para muchas aplicaciones web. Si alguien accede a la dirección de email y contraseña de un usuario, probablemente pueda hacerlo en muchas otras aplicaciones.

Los hashes no se crean iguales, se emplean algoritmos muy distintos para crear un hash. Los dos más usados en el pasado son MD5 y SHA-1. Los ordenadores de hoy en día pueden crackear fácilmente estos algoritmos. Dependiendo de la complejidad y longitud de la contraseña, se puede crackear en menos de una hora con los dos algoritmos nombrados (los ratios son 3650 millones de cálculos por segundo con MD5 y 1360 millones por segundo con SHA-1).

Por eso es importante usar algoritmos complejos. Si el hash es más largo reduce el riesgo de colisiones entre contraseñas (dos frases generando el mismo hash), pero también conviene que la aplicación se tome el tiempo necesario para generar el hash. Esto es porque el usuario apenas notará un segundo o dos más de tiempo de carga al logearse, pero se consigue que crackearlo tome muchísimo más tiempo, en case de que sea posible.

También es necesario protegerse frente a las Rainbow Tables. Las Rainbow Tables, como a la MD5 que puede verse en este enlace, son tablas de búsqueda inversa para hashes. El creador de las tablas precalcula los hashes MD5 para palabras comunes, frases, palabras modificadas y strings aleatorios. La facilidad de crackear un algoritmo MD5 hace posible la existencia de este tipo de tablas.

Generar este tipo de tablas para un algoritmo complejo tarda mucho más, pero es posible también. Una medida apropiada es añadir un salt al hash. En este contexto, salt es cualquier frase que se añade a la contraseña antes de crear el hash. Usando un salt se gana mucho terreno frente a este tipo de tablas. Se debería crear una Rainbow Table específica para tu aplicación y averiguar cuál es el salt en tu aplicación.

2. Mejora de los antiguos métodos de encriptación

Primero veamos las funciones básicas de hashing para PHP:

- md5

```
string md5 (string $str [, bool $raw_output = false ])
```

Calcula un hash con el algoritmo md5. Si se establece `$_rawoutput` como true se devolverá en raw binario con una longitud de 16. De normal devuelve un hash de 32 caracteres hexadecimal.

- sha1

```
string sha1 (string $str [, bool $raw_output = false ])
```

Calcula un hash con el algoritmo sha1. Si se establece `$_rawoutput` como true se devolverá en raw binario con una longitud de 20. De normal devuelve un hash de 40 caracteres hexadecimal.

- hash

```
string hash ( string $algo, string $data [, bool $raw_output = false ] )
```

La función toma primero el algoritmo que se desea emplear, `$algo`, y después el string que se desea encriptar, `$data`. El algoritmo puede ser md5, sha128, sha256...

Anteriormente, el siguiente código era un ejemplo de una buena protección de contraseñas:

```
class Password {
    const SALT = 'EstoEsUnSalt';
    public static function hash($password) {
        return hash('sha512', self::SALT . $password);
    }
    public static function verify($password, $hash) {
        return ($hash == self::hash($password));
    }
}

// Crear la contraseña:
$hash = Password::hash('micontraseña');

// Comprobar la contraseña introducida
if (Password::verify('micontraseña', $hash)) {
    echo 'Contraseña correcta!\n';
} else {
    echo "Contraseña incorrecta!\n";}
}
```

Durante mucho tiempo esto ha sido la mejor forma de protegerse, mejor que usar md5. Se usa un algoritmo mucho más complejo como el sha512, y fuerza a todas las contraseñas a usar un salt, pero tiene algunas carencias:

- Se utiliza un salt, pero todas las contraseñas utilizan el mismo, por lo que si alguien consigue averiguar una contraseña, o el acceso al código fuente donde puede mirar el hash, se puede hacer una Rainbow Table añadiendo el salt descubierto. La solución es crear un salt aleatorio para cada contraseña que se crea, y guardar el salt con la contraseña de forma que después se pueda recuperar.
- Se utiliza sha512, un complejo algoritmo que viene con PHP. Sin embargo también puede ser crackeado a un ratio de 46 millones de cálculos por segundo. Aunque es más lento de crackear que md5 y sha1, todavía no es un nivel de seguridad estable. La solución es utilizar algoritmos que son todavía más complejos y emplearlos varias veces. Por ejemplo emplear un algoritmo sha512 10 veces consecutivamente reduciría el intento de hackeo considerablemente.

Las dos soluciones ya vienen por defecto con la librería Hash de contraseñas de PHP.

3. La librería Hash de contraseñas de PHP

La extensión Hash de contraseñas crea un password muy complejo, incluyendo la generación de salts aleatorios. En forma más simple se utiliza la función `_passwordhash()`, con la contraseña que quieres "hashear", y la extensión lo hace directamente. Es necesario facilitar también el algoritmo que se desea emplear. La mejor opción de momento es especificar `PASSWORD_DEFAULT` (se actualiza siempre que se añade un algoritmo nuevo más fuerte), aunque también es posible `PASSWORD_BCRYPT`.

- `password_hash`

`string password_hash (string $password , integer $algo [, array $options])`

Es compatible con `crypt()` por lo que los hash de contraseñas creados con `crypt()` se pueden usar con `_passwordhash()`. Las opciones que se admiten son `salt` (para proporcionarlo manualmente, pero esta opción ya está obsoleta en PHP 7 por lo que

no conviene usarla) y *cost*, que denota el coste del algoritmo a usar (el valor predeterminado es 10).

```
$hash = password_hash('micontraseña', PASSWORD_DEFAULT, [15]);
```

El coste indica cuánto de complejo debe ser el algoritmo y por lo tanto cuánto tardará en generarse el hash. El número se puede considerar como el número de veces que el algoritmo hashea la contraseña.

Para poder verificar los passwords, deberíamos saber el salt que se ha creado. Si se usa `_passwordhash()` otra vez y se compara con el anterior, se puede ver que son distintos. Cada vez que se llama a la función, se genera un nuevo hash, por lo que la extensión facilita una segunda función: `_passwordverify()`. Llamando a esta función y pasando la contraseña proporcionada por el usuario, la función devolverá true si coincide con la almacenada:

```
if(password_verify($password, $hash)){  
    // Password correcto!  
}
```

Ahora la clase que habíamos puesto al principio se puede refactorizar por una mucho más segura:

```
class Password {  
    public static function hash($password) {  
        return password_hash($password, PASSWORD_DEFAULT, ['cost' => 15]);  
    }  
    public static function verify($password, $hash) {  
        return password_verify($password, $hash);  
    }  
}
```

4. Cambios en los métodos de encriptación

Usando la extensión de encriptación de PHP, tu aplicación estará con los últimos estándares en seguridad, aunque hace algunos años de decía que SHA-1 era lo mejor.

Esto significa que cada vez se van actualizando los algoritmos y por tanto la extensión se irá adaptando.

¿Y qué ocurre con las contraseñas antiguas? Para eso está la función `_password_needsrehash()`, que detecta si una contraseña almacenada no cumple con las necesidades de seguridad de la aplicación. La razón puede ser que hayas aumentado la complejidad con `cost`, o que PHP haya actualizado el algoritmo. Por esta razón se ha de elegir `PASSWORD_DEFAULT`, siempre se estará protegido con la opción más segura disponible.

Cuando un usuario se logea, ahora tendríamos una nueva tarea, llamar a `_password_needsrehash()`, que toma parámetros similares a `_passwordhash()`. Lo que hace la función `_password_needsrehash()` es decirte si el password necesita un rehash. Depende de ti cuándo generar un nuevo hash de contraseña y guardarlo, porque la extensión Hash desconoce cómo deseas hacerlo.

El siguiente es un ejemplo de una clase Usuario simulada para ver el funcionamiento de la extensión Hash, y sirve de orientación a cómo podría realizarse:

```
class User {
    // Opciones de contraseña:
    const HASH = PASSWORD_DEFAULT;
    const COST = 14;
    // Almacenamiento de datos del usuario:
    public $data;
    // Constructor simulado:
    public function __construct() {
        // Leer los datos de la base de datos almacenados en $data, como
        // $data->passwordHash o $data->username
    }
    // Funcionalidad de guardar los datos simulada:
    public function save() {
        // Guardar los datos de $data en la base de datos }
    }
}
```

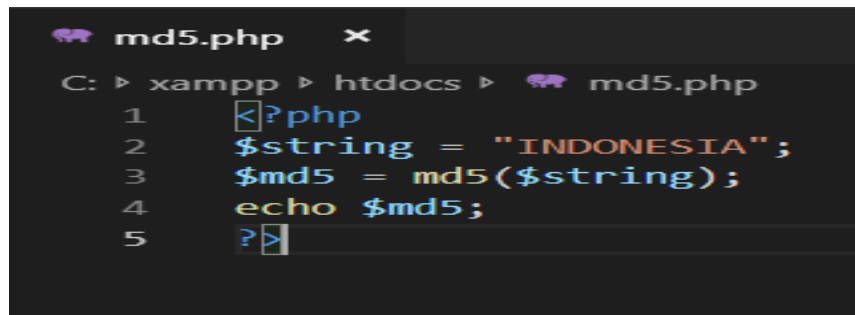
Ya hemos construido la base de la clase user, ahora vamos a ver el cambio de contraseña y el login:

```
// Permite el cambio de contraseña:
public function setPassword($password) {
    $this->data->passwordHash = password_hash($password, self::HASH, ['cost' =>
self::COST]);
}
// Logear un usuario:
public function login($password) {
    // Primero comprobamos si se ha empleado una contraseña correcta:
    echo "Login: ", $this->data->passwordHash, "\n";
    if (password_verify($password, $this->data->passwordHash)) {
        // Exito, ahora se comprueba si la contraseña necesita un rehash:
        if (password_needs_rehash($this->data->passwordHash, self::HASH, ['cost'
=> self::COST])) {
            // Tenemos que hacer rehash en la contraseña y guardarla. Simplemente se
llama a setPassword():
            $this->setPassword($password);
            $this->save();
        }
        return true; // O hacer lo necesario para indicar que el usuario se ha logeado.
    }
    return false;
}
}
```

(Lazaro, 2018)

Figura 2.13.

Encriptación de Contraseñas MD5 en PHP.

A screenshot of a code editor window titled 'md5.php'. The editor shows the following PHP code:

```
1 <?php
2 $string = "INDONESIA";
3 $md5 = md5($string);
4 echo $md5;
5
```

The code is displayed in a dark-themed editor with syntax highlighting. The file path in the background is 'C: > xampp > htdocs > md5.php'.

Nota. Encriptación MD5 en PHP

2.8.Herramientas

2.8.1.Gestor de Base de Datos MariaDB

MariaDB es un sistema de gestión de base de datos MySQL. El propósito de este sistema es proveer capacidades similares y extendidas en relación a MySQL, con un foco especial en mantener el software de forma libre a través de la licencia GNU GPL. Con aplicaciones muy amplias, puede ser usado por equipos de desarrollo de software. (Salas, s.f.)

2.8.2.Lenguaje de Programación

Es un lenguaje de programación de texto plano en UTF-8 de código abierto, adecuado para el desarrollo web y que se puede introducir HTML, esto quiere decir que podemos combinar código PHP con código HTML, PHP nos ayuda a generar páginas web dinámicas, se refiere que el contenido no es el mismo.

PHP es un preprocesador de hipertexto, es un lenguaje de programación de propósito general de código del lado del servidor originalmente diseñado para el preprocesado de texto plano en UTF-8. Posteriormente se aplicó al desarrollo web de contenido dinámico, dando un paso evolutivo en el concepto de aplicación en línea, por su carácter de servicio.

Su implementación en los documentos HTML era aparentemente muy sencilla. Hay que decir, que, PHP no genera HTML, sino que ofrece una salida de texto con

codificación UTF-8 compatible con los documentos HTML. El programador puede dotar a la salida de los tag's propios del HTML y los exploradores más comunes para navegar por internet, reconocerán muy rápidamente el formato UTF-8 y lo adaptarán ofreciendo una salida entendible.

Fue uno de los primeros lenguajes de programación del lado del servidor que se podían incorporar directamente en un documento HTML en lugar de llamar a un archivo externo que procese los datos. El código es interpretado por un servidor web con un módulo de procesador de PHP que genera el texto plano en formato UTF-8, ampliamente reconocido por el estándar HTML, dando como resultado, en los exploradores, una salida al usuario perfectamente entendible.

El PHP acrónimo de (Hypertext Preprocessor) es un lenguaje interpretado de alto nivel embebido en páginas HTML y ejecutado en el servidor. El PHP se inició como una modificación a Perl escrita por Rasmus Lerdorf a finales de 1994. Su primer uso fue el de mantener un control sobre quien visitaba su curriculum en su web. En los siguientes tres años, se fue convirtiendo en lo que se conoce como PHP/FI 2.0. Esta forma de programar llego a muchos usuarios, pero el lenguaje no tomo el peso actual hasta que Zeev Surasky y Andi Gutmans le incluyeron nuevas características en 1997, que dio por resultado el PHP 3.0. La versión 4 es la más reciente.

Con PHP se puede hacer cualquier cosa que se pueda realizar con un script CGI, como el procesamiento de información en formularios, foros de discusión, manipulación de cookies y páginas dinámicas. Un sitio con páginas dinámicas es el que permite interactuar con el visitante, de modo que cada usuario que visita la página vea la información modificada para requisitos particulares. Las aplicaciones dinámicas para el Web son frecuentes en los sitios comerciales e-commerce, donde el contenido visualizado se genera de la información alcanzada en una base de datos u otra fuente externa.

Una de sus características más potentes es su soporte para gran cantidad de bases de datos. Entre su soporte pueden mencionarse MySQL, Oracle, ProgreSQL, entre otras. PHP también ofrece la integración con varias bibliotecas externas, que permiten

que el desarrollador haga casi cualquier cosa desde generar documentos en pdf hasta analizar código XML. (Pelissier, 2013)

2.8.3.Servidor Web Apache

Apache Web Server, es un servidor de páginas Web desarrollado por la Apache Software Foundation, organización formada por miles de voluntarios que colaboran para la creación de software de libre distribución. Es uno de los servidores más utilizados en Internet ya que se trata de un servidor muy potente, flexible, rápido, eficiente y que siempre está adaptado a nuevos protocolos http. Apache se encuentra disponible para varias plataformas, desde Debían, hasta Windows XP y se le puede incrustar nuevos módulos que le permitirán ejecutar código Script como son JSP, PHP, etc. (Mestras, 2012)

2.8.4.Bootstrap

Bootstrap es un framework CSS de código abierto que favorece el desarrollo web de un modo más sencillo y rápido. Incluye plantillas de diseño basadas en HTML y CSS con la que es posible modificar tipografías, formularios, botones, tablas, navegaciones, menús desplegables, etc. También existe la posibilidad de utilizar extensiones de JavaScript adicionales.

Fue desarrollado inicialmente por Twitter en 2011 y permite crear interfaces de usuario limpias y compatibles con todo tipo de dispositivos. Entre las ventajas que tiene Bootstrap es que favorece el design responsive, el cual se utiliza para mejorar la experiencia de los usuarios en el sitio web y en consecuencia el posicionamiento.

Ventajas de Bootstrap

Entre las ventajas de utilizar bootstrap podemos enumerar las siguientes:

- Es de código abierto, y todo su código actualizado se encuentra en un repositorio de Github.
- Está mantenido y actualizado por Twitter.
- Es compatible con la mayoría de navegadores (Chrome, Safari, Mozilla...).
- Utiliza componentes vitales para los desarrolladores (HTML5, CSS3, jQuery o

GitHub).

- Sus plantillas son de sencilla adaptación responsive.
- Dispone de un conjunto de elementos web personalizables.
- Se integra con librerías JavaScript.
- Usa Less: un lenguaje de las hojas de estilo CSS preparado para enriquecer los estilos de la web. (Armetrics, Armetrics Bootstrap, 2022)

2.8.5.JQuery

JQuery es una biblioteca que facilita y agiliza la creación de páginas y aplicaciones web con JavaScript. jQuery hace uso de muchas tareas comunes que requieren líneas de código JavaScript y las envuelve en métodos que pueden resumirse con una sola línea de código. Dicho de un modo más sencillo, JQuery escribe en una línea de código lo que JavaScript habría escrito en 10 o 20 líneas.

Se puede descargar de forma gratuita. Además de la biblioteca JQuery, existen multitud de complementos jQuery que están disponibles para agregar aún más poder y funcionalidad a sus scripts.

Qué se puede hacer con jQuery

Con jQuery se pueden agregar efectos animados a los elementos (desvanecimiento, salida y expansión, etc.). También te permite hacer solicitudes XML, manipular el DOM, crear presentaciones de diapositivas y menús desplegados, entre otras muchas cosas. Otro de los puntos fuertes de jQuery es que facilita la escritura JavaScript que funciona en muchos navegadores diferentes. Es decir, con JavaScript a veces es necesario escribir diferentes líneas de código para cada navegador, pero con jQuery lo único que hay que hacer es llamar a la función adecuada para que el código se ejecute en diferentes navegadores. (Armetrics, Armetrics JQuery, 2022)

2.8.6.Json

JSON es un lenguaje usado para el intercambio de datos entre sistemas, está basado en la notación de los literales de objeto de Javascript. JSON son las siglas de JavaScript Object Notation, o sea, Notación de Objeto Javascript. Básicamente usa la misma notación o

forma con la que se escriben los objetos Javascript en el código, los literales de objeto, con algunas restricciones y añadidos extra.

La utilidad de JSON es la de intercambiar datos, por eso se conoce como lenguaje de intercambio de información o lenguaje de transporte. Sirve para la comunicación entre servicios web (web services) y los clientes que los consumen, enviando y recibiendo la información en formato JSON.

Características de Json

- JSON es un lenguaje de modelador de datos.
- Consiste en pares "clave - valor".
- Los valores pueden ser cadenas, números o booleanos, así como otros objetos JSON, con cualquier nivel de anidación.
- Es un formato flexible, ligero y fácilmente transferible a través de las redes.

Ventajas de Json

- La lectura del código resulta de fácil lectura y la información es suficientemente expresiva para poder ser leída por personas, además de máquinas.
- El tamaño de los archivos que se transfieren es ligero.
- El código está basado en el lenguaje Javascript, lo que es ideal para las aplicaciones web.
- Todos los lenguajes disponen de funciones para interpretar cadenas JSON y convertir datos en cadenas JSON válidas.
- Se escribe en archivos de texto plano con codificación UTF8, que es compatible con todos los sistemas. (JSON desarrolloweb.com, s.f.)

2.8.7.Html5

El HTML5 es un lenguaje de etiquetas, utilizado para la estructuración y la presentación de contenido en los sitios web. Se finalizó y publicó el 28 de octubre 2014 en el World Wide Web Consortium (W3C), siendo esta la quinta revisión del estándar HTML desde la creación de la World Wide Web. La versión anterior, HTML 4, se estandarizó en 1997.

Sus objetivos principales son mejorar el lenguaje dando soporte para los últimos

objetos multimedia mientras se mantiene fácilmente legible por los humanos y a su vez ser comprendido constantemente por ordenadores y dispositivos (navegadores web, programas de análisis, etc.).

A raíz de sus predecesores inmediatos HTML 4.01 y XHTML 1.1, HTML 5 es una respuesta al hecho de que el código HTML y XHTML, de uso común en la web, tienen una mezcla de características introducidas por diversas especificaciones, junto con las introducidas por los productos de software. También es un intento de definir un único lenguaje de etiquetas que puede ser escrito en HTML o XHTML. Según (Armetrics, 2022)

Permite una mayor interacción entre las páginas web y el contenido multimedia, lo que quiere decir con vídeos, audios, fotos, entre otros. Así como también facilita el proceso de codificar el diseño básico de la web.

Es un sistema que deja formatear el layout de una página web, y permite hacer ajustes al aspecto. Los navegadores de internet como Chrome o Firefox pueden saber cómo deben mostrar la web con todos sus elementos. Por eso el HTML5 no tiene grandes cambios al lado de su predecesor, pero se eleva en sofisticación.

Se diseñó para utilizarse en todos los desarrolladores de Open Web, es una página que tiene diversos recursos sobre la tecnología de HTML5 y, se clasifica en distintos grupos de acuerdo a su función:

Semántica: Que permite describir con mejor precisión el contenido de la web.

Conectividad: Deja al usuario comunicarse con el servidor de una manera más innovadora.

Sin conexión y almacenamiento: Les da la oportunidad a las páginas web de almacenar datos locales y lograr operar sin conexión de una forma más eficiente.

Multimedia: Permite un soporte para utilizar el audio y video nativo.

Gráficos y efectos 2D y 3D: Ofrece una amplia gama de nuevas características que manejan los gráficos de la página como Canvas 2D, WebGL, entre otros.

Rendimiento e Integración: Permite una mejor optimización de la velocidad y usar el hardware de mejor manera.

Acceso al dispositivo: Permite el uso de APIs con diferentes componentes internos de entrada y salida en los dispositivos. (Aritmetrics, 2022)

2.8.8. C^{ss}3

CSS significa hojas de estilo en cascada con énfasis en “Estilo”. Mientras que HTML se usa para estructurar un documento web (definiendo cosas como titulares y párrafos, y permitiéndole incrustar imágenes, videos y otros medios), CSS3 llega y especifica el estilo de su documento; los diseños de página, los colores y las fuentes están todos determinados con CSS3.

Piense en HTML como la base (cada casa tiene una) y CSS como las opciones estéticas (hay una gran diferencia entre una mansión victoriana y una casa moderna de mediados de siglo). (Albertof, 2021)

Siguiendo este fundamento resulta muy fácil para un diseñador web realizar cambios en la apariencia de una web sin afectar de manera dramática a su contenido. El contenido siempre será el mismo, solo cambia como aquello que podemos ver. CSS es fácil de entender y aprender, y nos da un potente control de cómo diseñar los documentos HTML.

Por ejemplo, gracias al CSS podemos definir que todas las etiquetas <H1> de un texto cambie su apariencia por aquella que definamos, cambiando colores, fuentes y añadiendo efectos sin modificar nada del contenido que sirve de base. Imagina ahora si tuvieras que hacerlo revisando línea a línea.

Por tanto la principal ventaja del CSS es que al poder ordenar nuestras preferencias de estilo en un único documento independiente ahorramos una considerable cantidad de tiempo al escribir nuestras órdenes CSS en forma de plantilla una única vez para ser aplicado luego a todas las páginas que componen nuestra página web.

Pero no es la única ventaja. Gracias a esto también el rendimiento de nuestra web se ve incrementado cargando de una manera más rápida pues no es necesario escribir etiquetas HTML continuamente y al tener que insertar una menor cantidad de código también su mantenimiento es mucho menor. (Robledano, 2019)

2.9.Pruebas de Software

2.9.1.Pruebas de Caja Blanca

Son pruebas estructurales. Conociendo el código y siguiendo su estructura lógica, se pueden diseñar pruebas destinadas a comprobar que el código hace correctamente lo que el diseño de bajo nivel indica y otras que demuestren que se comporta adecuadamente ante determinadas situaciones.

2.9.2.Pruebas de Caja Negra

En la teoría de sistemas y física, se denomina caja negra aquel elemento que es estudiado desde el punto de vista de las entradas y salidas o respuestas que produce, sin tener en cuenta su funcionamiento interno. De una caja negra interesará su forma de interactuar con el medio que lo rodea (en ocasiones, otros elementos que también podrían ser cajas negras) entendiendo qué es lo que hace, pero sin dar importancia a cómo lo hace. Por tanto, de una caja negra deben estar muy bien definidas sus entradas y salidas, es decir, su interfaz: en cambio, no se precisa definir ni conocer los detalles internos de su funcionamiento.

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

- Funciones incorrectas o ausentes.
- Errores de interfaz.
- Errores de estructuras de datos o en acceso a base de datos externas.
- Errores de rendimiento.
- Errores de inicio y fin.

2.9.3.Pruebas de Funcionalidad

Una vez finalizado el desarrollo de las primeras cuatro etapas de abrir se realizan las pruebas para garantizar el funcionamiento del sistema, tomando en cuenta los casos de uso representativos del mismo. El uso de las pruebas funcionales es para asegurar correcto trabajo de entrada de datos, la navegación en el sistema, procedimientos y obtención de resultados.

CAPÍTULO III

MARCO APLICATIVO

3.1.Introducción

En este capítulo se desarrollara las etapas y modelos correspondientes a la fase de obtención de requerimientos, diseño del sistema, análisis y la fase de implementación, siguiendo el proceso de desarrollo de la metodología UWE para este proyecto, detallados en el capítulo II.

3.2.Ingeniería Web Basada UML (UWE)

3.2.1.Obtención de Requerimientos

Para realizar un mejor proyecto se requiere la tarea de ingeniería de requerimientos que es muy importante y fundamental para que un sistema tenga éxito, así que se realizó las siguientes actividades correspondientes de la tabla:

Tabla 3.1

Obtención de Requerimientos.

Tareas Realizadas	Características
Entrevista	Se realizó las correspondientes entrevistas al directorio del Sindicato y a los Jefes de grupo de la Institución Señor de San Roque.
Observaciones	Se observó que en la Institución tienen algunos problemas en el área administrativa, porque la mayoría de los procesos se hacen de forma manual, lo que se requiere es mejorar y hacer más eficiente.
Documentación	Se me permitió observar y revisar el proceso que sigue la documentación de los Socios Propietarios.

3.2.1.1. Definición de Actores

Los Usuarios se presentan por diferentes actores de acuerdo al sistema, a continuación se muestra en la siguiente tabla los actores que interactúan con el sistema.

Tabla 3.2

Definición de Actores.

Actor	Descripción
Administrador de Sistema General	<ul style="list-style-type: none"> • Tiene acceso a todo el sistema • Es el que toma las decisiones principales, coordina las actividades con el Personal administrativo.
Personal Administrativo	<ul style="list-style-type: none"> ✓ Registra nuevo Conductor ✓ Registra Vehículo: clases, Marca, colores ✓ Registra Asignación de grupos ✓ Registra Asignación de línea ✓ Registra Asignación de ruta ✓ Generar Refuerzo de herramienta ✓ Generar Memorándum de trabajo ✓ Realiza búsquedas de jefes de grupo y conductores ✓ Genera Reportes
Agente de Parada	<ul style="list-style-type: none"> ▪ Control de Conductores de su grupo ▪ Visualiza Información de Agente de parada
Jefes de Grupo	<ul style="list-style-type: none"> ➤ Realiza búsqueda de conductores de su grupo ➤ Generar Refuerzo de herramienta ➤ Generar Memorándum de trabajo ➤ Genera Reportes de conductores ➤ Visualiza Información del jefe de grupo
Choferes	<ul style="list-style-type: none"> ❖ Verifica en que grupo se encuentra el conductor ❖ Visualiza Información del conductor

3.2.2.Requerimientos del Sistema

La obtención correcta de los requerimientos necesarios que lleguen a describir con claridad en forma consistente y compacta el comportamiento de un sistema de los requerimientos funcionales y requerimientos no funcionales.

3.2.2.1.Requerimientos Funcionales

En la siguiente tabla se describen los requerimientos funcionales mínimas que realiza el presente proyecto:

Tabla 3.3

Requerimientos Funcionales.

Ref.	Función	Categoría
R.1.1	El sistema debe tener seguridad en el acceso a la información	Evidente
R.1.2	Acceder al sistema por tipos de usuarios (General del Sindicato, Administrativo, Agente de parada, Jefe de grupo, Conductores)	Evidente
R.1.3	Registro de Movilidades	Evidente
R.1.4	Registro de Grupos	Evidente
R.1.5	Registro de Líneas	Evidente
R.1.6	Registro de Rutas	Evidente
R.1.7	Realiza Búsquedas	Evidente
R.1.8	Generar Memorándum de trabajo	Evidente
R.1.9	Generar Reportes	Evidente
R.1.10	Cerrar Sesión	Oculto

3.2.2.2.Requerimientos no Funcionales

En la siguiente tabla se describen los requerimientos no funcionales del presente proyecto las cuales restringen o condicionan el desarrollo e implementación del sistema:

Tabla 3.4

Requerimientos no Funcionales.

Ref.	Función	Categoría
R.2.1	El sistema debe visualizarse y funcionar correctamente en cualquier navegador: Mozilla, Opera Mini, Chrome.	Evidente
R.2.2	El ingreso al sistema será restringido, solo para usuarios autorizados de la Institución, por lo tanto requiere de identificación del usuario con una contraseña. De acuerdo al rol permitido.	Evidente
R.2.3	Soporte y mantenimiento periódico para asegurar el buen rendimiento del sistema.	Evidente
R.2.4	El sistema estará disponible las 24 horas para su acceso.	Evidente
R.2.5	El sistema debe tener seguridad en el acceso a la información del sistema bajo los roles del usuario.	Evidente

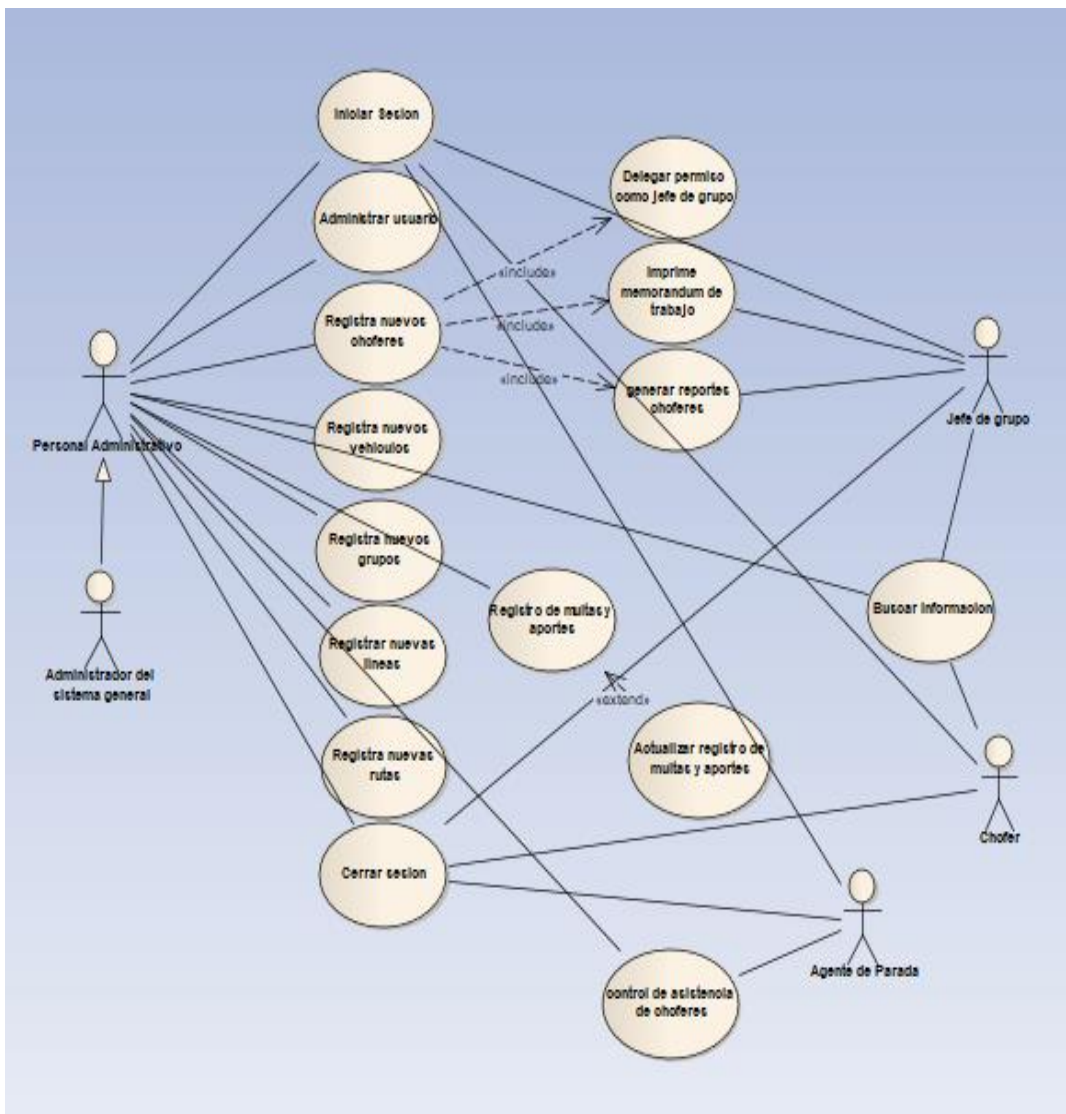
3.3.Fase de Diseño del Sistema

3.3.1.Diagrama de Caso de Uso General

Se hace el modelado donde se puede apreciar cómo interactúan los actores sobre los casos de uso del sistema:

Figura 3.1.

Diagrama de Caso de Uso General.



El diagrama de caso de uso general, muestra de manera resumida todas las interacciones de los actores con las funciones del sistema.

3.3.2. Diagrama de Caso de Uso Administración del Sistema

Se hace el modelado donde se puede apreciar cómo interactúa el administrador del sistema sobre los casos de uso:

Figura 3.2.

Diagrama de Caso de Uso: Administrador del Sistema.

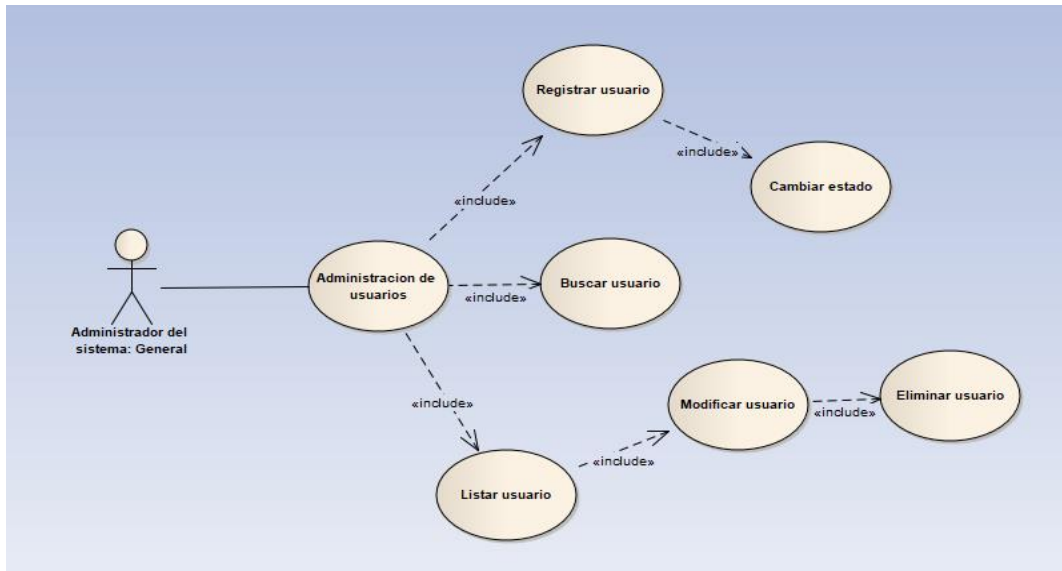


Tabla 3.5

Descripción de Caso de Uso: Administrador del Sistema.

Caso de Uso: Administración del Sistema	
Actores	Administrador del Sistema
Tipo	Primario Esencial
Descripción	El administrador registra, controla y designa el rol de cada usuario, Administrativo, Agente, Jefe de grupo, Conductor en base a las funciones que desempeña la Institución Señor de San Roque. Restringe el acceso al sistema habilitando y deshabilitando a usuarios.

3.3.3. Diagrama de Caso de Uso Personal Administrativo del Sistema

Se hace el modelado donde se puede apreciar cómo interactúa el personal administrativo del sistema sobre los casos de uso:

Figura 3.3.

Diagrama de Caso de Uso: Personal Administrativo.

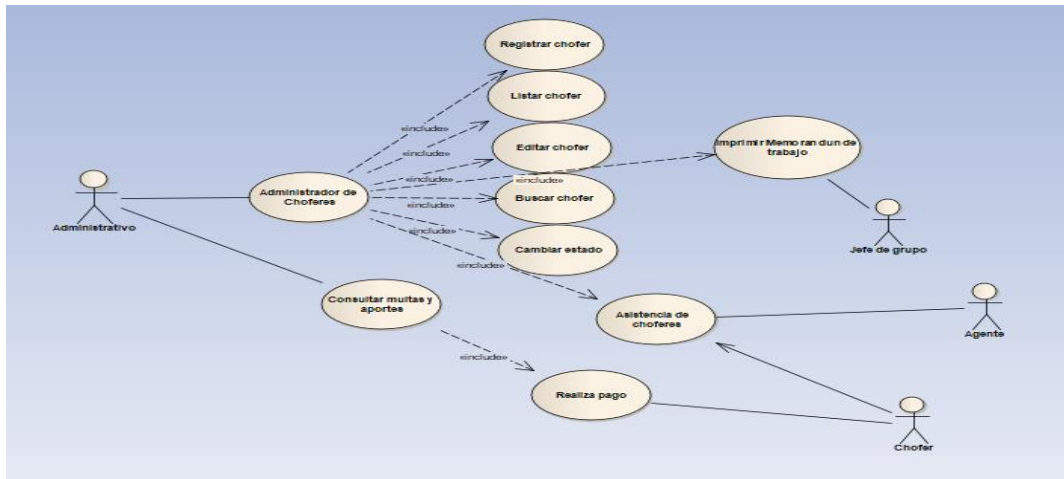


Tabla 3.6

Descripción de Caso de Uso: Personal Administrativo.

Caso de Uso: Personal Administrativo	
Actores	Personal Administrativo
Tipo	Primario Esencial
Descripción	El administrativo registra, controla y designa el rol de cada Conductor y asigna como Jefe de grupo en base a las funciones que desempeña la Institución Señor de San Roque y también registra usuario Agente de parada para que haga el control de ingreso de Conductor. Restringe el acceso al sistema habilitando y deshabilitando de los Conductores.

3.3.4. Diagrama de Caso de Uso Jefe de Grupo

Se hace el modelado donde se puede apreciar cómo interactúa el Jefe de grupo del sistema sobre los casos de uso:

Figura 3.4.

Diagrama de Caso de Uso: Jefe de Grupo.

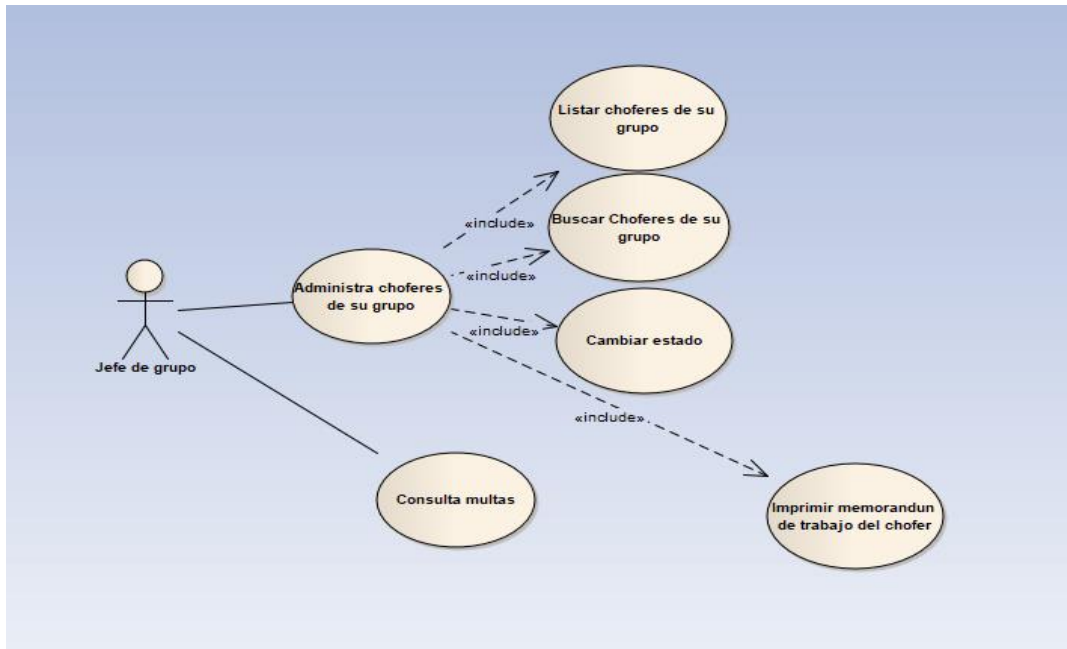


Tabla 3.7.

Descripción de Caso de uso: Jefe de Grupo.

Caso de Uso: Jefe de grupo	
Actores	Jefe de Grupo
Tipo	Secundario
Descripción	El Jefe de grupo controla a su grupo de base, busca al socio propietario si se encuentra en el grupo, tiene el acceso de imprimir memorándum de trabajo de cualquier socio propietario.

3.3.5. Diagrama Casos de Uso Asignación de Vehículos

Se hace el modelado donde se puede apreciar cómo puede interactuar el administrador del sistema para la asignación de vehículos sobre los casos de uso:

Figura 3.5.

Diagrama de Casos de Uso: Asignación de Vehículos.

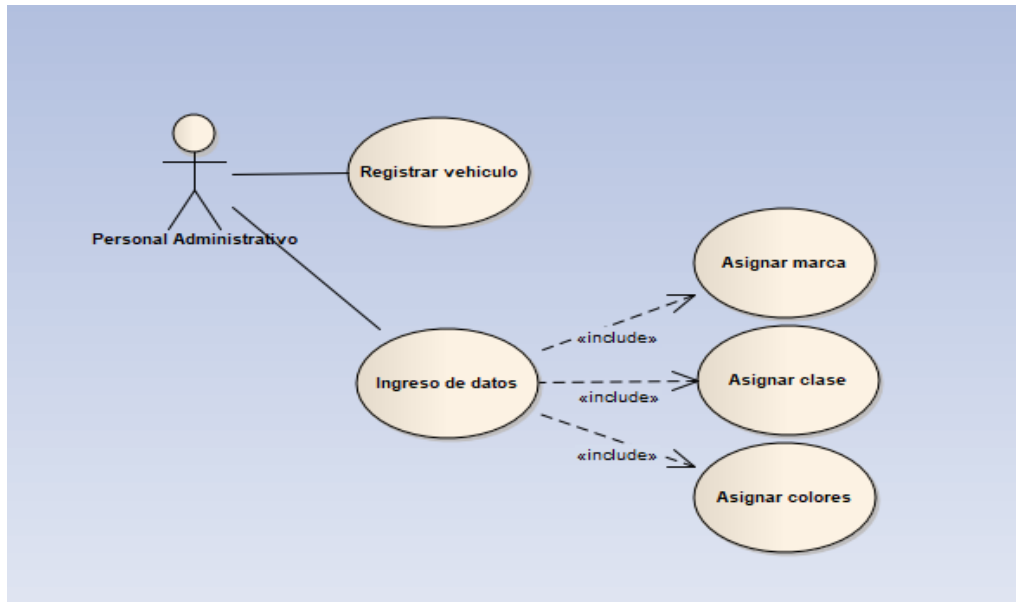


Tabla 3.8

Descripción de Casos de Uso: Asignación de Vehículos.

Caso de Uso: Asignación de Vehículos	
Actores	Personal Administrativo
Tipo	Primario esencial
Descripción	El Personal administrativo registra vehículos, ingresa el dato de: asigna marca, asigna clase, asigna colores.

3.3.6. Diagrama de Caso de Uso Reportes

Se hace el modelado donde se puede apreciar cómo interactúa el administrador del sistema para imprimir los reportes sobre los casos de uso:

Figura 3.6.

Diagrama de Casos de Uso: Reportes.

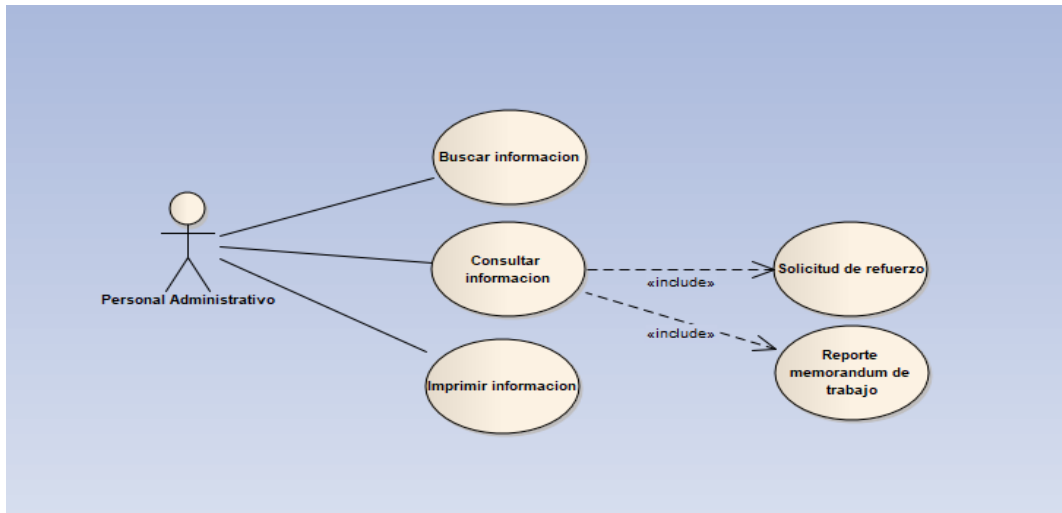


Tabla 3.9

Descripción de Casos de Uso: Reportes.

Caso de Uso: Reportes	
Actores	Personal Administrativo
Tipo	Primario esencial
Descripción	<p>El Personal administrativo podrá visualizar toda la información del Conductor (Socio: propietario o asalariado).</p> <p>El sistema no permite modificaciones al visualizar la información del Conductor.</p>

3.4. Diagrama de Clases

Figura 3.7.

Diagrama de Clases.

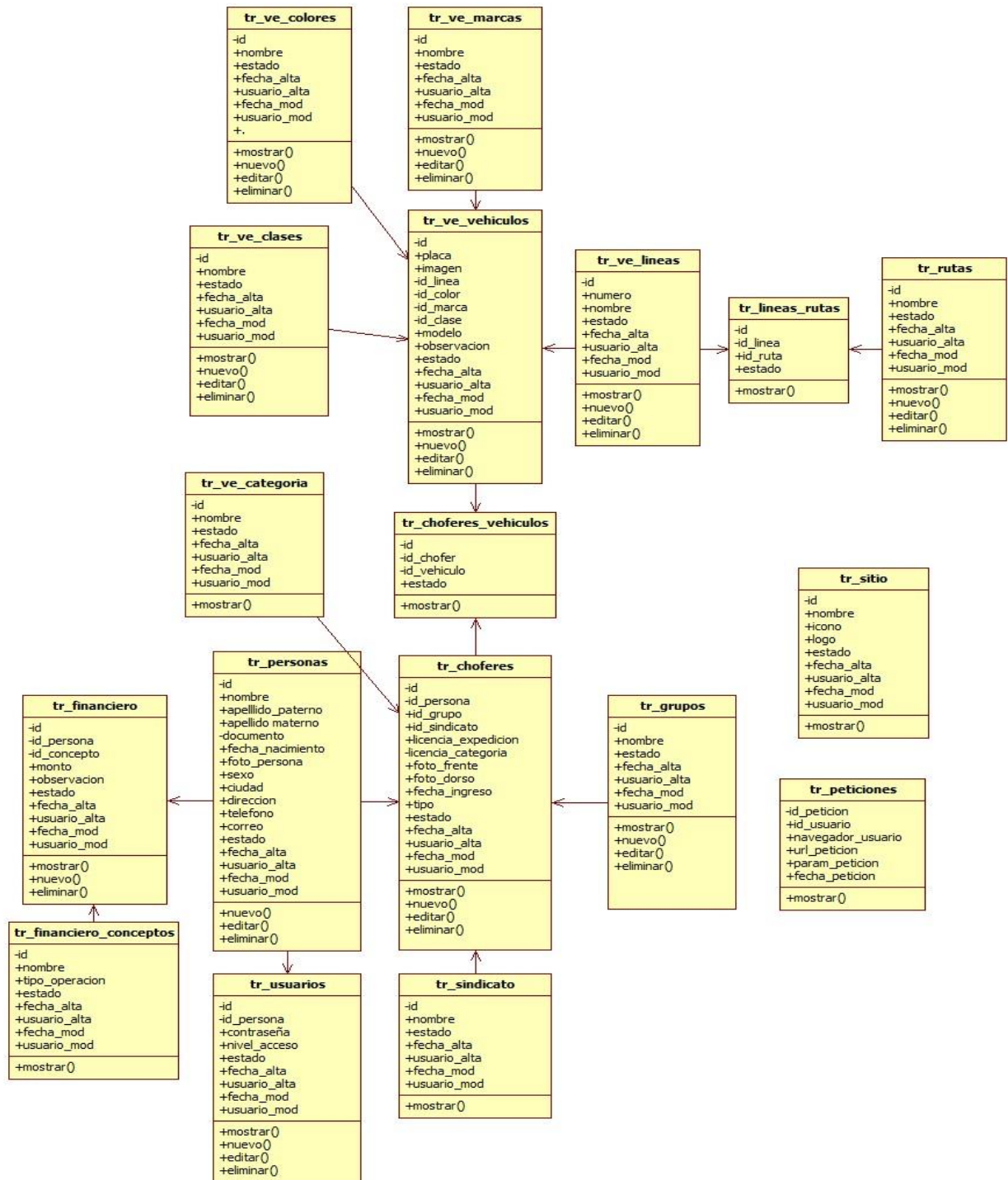
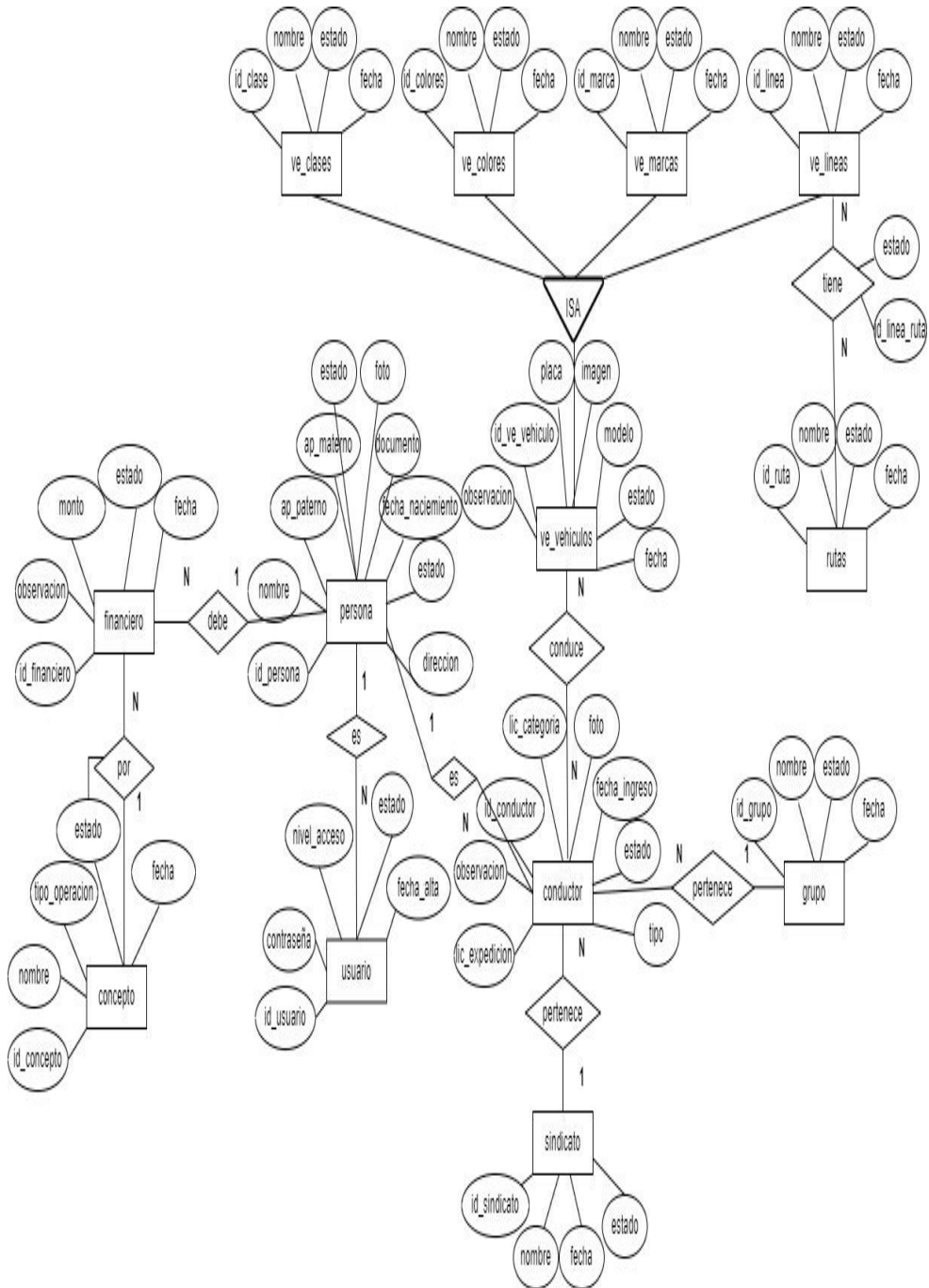


Diagrama Entidad Relación

Figura 3.8.

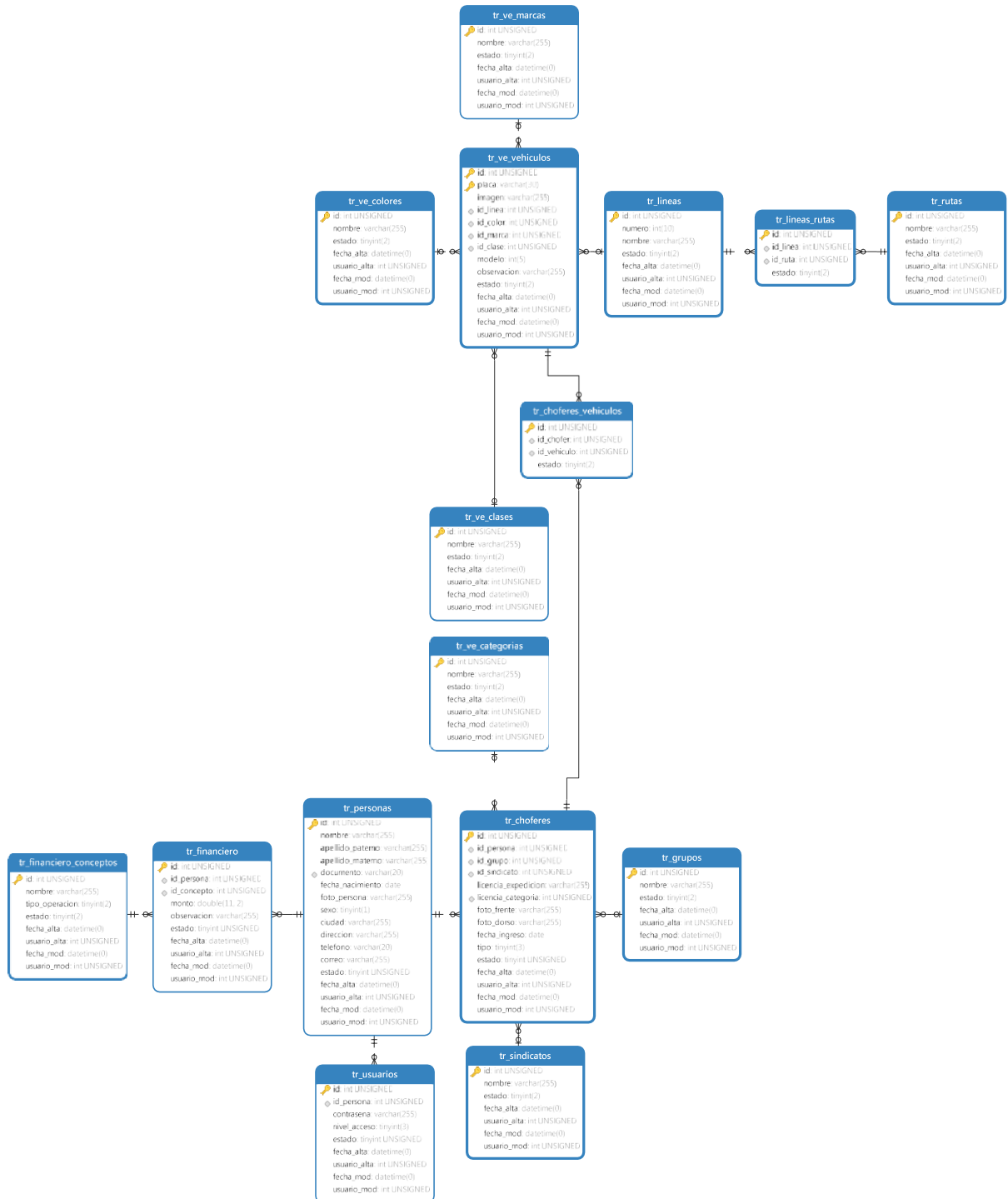
Diagrama de Flujo de Datos



3.5. Diagrama de Contenido

Figura 3.9.

Modelo de Contenido: Dato Relacional.

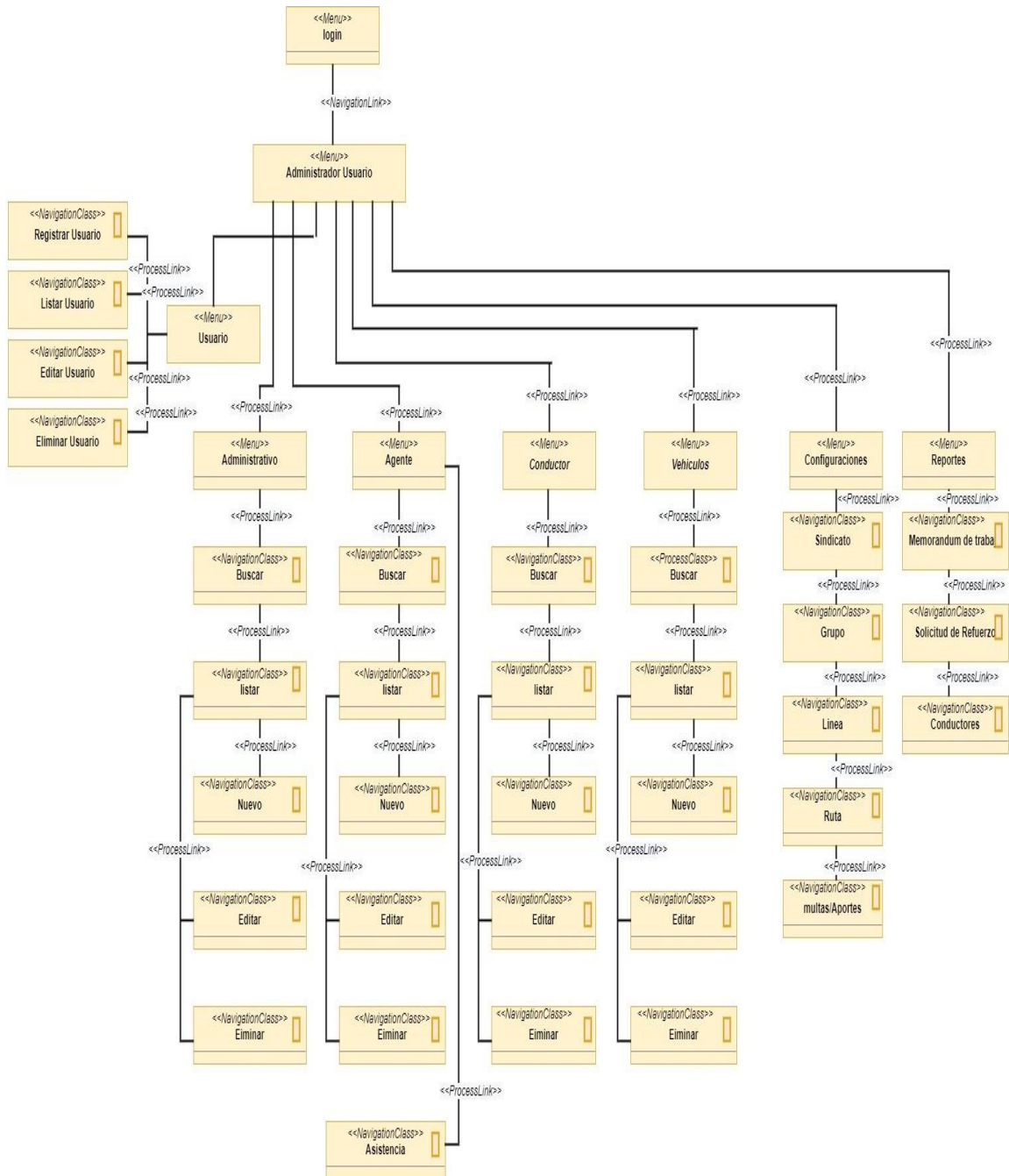


3.6. Diseño de Navegación

A continuación, se hace el modelo donde se aprecia la interacción de los usuarios en la navegación del sistema:

Figura 3.10.

Diseño de Navegación.



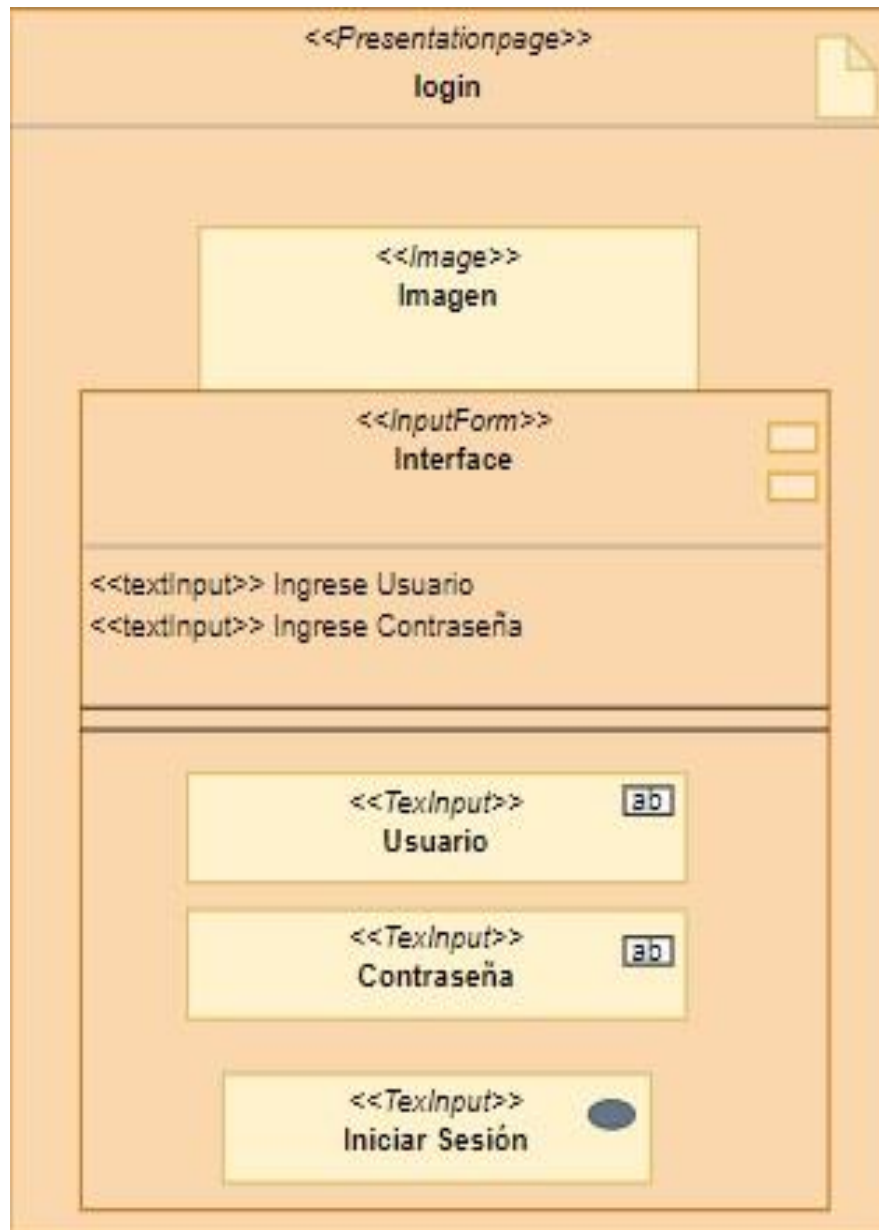
3.7. Diseño de Presentación

3.7.1. Modelo de Presentación Login

A continuación, se hace el modelo donde se aprecia la interacción de los diferentes usuarios para ingreso al sistema:

Figura 3.11.

Modelo de Presentación Login.



3.7.2. Modelo de Presentación Administrador

A continuación, se hace el modelo donde se aprecia la interacción dependiendo al rol de cada usuario para la navegación del sistema:

Figura 3.12.

Modelo de Presentación Administrador.



3.7.3. Modelo de Presentación Jefe de Grupo

A continuación, se hace el modelo donde se aprecia la interacción dependiendo al rol de cada usuario para la navegación del sistema:

Figura 3.13.

Modelo de Presentación Jefe de Grupo.

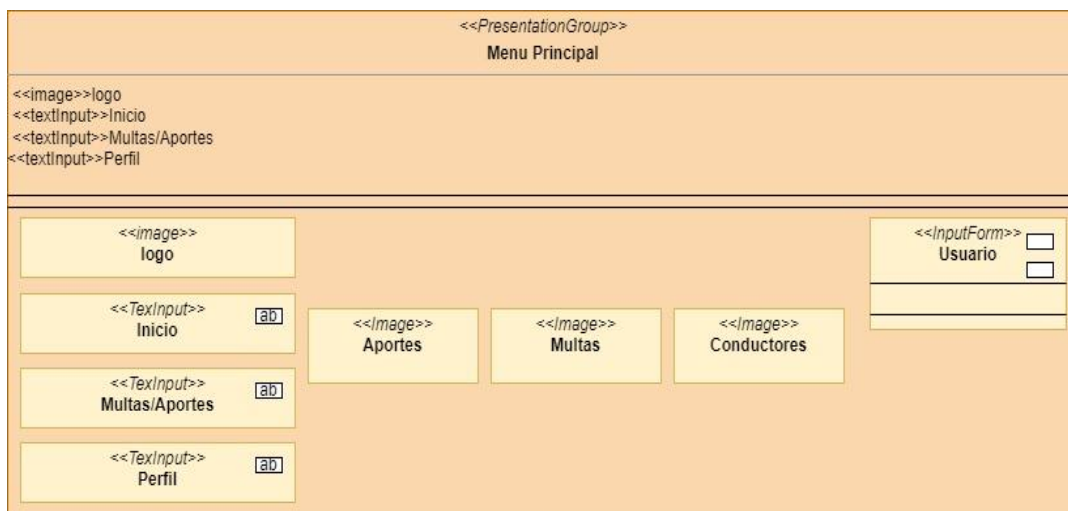


3.7.4. Modelo de Presentación Conductor

A continuación, se hace el modelo donde se aprecia la interacción dependiendo al rol de cada usuario para la navegación del sistema:

Figura 3.14.

Modelo de Presentación Conductor.



3.7.5. Modelo de Presentación Agente

A continuación, se hace el modelo donde se aprecia la interacción dependiendo al rol de cada usuario para la navegación del sistema:

Figura 3.15.

Modelo de Presentación Agente.



3.8. Fase de Codificación del Software

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

Figura 3.16.

Código Fuente.

```
<!-- PAGE-HEADER -->
<div class="page-header">
  <div class="page-title">Inicio/01
  <div class="breadcrumb">
    <li class="breadcrumb-item"><a href="javascript:void(0)">Inicio/01
    <li class="breadcrumb-item active">Inicio/11
  </div>
</div>
<!-- PAGE-HEADER END -->

<!-- ROW-1 -->
<div class="row">
  <div class="col-lg-12 col-md-12 col-sm-12 col-xl-12">
    <div class="row">
      <div class="col-lg-6 col-md-6 col-sm-12 col-xl-4">
        <div class="card overflow-hidden">
          <div class="card-body">
            <div class="d-flex">
              <div class="flex-grow-1">
                <div class="text-muted fs-12"><span class="text-green">+1
                <div class="text-green">+1
            </div>
          </div>
        </div>
      </div>
      <div class="col-lg-6 col-md-6 col-sm-12 col-xl-4">
        <div class="card overflow-hidden">
          <div class="card-body">
            <div class="d-flex">
              <div class="flex-grow-1">
                <div class="text-muted fs-12"><span class="text-green">+1
                <div class="text-green">+1
            </div>
          </div>
        </div>
      </div>
    </div>
  </div>
</div>
```

3.9.Fase de Pruebas

Las pruebas se utilizan para asegurar el correcto funcionamiento del sistema con sus respectivos módulos de la siguiente manera.

Figura 3.17.

Módulo de Usuarios: Administrador, Administrativo, Agente.

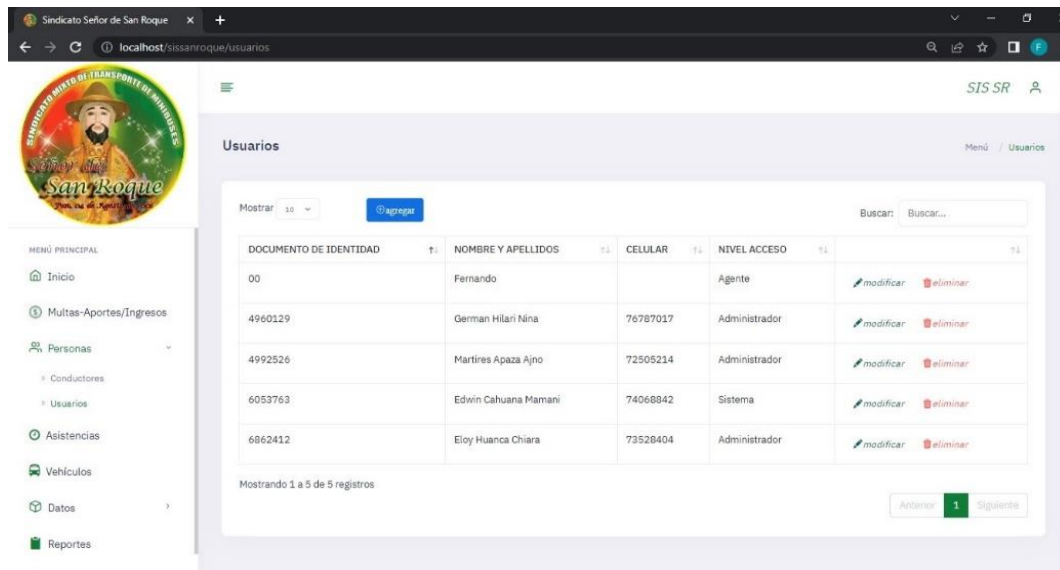


Figura 3.18.

Modulo Conductores.

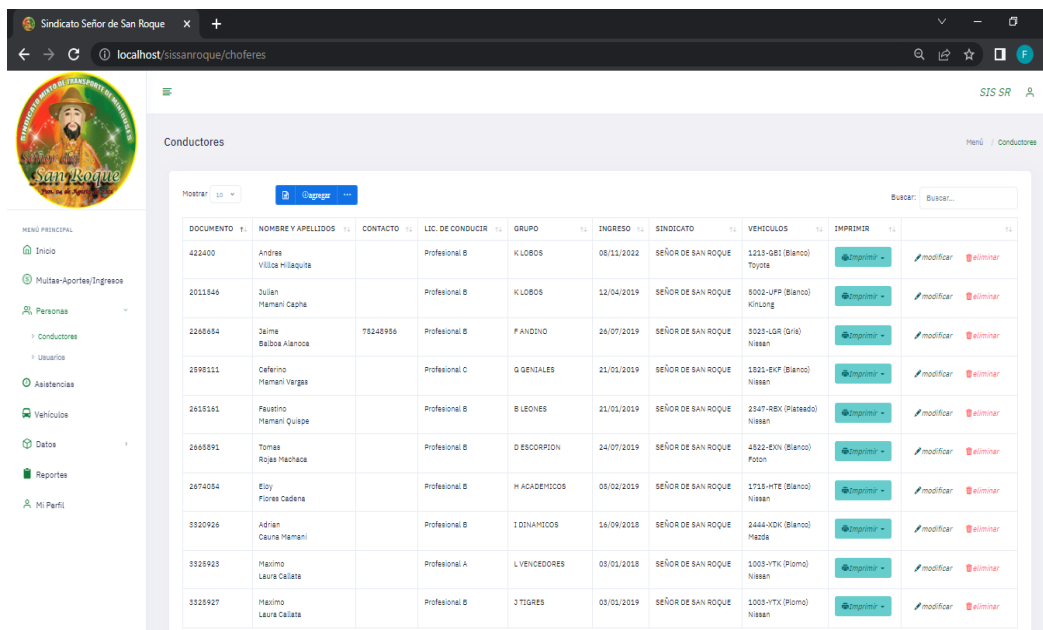


Figura 3.19.

Diseño Conductor: Datos como Persona.

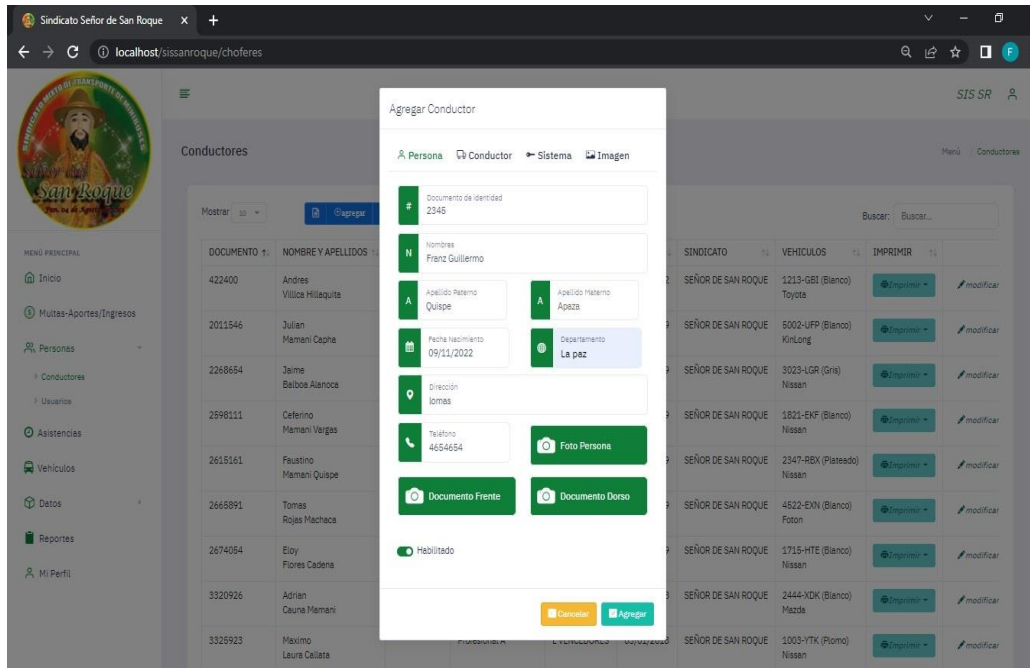


Figura 3.20.

Diseño Conductor: Datos como Conductor.

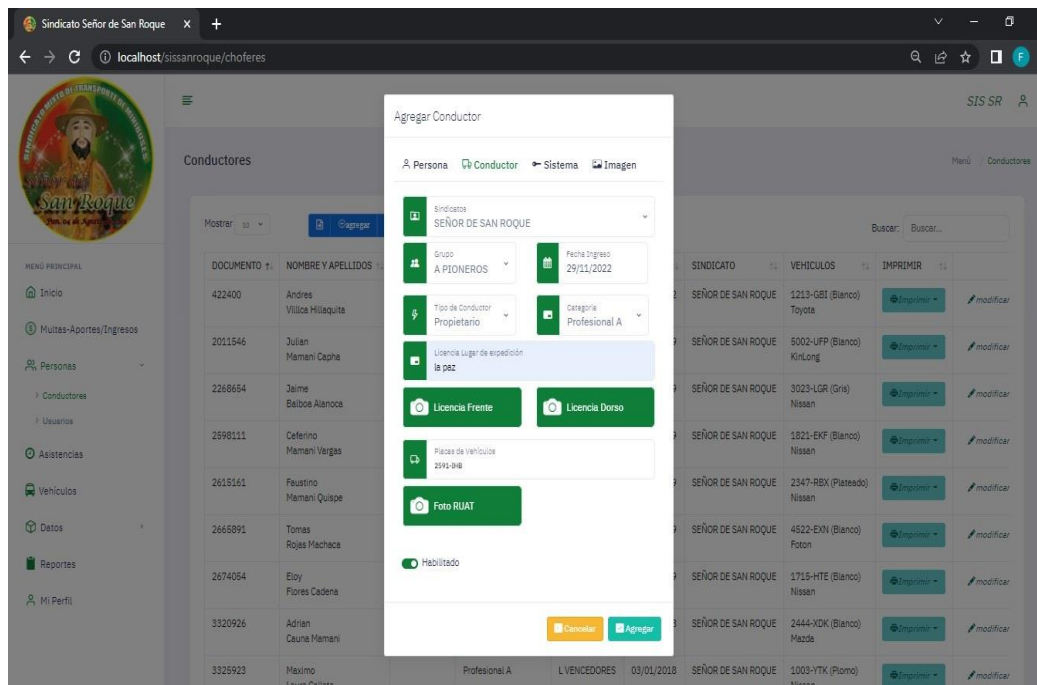


Figura 3.21.

Diseño Conductor: Datos como Usuario.

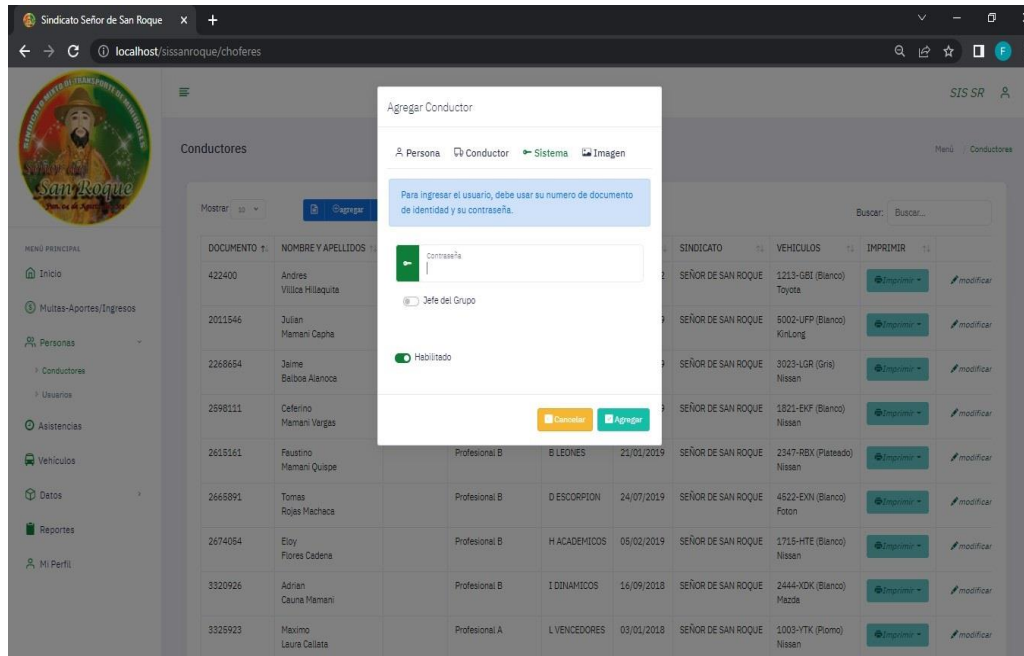


Figura 3.22.

Diseño Conductor: Archivos Subidos.

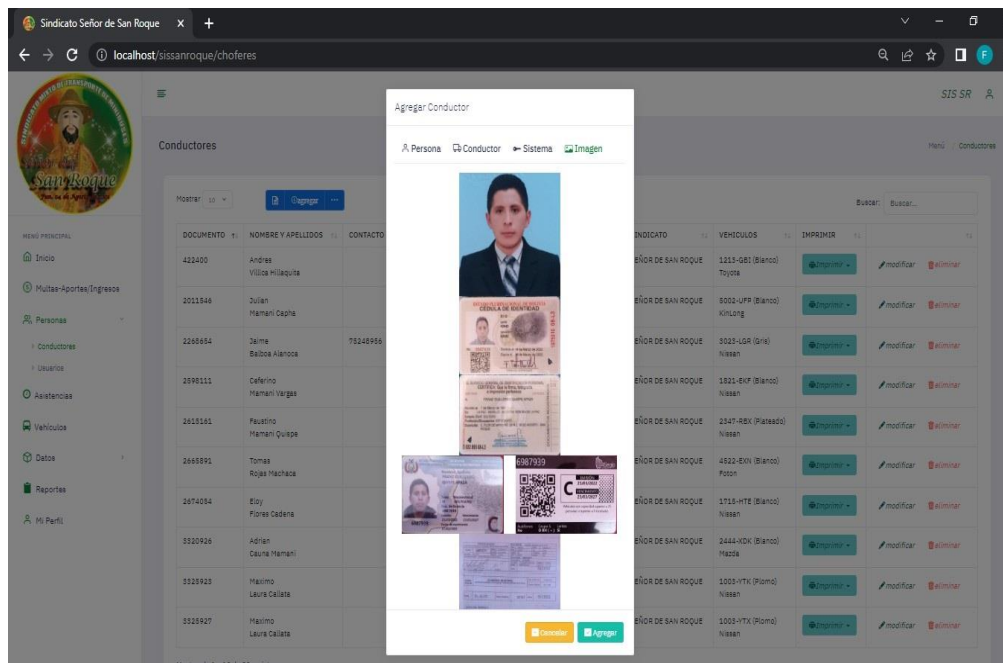


Figura 3.23.

Reporte Conductor: Memorándum de Trabajo.

Figura 3.24.

Reporte Conductor.

Identidad	Foto	Nombres	Apellidos	Contacto	Grupo	Ingreso	Sindicato	Vehiculo
789		JOSE	MENDEZ QUISSPE	465465 LOMAS	F Andino	07/08/2022	SEÑOR DE SAN ROQUE	PLACA : 2591IHB COLOR : Blanco MARCA : Nissan MODELO: 1994 CLASE : MINIBUS

Figura 3.25.

Modulo Aportes.

Multas-Aportes/Ingresos

Mostrar: 10 [+agregar](#) Buscar:

DOCUMENTO DE IDENTIDAD	NOMBRE Y APELLIDOS	MONTO	CONCEPTO	OBSERVACION	FECHA
6987939	Franz Guillermo Quispe Apaza	+ 30.00 + 300.00 + 30.00 Total: 360.00	Multa Ingresos Aporte	basica Ingreso basica aniversario	08 Noviembre 2022 08 Noviembre 2022 08 Noviembre 2022
9252117	Alberto Quispe Tapia	+ 30.00 + 300.00 Total: 330.00	Multa Ingresos	basica Ingreso	29 Noviembre 2022 11 Noviembre 2022

Mostrando 1 e 2 de 2 registros

Anterior **1** Siguiente

Mostrar: 10 [+agregar](#) Buscar:


DOCUMENTO DE IDENTIDAD	NOMBRE Y APELLIDOS	MONTO
422400	Andres Villca Hillaquita	- 0.00
2011546	Julian Mamani Cepha	- 0.00

Figura 3.26.

Módulo de Asistencia de Conductores mediante Placas.

Asistencias

Mostrar: 10 [+agregar](#) Buscar:

CONDUCTOR	PLACA	LINEA	DETALLES	TIPO	FECHA
Franz Guillermo Quispe Apaza	668-SDY	7e8 ROJO	MINIBUS Nissan Beige 	E	06/10/2022 10:56:50

Mostrando 1 e 1 de 1 registros

Anterior **1** Siguiente

Figura 3.27.

Módulo Vehículos.

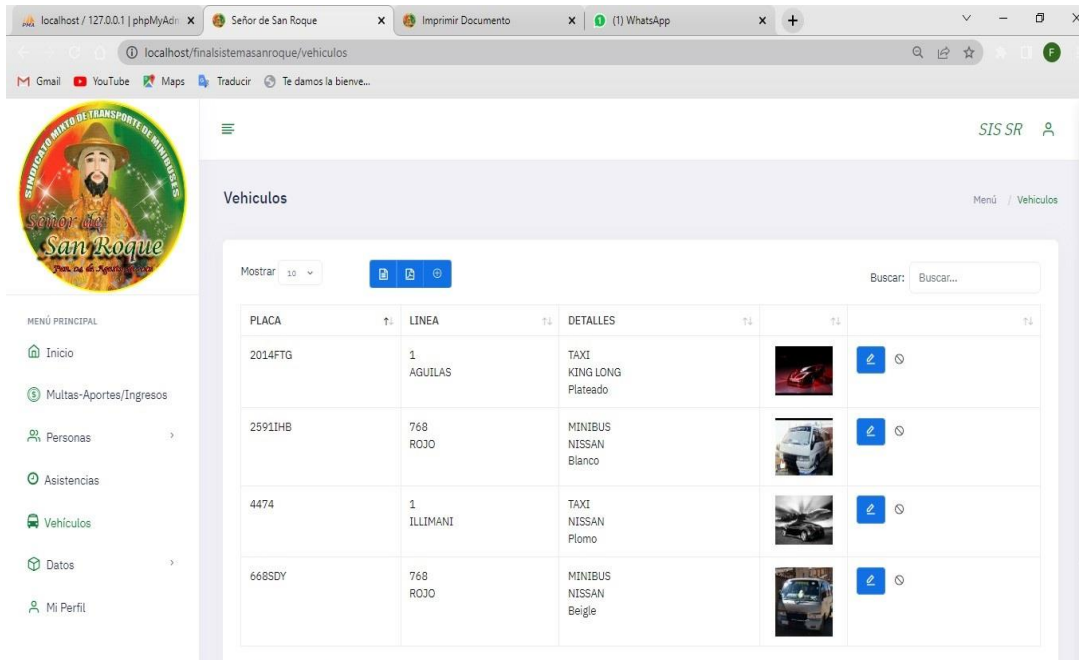


Figura 3.28.

Modulo Grupo.

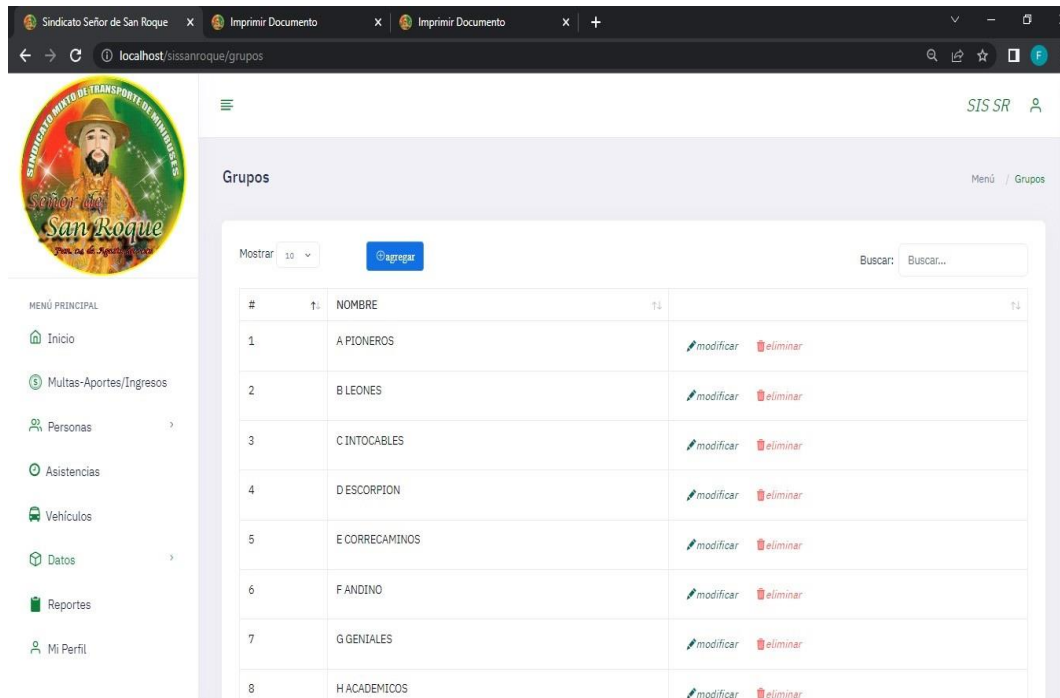


Figura 3.29.

Modulo Línea.

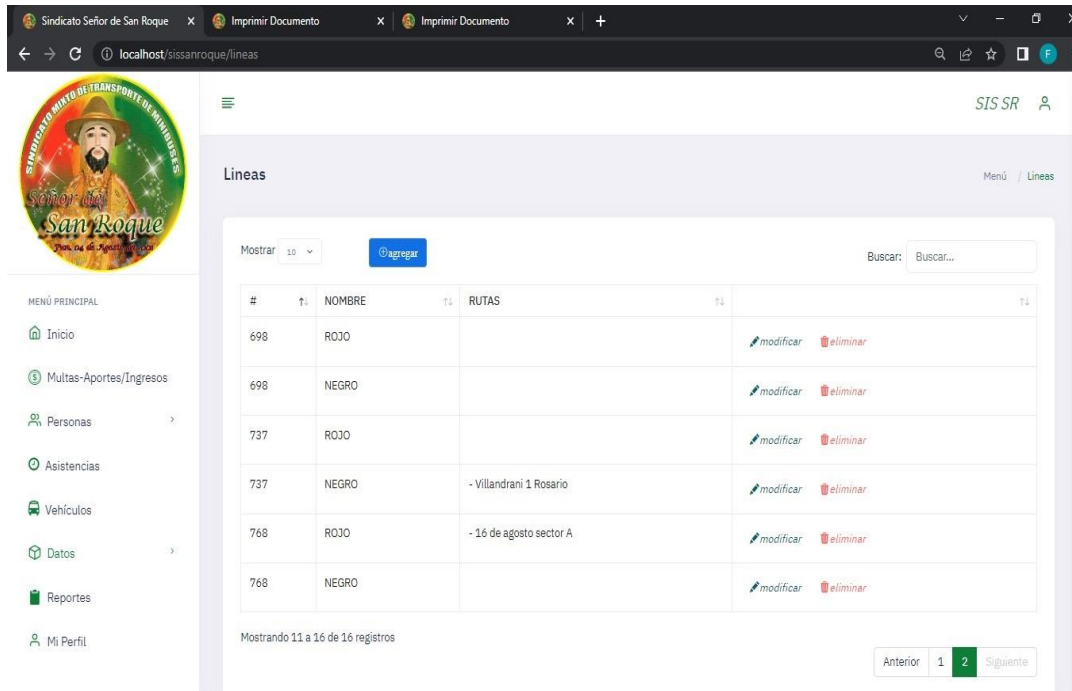
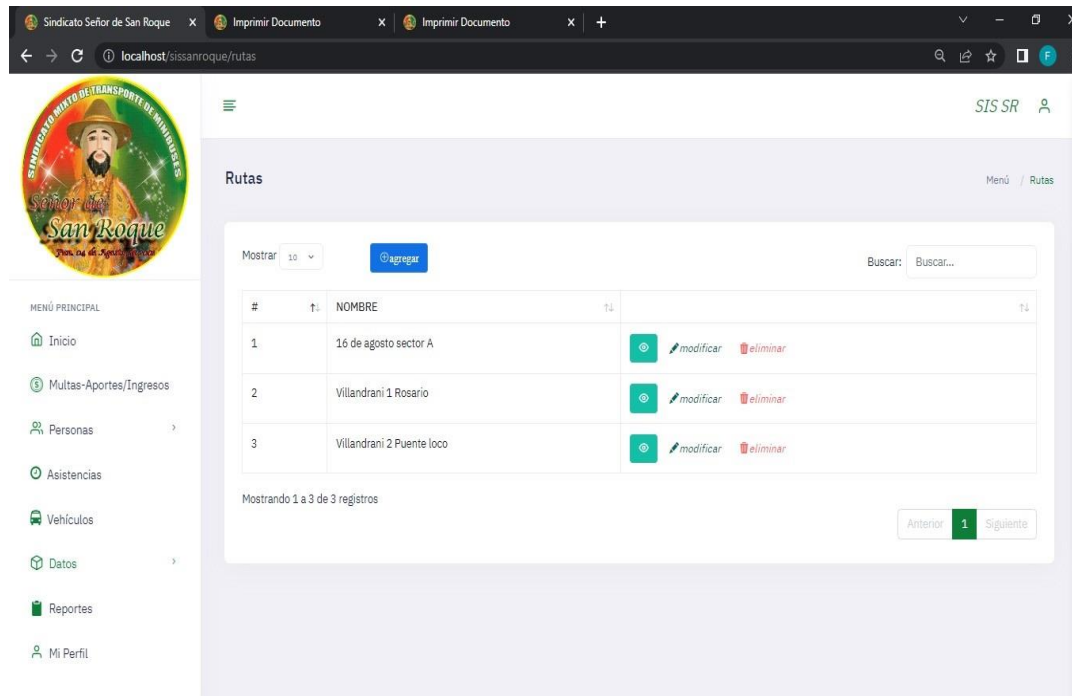


Figura 3.30.

Modulo Ruta.



3.10. Fase de Implementación del Sistema

El desarrollo de aplicaciones requiere de metodologías acorde a las características de la plataforma donde sean ejecutadas. La ingeniería web propone nuevas metodologías orientadas al desarrollo y modelación.

Sistema Web para el Seguimiento y Control de Socios Propietarios de Movilidades en el Sindicato Señor de San Roque.

Figura 3.31.

Logo del Sistema.



3.10.1. Interfaz Inicio de Sesión

- Objetivo. Estar autenticado en el sistema para poder acceder a cada módulo correspondiente.
- Descripción. Se ingresará con una cuenta registrado de tipo de usuario proporcionado por el administrador en el sistema.

Se debe ingresar al Sistema Web con una cuenta de Usuario y Contraseña proporcionados por el administrador.

Figura 3.32.

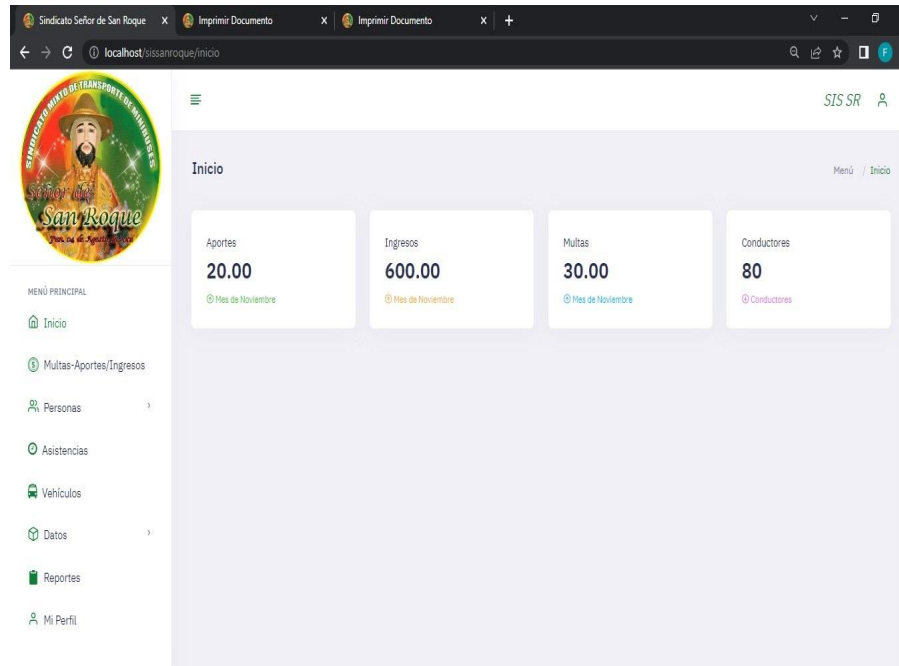
Interfaz Inicio de Sesión Login.



3.10.2. Interfaz Administrador

Figura 3.33.

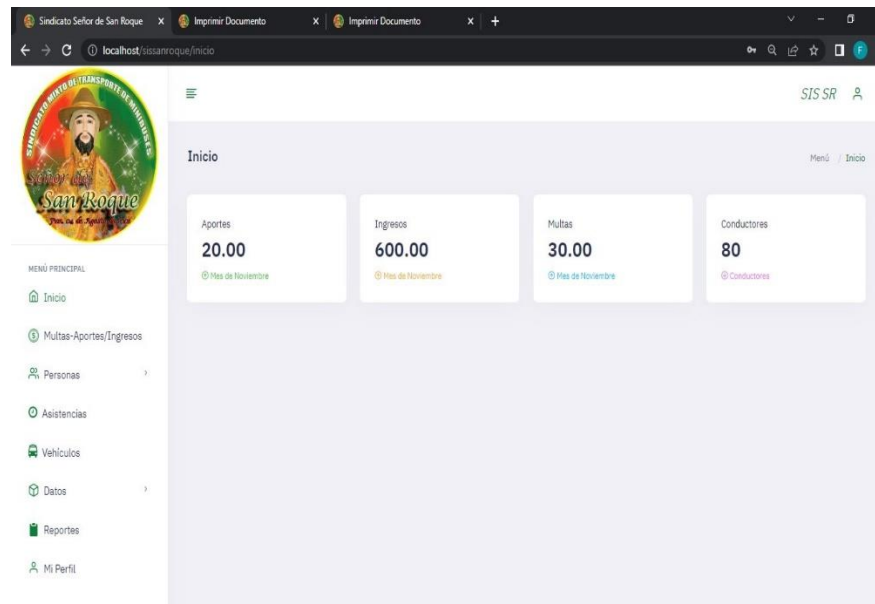
Interfaz Administrador del Sistema.



3.10.3. Interfaz Administrativo

Figura 3.34.

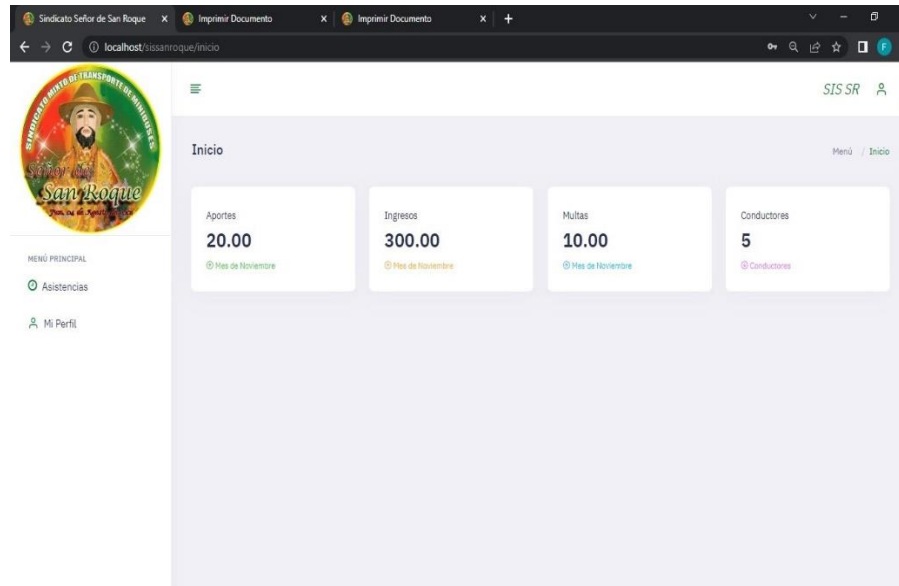
Interfaz Administrativo.



3.10.4. Interfaz Agente

Figura 3.35.

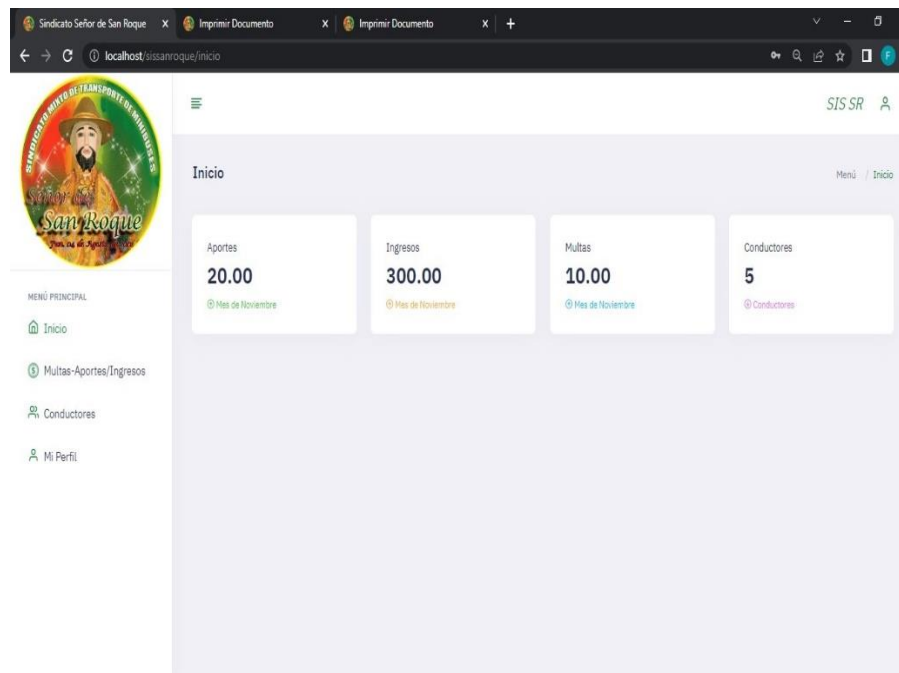
Interfaz Agente.



3.10.5. Interfaz Jefe de Grupo

Figura 3.36.

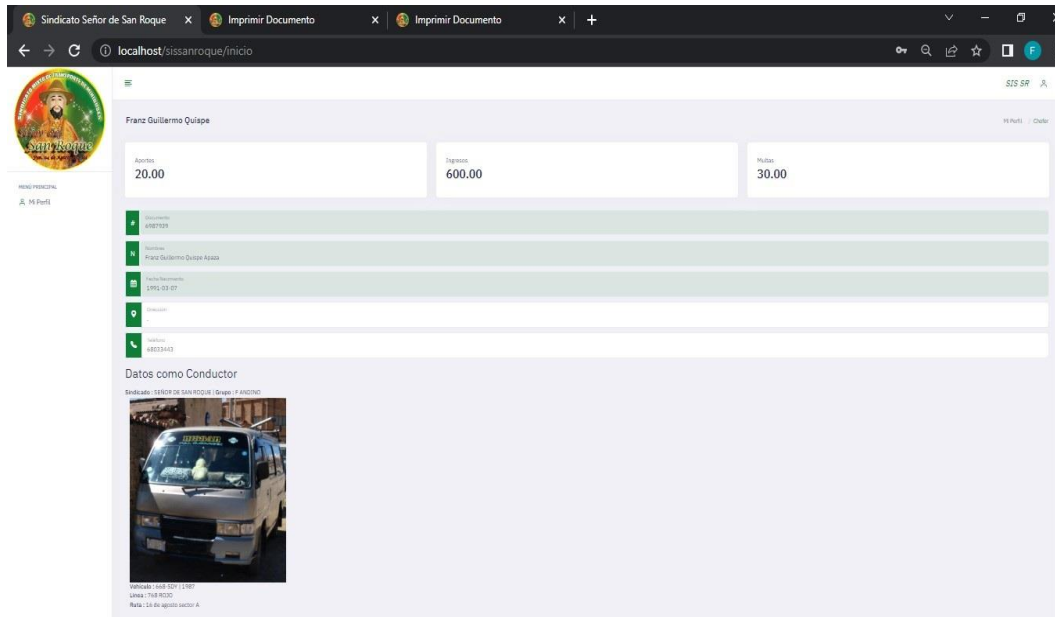
Interfaz Jefe de Grupo.



3.10.6. Interfaz Conductor

Figura 3.37.

Interfaz Conductor.



3.11. Fase de Mantenimiento

El sistema Web se ara mantenimiento cada 4 meses de la siguiente forma.

Cada uno de los tres tipos diferentes de mantenimiento de software se realiza por diferentes razones y propósitos. Es posible que una determinada pieza de software deba someterse a uno, dos o todos los tipos de mantenimiento a lo largo de su vida útil.

Los tres tipos son:

Mantenimiento correctivo de software

Mantenimiento preventivo de software

Mantenimiento perfectivo de software

Mantenimiento correctivo de software

El mantenimiento correctivo del software es la forma clásica y típica de mantenimiento (para el software y cualquier otra cosa). El mantenimiento de software correctivo es necesario cuando algo sale mal en una pieza de software, incluidos fallos y errores.

Estos pueden tener un impacto generalizado en la funcionalidad del software en general y, por lo tanto, deben abordarse lo antes posible.

Mantenimiento preventivo de software

El mantenimiento preventivo de software está mirando hacia el futuro para que su software pueda seguir funcionando como se desee durante el mayor tiempo posible.

Esto incluye realizar los cambios necesarios, actualizaciones, adaptaciones y más. El mantenimiento preventivo del software puede abordar pequeños problemas que en un momento dado pueden carecer de importancia, pero pueden convertirse en problemas mayores en el futuro. Estos se denominan fallas latentes que deben detectarse y corregirse para asegurarse de que no se conviertan en fallas efectivas.

Mantenimiento perfectivo de software

Al igual que con cualquier producto en el mercado, una vez que el software se lanza al público, surgen nuevos problemas e ideas. Los usuarios pueden ver la necesidad de nuevas características o requisitos que les gustaría ver en el software para convertirlo en la mejor herramienta disponible para sus necesidades. Es entonces cuando entra en juego el mantenimiento perfectivo del software.

El mantenimiento perfectivo de software tiene como objetivo ajustar el software agregando nuevas características según sea necesario y eliminando características que son irrelevantes o no efectivas en el software dado. Este proceso mantiene el software relevante a medida que el mercado y las necesidades del usuario cambian.

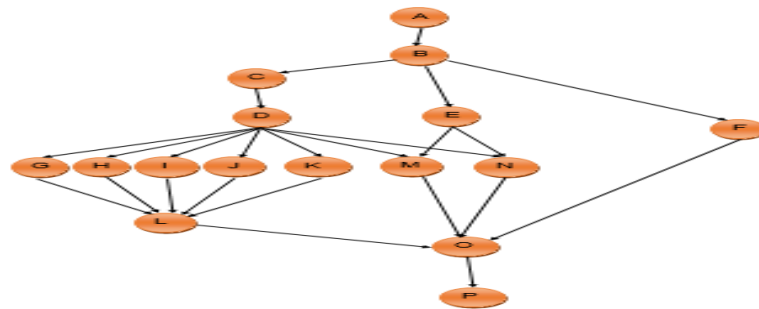
3.12.Métodos de Pruebas de Software

3.12.1.Pruebas de Caja Blanca

Esta prueba se orienta al cálculo de las regiones que deben ser consideradas como partes independientes del sistema, se realizó los pasos de las entradas del trabajo denominado Sistema Web para el Seguimiento y Control de Socios Propietarios de Movilidades, contando desde es el login que es la verificación de usuario A, hasta la salida del sistema P que se ejecutan cada una de las regiones, asegurando así que cada región se ejecuta al menos una vez. De forma general, se debe seguir:

Figura 3.38.

Pruebas de Caja Blanca.



Donde:

Inicio del Sistema (A)

Menú principal del Sistema (B)

Administrador del General del Sindicato (C)

Modulo del Personal Administrativo (D)

Modulo Jefe de Grupo (E)

Módulo de Conductores (F)

Módulo de Vehículos (G)

Módulo de Grupo (H)

Módulo de Líneas (I)

Módulo de Rutas (J)

Módulo de Aportes y Multas Financiamiento (K)

Reportes (L)

Generar Solicitud de Refuerzo (M)

Generar Memorándum de trabajo conductor (N)

Fin ciclo de Sistema (O)

Fin del Sistema (P)

Analizado el grafo generado a partir de las características del sistema, ahora se procede a determinar.

La Complejidad Ciclo matica de acuerdo a:

Numero de Nodos = 16

Número de Artistas = 24

$$V(G) = A - N + 2$$

Donde:

N= Numero de Nodos.

A= Numero de aristas

Remplazando los Valores tenemos:

$$V(G) = A - N + 2$$

$$V(G) = 24 - 16 + 2$$

$$V(G) = 10$$

Determinar el conjunto básico de caminos linealmente independientes. Los caminos que deben ser probados dadas ciertas variables son 10. Estos caminos son los siguientes.

Sacamos Caminos independientes:

Camino 1: A – B – C – D – G – L – O – P

Camino 2: A – B – C – D – H – L – O – P

Camino 3: A – B – C – D – I – L – O – P

Camino 4: A – B – C – D – J – L – O – P

Camino 5: A – B – C – D – K – L – O – P

Camino 6: A – B – C – D – M – O – P

Camino 7: A – B – C – D – N – O – P

Camino 8: A – B – E – M – O – P

Camino 9: A – B – E – N – O – P

Camino 10: A – B – F – O – P

3.12.2.Pruebas de Caja Negra

Las pruebas de caja negra o también conocidas como pruebas de comportamiento se centran en los requisitos funcionales del software. Para realizar la prueba de caja negra se realiza las pruebas a la interfaz mostrada a continuación.

Figura 3.39.

Pruebas de Caja Negra Iniciar Sesión.



Tabla 3.10

Valores de Limites de Inicio de Sesión.

Campos	Entrada Valida	Entrada Invalida
Documento de Identidad	Cadena de texto o número.	Caracteres especiales, espacios en blanco.
Contraseña	Cadena de texto o número.	Caracteres especiales, espacios en blanco.

Tabla 3.11

Pruebas de Caja Negra Inicio de Sesión.

Entradas		Salidas	Resultados
Usuario	Contraseña		
7410852	admin_7410852	Ingreso Usuario y Contraseña	El sistema valida que no se ingresen datos en blanco.
		Ingresar al Sistema	Introducir los datos validos al sistema concede el acceso al mismo.

3.12.3.Pruebas de Funcionalidad

Tabla 3.12

Pruebas de Funcionalidad.

Procedimiento	Descripción	Valor
Prueba Requerida	Registro de Usuario	Si
Usuario	Administrador, Administrativo Agente, jefe de grupo, conductor.	
Secuencia de Prueba		
Procedimiento	Resultados	Clasificación de Funcionalidad
Para Ingresar al sistema debe ingresar con su documento de identidad como usuario y la contraseña.	El sistema Valida los datos una vez que sean correctos e ingresa al sistema, si lo datos son incorrectos le mostrar un mensaje.	Si
Fallas	Descripción	
Ninguna	Ninguna	
Pruebas	Resultados	Positivo Negativo
Ingresar al sistema con su documento de identidad como usuario y su contraseña.	El usuario ingresara al sistema si los datos son correctos, y según el grado de privilegios que tenga.	si
El administrador puede registrar a un nuevo usuario o Conductor.	El administrador debe tener acceso a la modificación de datos del personal de usuario y Socios Propietarios del sistema.	si
Una vez que se ingresa al sistema comprueba que tenga acceso a todas las áreas que puede realizar según sus privilegios.	El usuario debe tener acceso según su privilegio.	si

CAPÍTULO IV
METRICAS DE
CALIDAD, ESTIMACION
DE COSTOS Y
SEGURIDAD

4.1.Introducción

En este capítulo IV presentaremos la calidad del sistema con el objetivo de que la aplicación obtenga la calidad y seguridad del sistema, este sea necesaria y suficiente para llegar a satisfacer las necesidades del usuario final.

4.2.Calidad de Software

4.2.1.Funcionalidad

Tabla 4.1

Numero de Entrada de Usuario.

Parámetros de Medición	Cuenta
Número de Entradas de Usuario	900
Número de Salidas de Usuario	904
Numero de petición de Usuario	28
Numero de archivos	900
Numero de Interfaces externas	0

Para calcular los puntos de función se tiene que hacer un cálculo de la cuenta total en la siguiente tabla:

Tabla 4.2*Calculo de Funcionalidad de Punto de Función.*

Parámetros de Medición	Cuenta	Factor de Complejidad			Total
		Simpe	Medio	Complejo	
Número de Entradas de Usuario	900	3	4	6	2700
Número de Salidas de Usuario	904	4	5	7	3616
Numero de petición de Usuario	28	3	4	6	84
Numero de archivos	900	7	10	15	6300
Numero de Interfaces externas	0	5	7	10	0
Cuenta Total					12,700

Para determinar los valores de ajuste de complejidad según las respuestas a las siguientes preguntas que se muestra en la siguiente tabla:

Tabla 4.3

Factores de Complejidad.

	Complejidad						Fi
	Sin	Incidencia	Moderado	Medio	Significativo	Esencial	
	0	1	2	3	4	5	
1. ¿Requiere el sistema copias de seguridad y de recuperación fiables?						X	5
2. ¿Se requiere comunicación de datos?						X	5
3. ¿Existen funciones de procesos distribuidos?					X		4
4. ¿Es crítico el rendimiento?				X			3
5. ¿El sistema web se ejecuta en un entorno operativo existente y fuertemente utilizado?						X	5
6. ¿Requiere el sistema entrada interactiva de datos?						X	5
7.- ¿Se requiere que el sistema tenga entradas a datos con múltiples ventanas?					X		4
8. ¿Se actualizan los archivos maestros de forma interactiva?				X			3
9. ¿Son complejos las entradas, las salidas, los archivos o las peticiones?					X		4
10. ¿Es complejo el procesamiento interno del sistema?					X		4
11. ¿Se ha diseñado o utilizado el código para ser reutilizable?						X	5
12. ¿Se ha diseñado el sistema web o aplicación para facilitar al usuario y para ser fácilmente utilizado?						X	5
Factor de Complejidad Total							52

Calculando la siguiente ecuación:

$$PF = CuentaTotal \times (0.65 + 0.1 \times \sum Fi)$$

$$PF = 12.700 \times (0.65 + 0.01 \times 52)$$

$$PF = 12.700 \times (1.17)$$

$$PF = 14.859$$

Asemos una comparación para hallar el punto ideal al 100% de los valores de complejidad al máximo total el valor de $\sum Fi = 60$:

$$PFmax = CuentaTotal \times (0.65 + 0.01 \times \sum Fi)$$

$$PFmax = 12.700 \times (0.65 + 0.01 \times 60)$$

$$PFmax = 12.700 \times (1.25)$$

$$PFmax = 15.875$$

Calculando de % de funcionalidad real:

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

$$Funcionalidad = \frac{14.859}{15.875} * 100\%$$

$$Funcionalidad = 0.936 * 100\%$$

$$Funcionalidad = 93.6\%$$

El sistema Web tiene una funcionalidad del 93.6% esto requiere decir que el sistema tiene redondeando con un 94 % de funcionamiento para la Institución Señor de San Roque, cumple con los requisitos.

4.2.2.Confiabilidad

La Confiabilidad tiene la siguiente formula de la ecuación:

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

Para lo que consideramos un periodo de 20 días como tiempo de prueba donde se define que cada 10 ejecuciones se presenta 1 falla conociendo la funcionalidad de 93.6% del sistema.

Calculando la siguiente ecuación:

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

$$F(t) = 0,936 * e^{(-\frac{1}{10} * 20)}$$

$$F(t) = 0.127\%$$

Reemplazando en las fórmulas de probabilidades

$$P(T \leq t) = F(t) \Rightarrow 0.127 = 12.7\%$$

$$P(T \leq t) = 1 - F(t) \Rightarrow 1 - 0.127 = 0.873 * 100\% = 87.3\%$$

Por lo tanto, la confiabilidad del sistema es de 87,3 % en un periodo de 20 días como tiempo de prueba.

4.2.3. Usabilidad

La Usabilidad tiene la siguiente fórmula de la ecuación:

$$FU = \left[\sum \left(\frac{x_i}{n} \right) * 100 \right]$$

Tabla 4.4

Escala de Valores de Preguntas.

Usabilidad	Valoración
Excelente	6
Muy bueno	5
Bueno	4
Regular	3
Malo	2
Pésimo	1

Tabla 4.5*Preguntas para determinar la Usabilidad del Sistema.*

Nro.	Preguntas	Si	No	Ponderación
1	¿Puede Utilizar con facilidad el sistema?	6	0	1
2	¿El sistema es comprensible?	5	1	0.835
3	¿El sistema cuenta con interface agradable a la vista?	6	0	1
4	¿La respuesta del sistema es satisfactoria?	5	1	0.835
5	¿Le parecen complicada las funciones del sistema?	4	2	0.668
6	¿Es fácil aprender a manejar el sistema?	5	1	0.835
7	¿Se hace dificultoso aprender a manejar el sistema?	4	2	0.668
8	¿El sistema satisface las necesidades que usted requiere?	6	0	1
9	¿Durante el uso del sistema se produjo errores?	5	1	0.835
Total				7.676

Calculamos la Usabilidad de la siguiente ecuación:

$$FU = [\sum(\frac{x_i}{n}) * 100]$$

$$FU = [\sum(\frac{7.676}{9}) * 100]$$

$$FU = [0.85 * 100]$$

$$FU = 85\%$$

De acuerdo a los datos obtenidos en la tabla existe un 85% de entendimiento y comprensión de los usuarios respecto al trabajo realizado del sistema.

4.2.4.Mantenibilidad

La Mantenibilidad tiene la siguiente formula de la ecuación:

$$IMS = \frac{Mt - (Fc + Fa + FE)}{Mt}$$

Calculando:

$$IMS = [8 - (0 + 2 + 0)] / 8$$

$$IMS = [0.85 * 100]$$

$$IMS = 85\%$$

Por lo tanto el resultado que establece es de 85% del sistema, lo cual no requiere mantenimiento.

4.2.5.Portabilidad

La Portabilidad tiene la siguiente formula de la ecuación:

$$P = 1 - EP / EI$$

Dónde:

EP=0.5 días portar el sistema

EI= 2,5 días implementar el sistema

Remplazando en la formula tenemos:

$$P = 1 - EP / EI$$

$$P = 1 - 0.5 / 2.5$$

$$P = 0.80$$

Lo que significa que existe un 80% de que el usuario instale el software con éxito, entonces la portabilidad es óptima.

4.2.6.Resultado Final

Tabla 4.6

Resultado Total.

Factor	Resultado
Funcionalidad	93.6
Usabilidad	85
Mantenibilidad	85
Portabilidad	80
Confiabilidad	87.3
Evaluación de Calidad Total	86.18

Entonces podemos ver que de un usuario que acceda al sistema tendrá una satisfacción del 86 % al utilizarla.

4.3.Estimación de Costos

De esta manera, el objetivo de la estimación de costo de software consiste en estimar el tamaño, el esfuerzo, la complejidad y el costo del proyecto para poder encontrar la mejor decisión de desarrollo y asegurar que el gasto se encuentre de acuerdo con lo presupuestado.

A continuación se describe los componentes relacionales con su complejidad asignada a cada uno de los factores que se deben considerar para la estimación del proyecto.

Las ecuaciones que se utilizan en los tres modelos son:

En Persona – Mes

Ecuación 1.

$$E = a (kl^{(b)} * m^{(x)})$$

En Meses:

Ecuación 2.

$$Tdev = c (E^d)$$

En Personas:

Ecuación 3.

$$P = E / Tdev$$

Donde:

E: es el esfuerzo requerido por el proyecto, en persona-mes

Tdev: es el tiempo requerido por el proyecto, en meses

P: es el número de personas requerido por el proyecto

a, b, c y d son constantes con valores definidos en una tabla, según cada submodelo

KI: es la cantidad de líneas de código, en miles.

m(X): Es un multiplicador que depende de 15 atributos.

Para calcular el esfuerzo, es necesario hallar la variable **KI o KLDC** (cantidad de líneas de código en miles), donde los PF son 14.859, las líneas por cada PF equivalen a 720.

Tras saber que 720 por cada LDC por cada PF, por ser PHP, el resultado será el siguiente:

$$LDC = PF_{obtenido} * FactorLDC / PF$$

$$LDC = 14.859 * 720 / 1000 \Rightarrow 10.698$$

Ya que el número de líneas de código no supera los 50 KLDC, los coeficientes que utilizaremos se la siguiente tabla, será de la fila de orgánico 2.40.

4.3.1. Método de Estimación COCOMO

Tabla 4.7

Coficiente de a, b, c, d COCOMO.

MODO	A	B	C	D
Orgánico	2.40	1.05	2.50	0.38
Semi – Orgánico	3.00	1.12	2.50	0.35
Empotrado	3.60	1.20	2.50	0.33

Tabla 4.8*Valores de coste COCOMO DETALLADO*

Atributos	Valor					
	Muy Bajo	Bajo	Nominal	Alto	Muy Alto	Extra Alto
Atributos de Software						
Fiabilidad	0,75	0,88	1,00	1,15	1,40	
Tamaño de Base de Datos		0,94	1,00	1,08	1,16	
Complejidad	0,70	0,85	1,00	1,15	1,30	1,65
Atributos de Hardware						
Restricciones de tiempo de ejecución			1,00	1,11	1,30	1,66
Restricciones de memoria virtual			1,00	1,06	1,21	1,56
Volatilidad de la máquina virtual		0,87	1,00	1,15	1,30	
Tiempo de respuesta		0,87	1,00	1,07	1,15	
Atributos de Personal						
Capacidad de análisis	1,46	1,19	1,00	0,86	0,71	
Experiencia en la aplicación	1,29	1,13	1,00	0,91	0,82	
Calidad de los programadores	1,42	1,17	1,00	0,86	0,70	
Experiencia en la máquina virtual	1,21	1,10	1,00	0,90		
Experiencia en el lenguaje	1,14	1,07	1,00	0,95		
Atributos del Proyecto						
Técnicas actualizadas de programación	1,24	1,10	1,00	0,91	0,82	
Utilización de herramientas de software	1,24	1,10	1,00	0,91	0,83	
Restricciones de tiempo de desarrollo	1,24	1,10	1,00	1,04	1,10	

Fuente: Boehm 1989.

$$FAE=1.00*1.00*0.85*1.00*1.00*1.15*1.00*0.71*0.91*1.00*0.90*1.00*1.00*0.91*1.00$$

$$FAE=0.51725$$

Por tanto, nuestro Factor de ajuste será $FAE=0.51725$

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

Calculo de esfuerzo desarrollo, en Persona – Mes:

Ecuación 1.

$$E = a (kl^{(b)} * m^{(x)}) \text{ (Personas/Mes)}$$

$$E = 2.4 (10.698^{(1.05 * 0.51725)})$$

$$E = 8.694 \text{ (Personas Mes)}$$

$$E = 9 \text{ (Personas Mes)}$$

Cálculo del Tiempo de desarrollo, en Meses:

Ecuación 2.

$$Tdev = c * E^d \text{ (Meses)}$$

$$Tdev = 2.50 * 8.694^{0.38}$$

$$Tdev = 5.686 \text{ (Meses)}$$

Calculo del personal requerido, En personas:

Ecuación 3.

$$P = E / Tdev \text{ (Personas)}$$

$$P = 8.694 / 5.686 \text{ (Personas)}$$

$$P = 1.529 \text{ (Personas)}$$

$$P = 1.559 \text{ (Equivale a Personas)}$$

- **Costo en el personal**

Suponiendo que el personal se les pague un promedio de 400 \$us/mes y se trabaje los 6 meses entonces el costo del proyecto será:

$$\text{Costo} = 400 * 6 * 2 \text{ (salario * meses trabajo * personas)}$$

$$\text{Costo} = 4.800 * 7 = 33600$$

- **Costo en la elaboración del proyecto**

Gastos realizados en diferentes fases de la elaboración del proyecto.

Tabla 3.16*Costo de elaboración del proyecto*

Detalle	Importe (bs)
Internet	300
Investigacion	90
Total	390

•Costo total del software

El costo total del software es la sumatoria, costo de desarrollo, implementación y elaboración del proyecto. Lo veremos en la siguiente tabla con una cantidad aproximada.

Tabla 3.17*Costo total del proyecto*

DETALLE	IMPORTE (bs)
Costo de desarrollo	33600
Costo de elaboración del proyecto	390
Total	33990bs

4.4.Seguridad del sistema**4.4.1.Uso de Password**

Calcula un hash con el algoritmo md5 devolviendo un hash de 32 caracteres hexadecimal.

```
function contrasena($password)
{
return substr(md5($password), -5).md5($password);
}
```

4.4.2. Encriptación de datos

Las contraseñas de los usuarios están encriptados con `aes_encrypt`, de esta forma se tiene asegurado que las acciones solamente sean únicamente responsables.

```
INSERT INTO `tr_usuarios` (`id`, `id_persona`, `contrasena`, `nivel_acceso`,  
`id_grupo`, `estado`)
```

```
VALUES (4, 15, aes_encrypt(' 5d93f6755fdb32fb3a75bd546021a6415d93f ', '12345'),  
0, 2, 1);
```

-Para descifrar el password debemos conocer la clave de cifrado: '12345':

```
select cast(aes_decrypt(contrasena, '12345') as char) from tr_usuarios;
```

4.4.3. Seguridad de la base de datos

Hacer una backup de la base de datos

Figura 4.1.

Backup de la base de datos.



Nota. Hacemos un backup en la base de datos para no perder los registros de los conductores afiliados en la Institución Señor de san Roque.

CAPÍTULO V
CONCLUSIONES Y
RECOMENDACIONES

5.1.Conclusiones

Llegamos a la conclusión de que el objetivo general del proyecto se ha logrado satisfactoriamente con el desarrollo del sistema web para el seguimiento y control de socios propietarios de movilidades en el Sindicato Señor de san Roque, utilizando las herramientas de diseño y desarrollo propuestos para la culminación del trabajo final.

- ✓ Se ha logrado fundamentar los conocimientos teóricos de la Metodología UWE de acuerdo al requerimiento de la Institución, la calidad del costo de software usando el COCOMO y para la seguridad del aplicativo se utilizó la encriptación de contraseñas con MD5.
- ✓ Se logró Satisfactoriamente realizar un diagnóstico para la afiliación de los conductores mediante el aplicativo de forma rápida y eficaz.
- ✓ Se ha Logrado realizar todas las tareas o módulos mencionados, con el sistema ya implementado se puede realizar búsquedas y consultas de los registros existentes de los jefes de grupos y los conductores así también hacer un control y seguimiento de cada socio propietario de la Institución.

5.2.Recomendaciones

- ✓ Para posteriores versiones se recomienda a futuro ampliar el sistema con más módulos de trabajos realizados con los directorios del sindicato, los directorios de jefes de grupos, un módulo para la generación de planillas para el control del personal, para así tener una información centralizada de todos los procesos que se realizan en la Institución.
- ✓ Seguir mejorando la productividad del trabajo de investigación para hacer más eficiente y relevante su uso.

Bibliografía

- [González Lozano, 2018] *Desarrollo web con PHP y MySQL. 5ta Edición*
- [Pressman, 2002] *Pressman Roger S, 2002 Ingeniería De Software Un Enfoque Practica. 5ta Edición Madrid España.*
- [Senn, 1999] *Análisis Y Diseño De Sistemas De Información 2da Edición*
- Albertof. (13 de julio de 2021). *CSS3*. Obtenido de <https://graficofajardo.es/que-es-css3/>
- Alegsa. (27 de agosto de 2018). *Grafico general de un Sistema*. Obtenido de <https://www.alegsa.com.ar/Dic/sistema.php>
- Alegsa.com.ar. (27 de agosto de 2018). *Definicion de Sistema*. Obtenido de <https://www.alegsa.com.ar/Dic/sistema.php>
- Arias Chavez, M. (16 de junio de 2006). *Ingenieria de Requerimientos* . Obtenido de <https://webcache.googleusercontent.com/search?q=cache:FqLYXDkemDAJ:https://revistas.ur.ac.cr/index.php/intersedes/article/download/790/851/+&cd=3&hl=es&ct=clnk&gl=bo>
- Arimetrics. (2022). *Arimetrics Bootstrap*. Obtenido de <https://www.arimetrics.com/glosario-digital/bootstrap>
- Arimetrics. (2022). *Arimetrics JQuery*. Obtenido de <https://www.arimetrics.com/glosario-digital/bootstrap>
- Arimetrics. (2022). *Arimetrics HTML5*. Obtenido de <https://www.arimetrics.com/glosario-digital/html5>
- Blog, R. (20 de Abril de 2018). *Requerimientos Funcionales y No Funcionales*. Obtenido de <https://medium.com/@requeridosblog/requerimientos-funcionales-y-no-funcionales-ejemplos-y-tips-aa31cb59b22a>
- Chuburu, L. (2020). *Jquery*. Obtenido de <https://www.laurachuburu.com.ar/tutoriales/que-es-jquery-y-como-implementarlo.php>
- Chuquimia, G. (s.f.). *Factores de calidad ISO 9126*. <http://virtual.usalesiana.edu.bo/web/conte/archivos/303.pdf>.
- Cruz Perez, M. R. (2009). *Aplicación de la Norma Aplicación de la Norma ISO 9126*. <https://fit.um.edu.mx/CI3/publicaciones/TechnicalReportCOMP-022-2009.pdf>.
- Cruz Sirpa, M. A. (2020). *Modelo de realidad aumentada para el control de tramos del transporte público basado en android y código qr” caso: sindicato “simon bolivar*. Obtenido de <http://repositorio.upea.bo/bitstream/123456789/113/1/TESIS-Maribel%20Ana%20Cruz%20Sirpa.pdf>
- faraaz. (15 de diciembre de 2020). *Cara Hash String Menggunakan MD5 di PHP*. Obtenido de <https://inwepo.co/cara-hash-string-menggunakan-md5-di-php/>

- Guzman, D. (28 de junio de 2018). *JSON*. Obtenido de <http://codingornot.com/que-es-json>
- James, S. (1992). *Análisis y Diseño de Sistemas de Informacion*. Obtenido de <https://www.monografias.com/trabajos29/ciclo-sistema/ciclo-sistema>
- JavaScript desarrolloWeb.com*. (s.f.). Obtenido de <https://desarrolloweb.com/home/javascript>
- JSON desarrolloweb.com*. (s.f.). Obtenido de <https://desarrolloweb.com/home/json>
- Lazaro, D. (2018). *Encriptacion y contraseñas en PHP*. <https://diego.com.es/encriptacion-y-contrasenas-en-php>.
- Mamani Condori, W. (2020). *Sistema Web de Control y Seguimiento de servicios y Gestion de Clientes para la Empresa Consultora Contadores Publicos*. Obtenido de <https://repositorio.umsa.bo/bitstream/handle/123456789/27967/PG-3655.pdf?sequence=1&isAllowed=y>
- Marcago. (02 de Abril de 2022). *Diseño Web Corporativo HTML5*. Obtenido de <https://marcago.com/disenio-web/html5/>
- Mestras, P. (2012). *Servidores Web Apache*. Obtenido de <https://www.fdi.ucm.es/profesor/jpavon/web/31-ServidoresWeb-Apache.pdf>
- Mita, R. M. (2015). *Sistema web de gestión e información para control y seguimiento caso: cooperativa de transporte de turismo “3 de mayo” Ltda*. Obtenido de <https://repositorio.umsa.bo/xmlui/bitstream/handle/123456789/7555/T.2959.pdf?sequence=1&isAllowed=y>
- Nieves Guerrero, C. G., Ucan Pech, J. P., & Mendez Dominguez, V. H. (2014). *UWE en Sistema de Recomendación de Objetos de Aprendizaje. Aplicando Ingeniería Web: Un Método en Caso de Estudio*. Obtenido de https://www.researchgate.net/publication/280580830_UWE_en_Sistema_de_Recomendacion_de_Objetos_de_Aprendizaje_Aplicando_Ingenieria_Web_Un_Metodo_en_Caso_de_Estudio
- Olsina, R. (2008). *Metodologia Uwe*.
- Pelissier, C. (2013). *Programacion con PHP*. Obtenido de <http://profesores.elo.utfsm.cl/~agv/elo330/2s02/projects/pelissier/informe.html>
- Primaria. (1 de mayo de 2022). *COCOMO II Coeficiente el modelo Basico*. Obtenido de <https://es.wikipedia.org/wiki/COCOMO>
- Ramirez Cortes, J. H. (2022). *Desarrollo e Implementacion de un Sistema de Seguimiento y Control del Servicio Vehicular en la Estacion la Juanita, Bogota*. Obtenido de https://repository.ucc.edu.co/bitstream/20.500.12494/46441/7/2022-Ramirez_Desarrollo_Sistema_Seguimiento.pdf
- Rivera, B. (2018). *El control Administrativo. Santiago de chile*. Obtenido de

- <http://revistaderecho.um.edu.uy/wp-content/uploads/2013/02/Bejar-Rivera-yOrrico-Galvez-El-Control-Administrativo-en-Mexico.pdf>.
- Robledano, A. (2019). *OpenWebinars Desarrollo Web*. Obtenido de <https://openwebinars.net/blog/que-es-css/>
- Roque, V. (2013). *Definicion de la metodologia UWE*. Obtenido de <https://es.scribd.com/document/173718395/Definicion-de-Metodologia-UWE>
- Salas, M. (s.f.). *Gestor de Base de Datos MariaDB*. Obtenido de <https://www.hostinglatam.cl/caracteristicas-de-mariadb-un-proyecto-derivado-de-mysql/>
- Scotland Muñoz, J. A. (2020). *Sistema de Gestion Web para el Control Administrativo de Socios y Unidades de Transporte Cooperativa de taxis Taxibucay*. Obtenido de [https://cia.uagraria.edu.ec/Archivos/SCOTLAND%20MU%C3%91OZ%20JOSHEP%20ANTONNY_compressed%20\(wecompress.com\).pdf](https://cia.uagraria.edu.ec/Archivos/SCOTLAND%20MU%C3%91OZ%20JOSHEP%20ANTONNY_compressed%20(wecompress.com).pdf)
- stack, o. (s.f.). *Aprendizaje sublimetext3*. <https://riptutorial.com/Download/sublimetext3-es.pdf>.
- Tenazoa Cuba, K. C. (2022). *Implementacion de un Sistema Web para el Control de Unidades de una Empresa de Transporte Urbano*. Obtenido de <http://repositorio.ulasamericas.edu.pe/bitstream/handle/upa/2047/1.%20EJEMPLAR%20TESIS%20-%20KEVIN%20CARLOS%20TENAZOA%20CUBA.pdf?sequence=1&isAllowed=y>
- Uriarte, J. M. (26 de Agosto de 2021). *Sistema de Informacion*. Obtenido de <https://www.caracteristicas.co/sistema-de-informacion/>

ANEXOS

ANEXO A

ARBOL DE

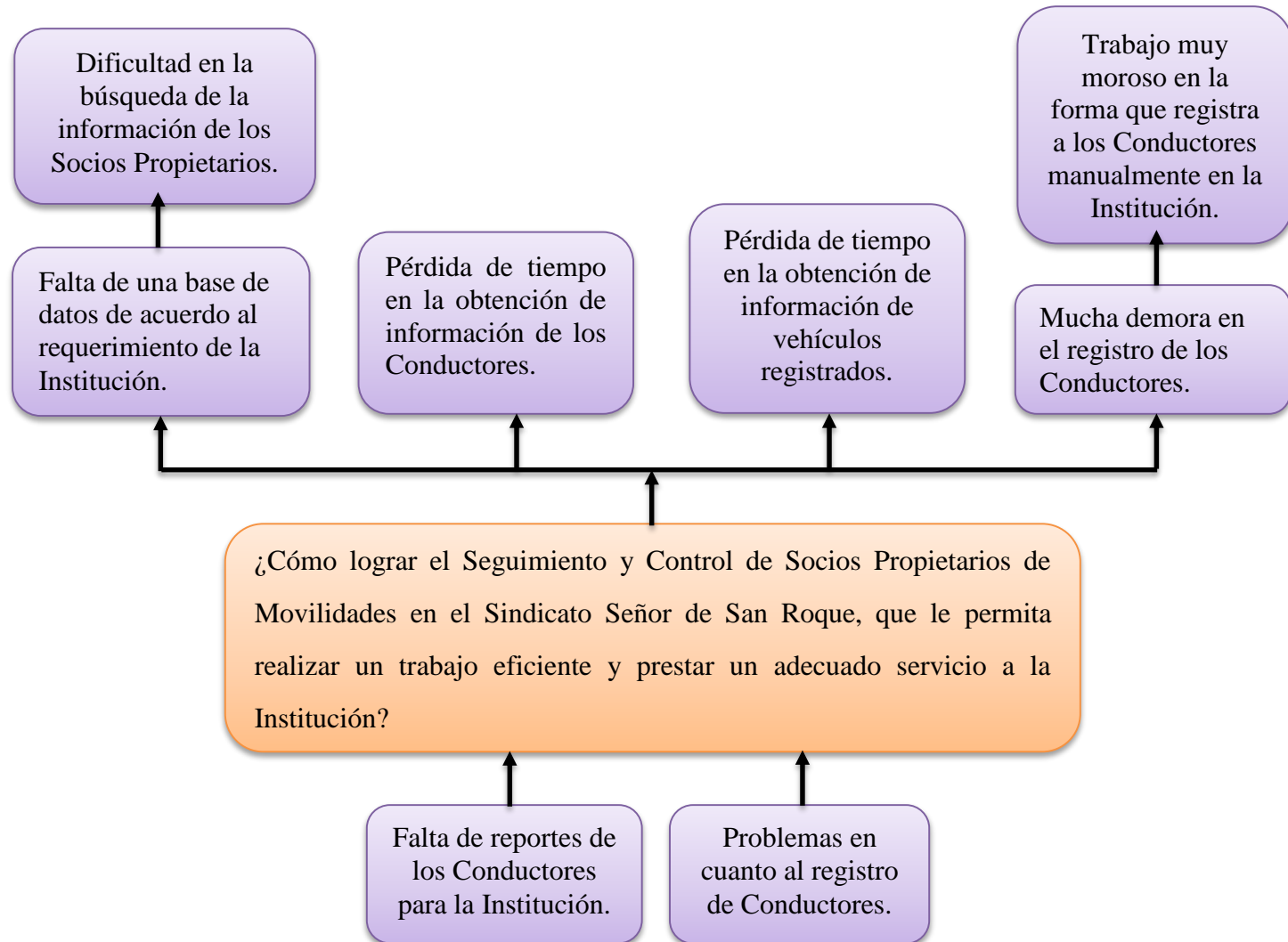
PROBLEMAS

ANEXO B

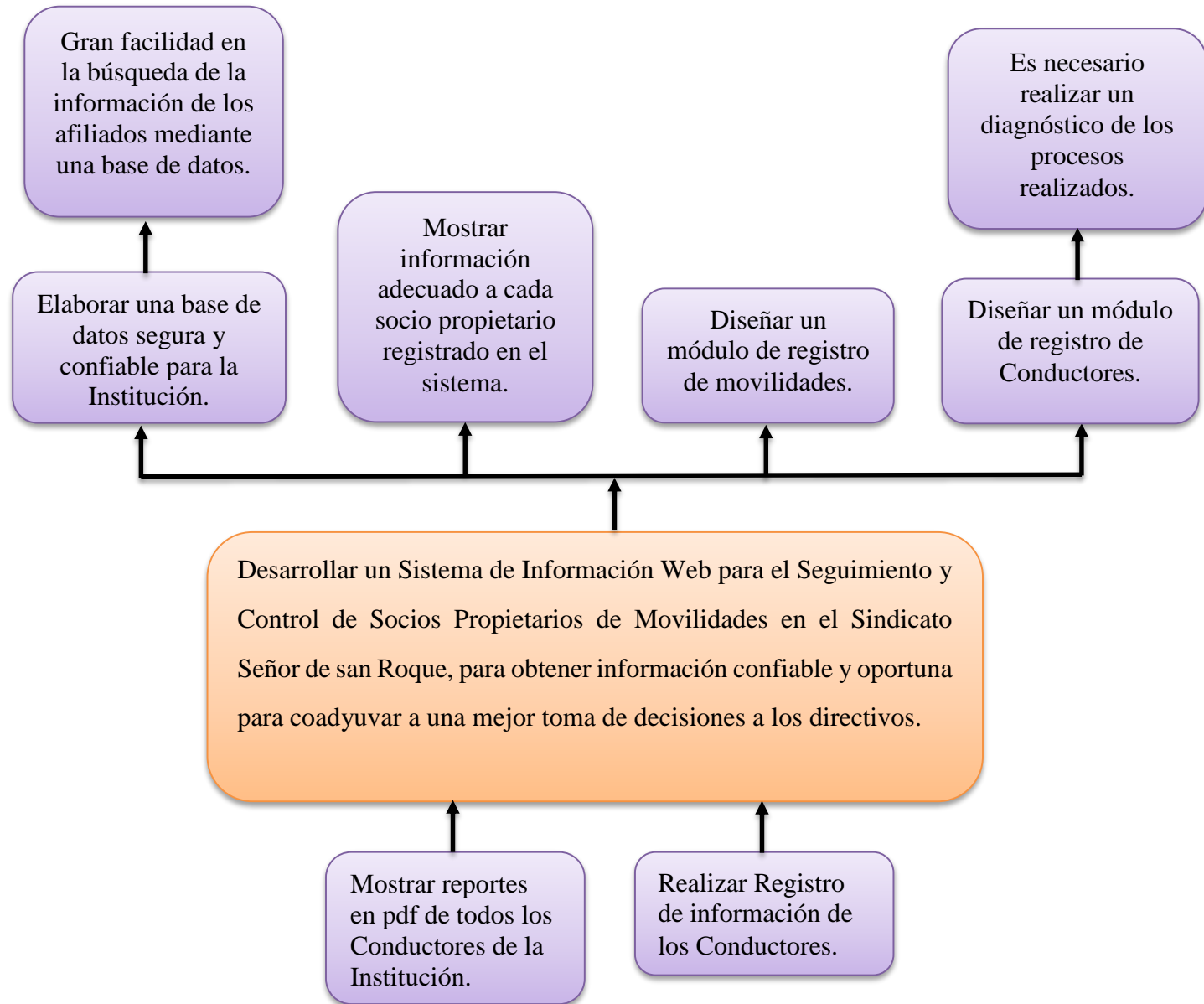
ARBOL DE

OBJETIVOS

Anexo A. Árbol de Problema



Anexo B. Árbol de Objetivos



ANEXO C
DOCUMENTACIÓN
DE LA
INSTITUCIÓN

DOCUMENTACIÓN

Sindicato

Señor de San Roque



**SISTEMA WEB PARA EL
SEGUIMIENTO Y CONTROL DE
SOCIOS PROPIETARIOS DE
MOVILIDADES**

FORMULARIO
DE LA ENTREVISTA DEL
SINDICATO SEÑOR DE SAN ROQUE

Nombre:.....

Cargo:.....

Fecha:.....

1. ¿Cómo se maneja la información en cuanto al registro de los conductores de la Institución?

2. ¿De qué manera hacen el registro de cada conductor y como se sabe que se encuentra en la Institución?

3. ¿Cómo se procede la búsqueda de los conductores?

4. ¿Cómo se procede la búsqueda de las movilidades?

5. ¿Cómo se realiza el registro de nuevos socios Propietarios a la Institución de Conductores y Movilidades ? y quienes se encargan de registrarlos

6. ¿Cómo se puede saber que un conductor está en tal grupo?

7. ¿De qué manera se almacena los datos de los conductores en la Institución?

8. ¿De qué manera se almacena los datos de las movilidades en la Institución?

FORMULARIO
DE LA ENTREVISTA DEL
SINDICATO SEÑOR DE SAN ROQUE

Nombre:.....

Grupo:.....

Fecha:.....

1. ¿Cómo se maneja la información en cuanto al registro de los conductores en el grupo?

2. ¿De qué manera hacen el registro de cada conductor y como se sabe que se encuentra en el grupo?

3. ¿Cómo se procede la búsqueda de los conductores?

4. ¿Cómo se puede saber que un conductor está en el grupo?

5. ¿De qué manera se almacena los datos de los conductores en el grupo?

6. ¿De qué manera se almacena los datos de las movilidades en el grupo?

CUESTIONARIO

SINDICATO SEÑOR DE SAN ROQUE

1. El proceso de registro del conductor:

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

2. El proceso de registro de la movilidad:

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

3. Las herramientas con la cual se hace el registro al conductor:

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

4. La búsqueda e información de los registros de conductores:

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

5. El control de conductores y movilidades es:

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

6. Las consultas de registrados de conductores es:

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

7. Sería bueno contar con un sistema de acuerdo al requerimiento a la institución sería de utilidad:

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

RECOLECCIÓN DE DATOS

Fotos del Sindicato Señor de san Roque



1. Conductores del Sindicato Señor de San Roque



1. Recoleccion de datos



RECOLECCIÓN DE DATOS

Información del Sindicato Señor de san Roque

En la Siguiete Tabla tenemos la Cantidad de Movilidades del Sindicato Señor de San Roque, tanto minibuses, taxis y trufis, que dan servicio a ala población.

Movilidades	Cantidad
Minibús	1546
Taxi	136
Trufis	118
Total	1800

Total de Socios Propietarios de la Institución del Sindicato Señor de San Roque. Como propietarios se tiene 1800 conductores y como asalariados 156 en un total de 1856 conductores.

Conductores	Cantidad
Propietarios	1800
Asalariados	156
Total	1856

Se puede Observar la Cantidad de Movilidades, Grupos y Líneas del Sindicato Señor de San Roque.

Tanto minibuses, taxis y trufis.

N	Grupos	Líneas	Cantidad	Tipo
1	A pioneros	698 negro	180	Minibús
2	B leones	737 negro	150	Minibús
3	C intocables	490 negro	80	Minibús
4	D escorpión	768 negro	120	Minibús
5	E correcaminos	737 rojo	112	Minibús
6	F andino	768 rojo	70	Minibús
7	G geniales	698 rojo	70	Minibús
8	H académicos	470 rojo	67	Minibús
9	I dinámicos	468 negro	120	Minibús
10	J tigres	470 rojo	113	Minibús
11	K lobos	402 negro	90	Minibús
12	L vencedores	468 rojo	124	Minibús
13	N diablos rojos	402 rojo	136	Minibús
14	P ejecutivos	490 rojo	114	Minibús
15	Carrys Illimani	100 negro	16	Trufis
16	Carrys escorpión	143 rojo	49	Trufis
17	Carrys Halcones	100 rojo	53	Trufis
18	Rey de reyes		28	Taxi
19	Fenix del norte		30	Taxi
20	Aguilas reales		28	Taxi
21	Aguilas por siempre		23	Taxi
22	Aguilas		27	Taxi
Total			1800	

ANEXO D

MANUAL DEL

USUARIO

MANUAL DE USUARIO

Sindicato

Señor de San Roque



**SISTEMA WEB PARA EL
SEGUIMIENTO Y CONTROL DE
SOCIOS PROPIETARIOS DE
MOVILIDADES**

7.Introducción

En el presente manual se explicara al usuario de cómo va poder utilizar el Sistema de información web para el seguimiento y control de socios propietarios de movilidades.

8.Objetivo

Acceder y guiar al usuario mediante este manual para el uso del sistema en forma correcta.

9.Requerimientos del Sistema

- Requerimiento de Hardware

Se Necesita un Ordenador como una computadora o laptop.

- Requerimiento de Software

Sistema Operativo Windows o Linux.

10.Tipos de usuario

Para el ingreso del sistema tenemos cinco tipos de usuario

- Usuario administrador
- Usuario Administrativo
- Usuario Jefe de Grupo
- Usuario Conductor
- Usuario Agente

11.Implementación del sistema

Para dicho ingreso al Sistema de información web para el seguimiento y control de socios propietarios de movilidades, se debe acceder al sistema a través de estos navegadores de internet.



Google Chrome



Mozilla Firefox

Ingresar a cualquiera de estos dos navegadores ala siguiente direccion:

<http://localhost/sissanroque/login>

se podra ver la pagina de inicio del sistema tal como se muestra en la siguiente figura.

Login del Sistema

Como podemos ver la pantalla de acceso del sistema, a la cual debera ingresar con un usuario y contraseña asignados por el administrador del Sindicato mixto señor de san roque.



Paso 1.- Login

2-Ingresa su documento de identidad como usuario.

3-Ingresa la contraseña con lo que se dio en la Institucion.

4-Click al Ingresar, y así Ingresara al sistema.

Inicio

En esta parte del Inicio del sistema muestra el logo del sindicato, la barra del menu del sistema: Inicio, Multas, aportes, ingresos, personas, asistencias vehiculos datos, mi perfil, la Barra horizontal serrar session, los montos aportes, ingresos, multas y cantidad de conductores.

The screenshot shows a web browser window with the URL localhost/sissanroque/inicio. The page features a sidebar menu on the left and a main content area. The sidebar menu includes: Inicio, Multas-Aportes/Ingresos, Personas, Asistencias, Vehiculos, Datos, Reportes, and Mi Perfil. The main content area displays four summary cards: Aportes (20.00), Ingresos (600.00), Multas (30.00), and Conductores (80). Each card includes a small icon and the text 'Mes de Noviembre'. The top right corner shows the user 'SIS SR' and a profile icon. Numbered callouts point to: 1. Logo of the Sindicato Señor de San Roque; 2. 'Personas' menu item; 3. User profile 'SIS SR'; 4. 'Aportes' card; 5. 'Ingresos' card; 6. 'Multas' card; 7. 'Conductores' card.

Paso 2.- Inicio del Sistema

8. Logo del Sindicato.

9. Barra de menu del Sistema.

10. Barra horizontal donde serrar session y cambiar contraseña.

11. Monto de aportes.

12. Monto de Ingresos.

13. Monto de Multas.

14. Cantidad de Conductores.

Usuario

Como podemos ver en pantalla el listado del administrador, administrativo y agente.

Mostrar 10 Buscar:

DOCUMENTO DE IDENTIDAD	NOMBRE Y APELLIDOS	CELULAR	NIVEL ACCESO	
00	Fernando		Agente	<input type="button" value="modificar"/>
4960129	German Hilari Nina	76787017	Administrador	<input type="button" value="modificar"/> <input type="button" value="eliminar"/>
4992526	Martires Apaza Ajno	7250521	Administrador	<input type="button" value="modificar"/> <input type="button" value="eliminar"/>
6053763	Edwin Cahuana Mamani	74066	Sistema	<input type="button" value="modificar"/> <input type="button" value="eliminar"/>
6862412	Eloy Huanca Chiara	73528404	Administrador	<input type="button" value="modificar"/> <input type="button" value="eliminar"/>

Mostrando 1 a 5 de 5 registros

Anterior 1 Siguiente

Paso 3.- Usuarios del sistema

1. Agregar nuevo usuario.
2. Listado en una tabla de los usuarios registrados que existe en el sistema.
3. Cuadro de texto buscar persona, a la persona que desee buscar mediante su documento de identidad o nombre de dicho usuario.
4. Botón de modificar información del usuario.
5. Botón de eliminar al usuario.

Agregar Usuario

Sindicato Señor de San Roque

localhost/sissanroque/usuarios

SIS SR

Menú Usuarios

Usuarios

Mostrar: 10

DOCUMENTO DE IDENTIDAD

00

4960129

4992526

6053763

6862412

Mostrando 1 a 5 de 5 registros

Para ingresar el usuario, debe usar su número de documento de identidad y su contraseña.

Contraseña

Control total

Agente de Parada

Habilitado

Cancelar Agregar

Anterior 1 Siguiente

1. Se muestra el formulario para el llenado de datos.

2. Despliegue de habilitar.

3. Botón agregar usuario.

4. Botón cancelar usuario.

Sindicato Señor de San Roque

localhost/sissanroque/usuarios

SIS SR

Menú Usuarios

Usuarios

Mostrar: 10

DOCUMENTO DE IDENTIDAD

00

4960129

4992526

6053763

6862412

Mostrando 1 a 5 de 5 registros

Para ingresar el usuario, debe usar su número de documento de identidad y su contraseña.

Contraseña

Control total

Agente de Parada

Habilitado

Cancelar Agregar

Anterior 1 Siguiente

1. Damos acceso como administrador y agente de parada.

2. Ponemos una contraseña para el usuario.

Conductores

Como podemos ver en pantalla el listado de conductores de forma detalladamente el ingreso del conductor, en que grupo se encuentra y su movilidad.

The screenshot shows a web application interface for 'Sindicato Señor de San Roque'. The main content area displays a table of drivers. The table has columns for Documento, Nombre y Apellidos, Contacto, Lic. de Conductor, Grupo, Ingreso, Sindicato, Vehículos, and acciones (Imprimir, modificar, eliminar). The interface includes a sidebar menu on the left and a top navigation bar. Six orange callout boxes with numbers 1 through 6 point to specific elements: 1 points to the 'Agregar' button, 2 points to the 'Lic. de Conductor' column, 3 points to the search bar, 4 points to the 'Imprimir' button, 5 points to the 'eliminar' button, and 6 points to the 'modificar' button.

DOCUMENTO	NOMBRE Y APELLIDOS	CONTACTO	LIC. DE CONDUCTOR	GRUPO	INGRESO	SINDICATO	VEHICULOS	IMPRIMIR	
422400	Andrea Villca Hillaquita		Profesional B	K LOBOS	08/11/2022	SEÑOR DE SAN ROQUE	1213-GB1 (Blanco) Toyota	Imprimir	modificar
2011546	Julian Mamaní Capta		Profesional B	K LOBOS	12/04/2019	SEÑOR DE SAN ROQUE	8002-UPP (Blanco) KinLong	Imprimir	modificar eliminar
2268954	Jaime Balboa Alanoca	78248956	Profesional B	F ANDINO	26/07/2019	SEÑOR DE SAN ROQUE	8023-LOR (Gris) Nissan	Imprimir	modificar
2598111	Celestino Mamaní Vargas		Profesional C	G GR	019	SEÑOR DE SAN ROQUE	1821-EXF (Blanco) Nissan	Imprimir	modificar eliminar
2615181	Pavolino Mamaní Quispe		Profesional B	B LEONES	21/01/2019	SEÑOR DE SAN ROQUE	2347-REX (Plateado) Nissan	Imprimir	modificar eliminar
2668991	Tomas Rojas Machaca		Profesional B	D ESCORPION	24/07/2019	SEÑOR DE SAN ROQUE	4622-EVN (Blanco) Foton	Imprimir	modificar eliminar
2674054	Eloy Flores Cadena		Profesional B	H ACADEMICOS	08/04/2019	SEÑOR DE SAN ROQUE	1718-HTE (Blanco) Nissan	Imprimir	modificar eliminar
3320926	Adrian Cauna Mamaní		Profesional B	I DINAMICOS	16/09/2018	SEÑOR DE SAN ROQUE	2444-XDK (Blanco) Mazda	Imprimir	modificar eliminar
3325923	Maximo Leiva Callata		Profesional A	L VENCEDORES	03/01/2018	SEÑOR DE SAN ROQUE	1003-VTK (Plomo) Nissan	Imprimir	modificar eliminar
3325927	Maximo Leiva Callata		Profesional B	T TIGRES	03/01/2019	SEÑOR DE SAN ROQUE	1003-VTK (Plomo) Nissan	Imprimir	modificar eliminar

Paso 4.- Conductores

1. Agregar nuevo Conductor.
2. Listado en una tabla de los usuarios registrados que existe en el sistema.
3. Cuadro de texto buscar persona, a la persona que desee buscar mediante su documento de identidad o nombre de dicho conductor.
4. Botón de modificar información del conductor.
5. Botón de eliminar al conductor.
6. Imprimir memorándum, reporte.

Agregar Conductor

The screenshot shows the 'Agregar Conductor' form with the following fields and options:

- Documentación de identidad: 2345
- Nombres: Franz Guillermo
- Apellidos: Paterno Quispe
- Apellido Materno: Apaza
- Fecha nacimiento: 09/11/2022
- Departamento: La paz
- Dirección: Iomas
- Teléfono: 4654654
- Documentos: Documento Frente, Documento Dorsal
- Habilitado:
- Botones: Cancelar, Agregar

1. Se muestra el formulario para el llenado de datos.

2. Despliegue de habilitar.

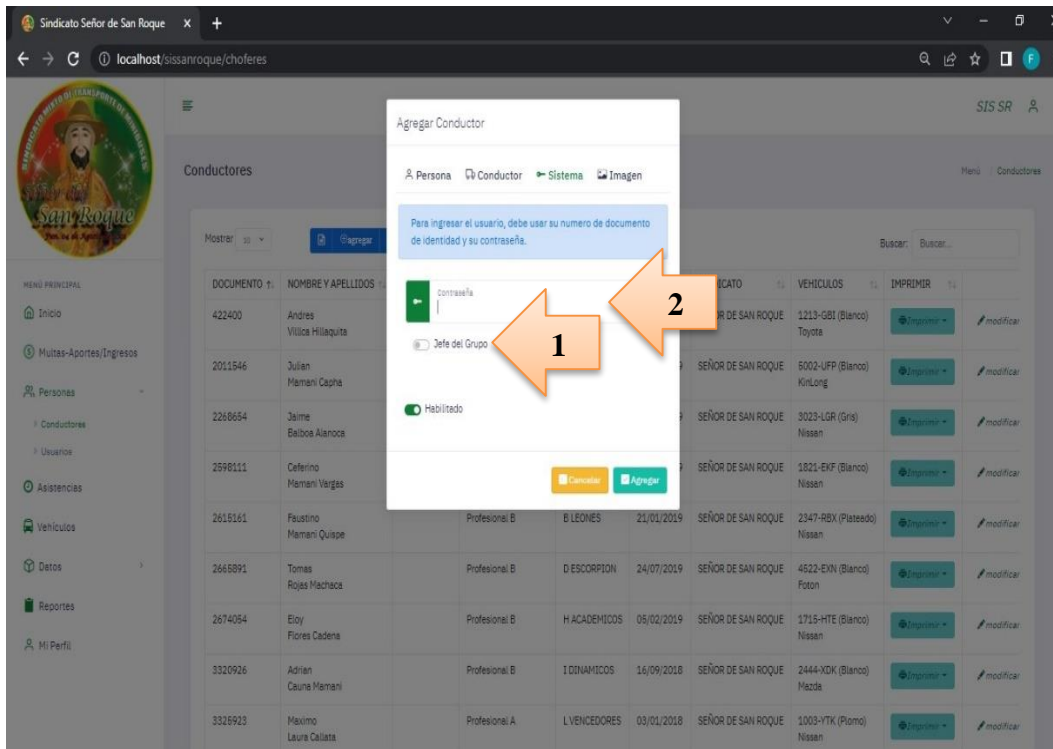
3. Botón agregar conductor.

4. Botón cancelar conductor.

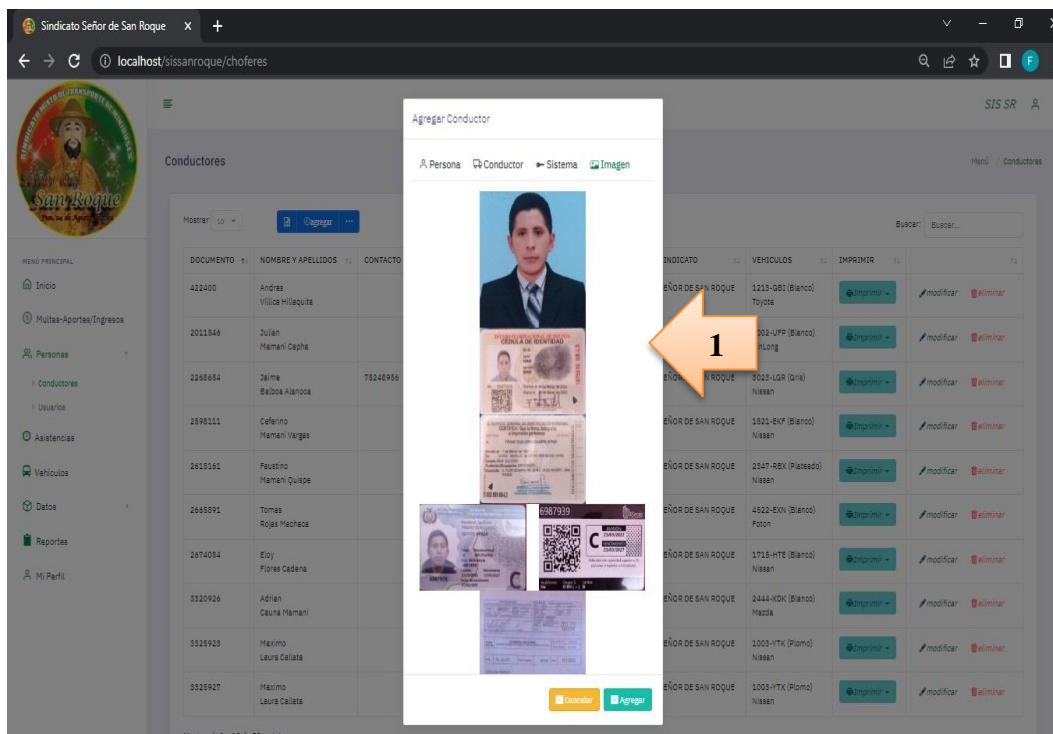
The screenshot shows the 'Agregar Conductor' form with the following fields and options:

- Sindicato: SEÑOR DE SAN ROQUE
- Socio: A PIONEROS
- Fecha ingreso: 29/11/2022
- Tipo de conductor: Propietario
- Categoría: Profesional A
- Licencia lugar de expedición: la paz
- Placas de Vehículo: 2019-94E
- Documentos: Licencia Frente, Licencia Dorsal
- Foto RIAT
- Habilitado:
- Botones: Cancelar, Agregar

1. Datos como conductor.



1. Damos acceso si es jefe de grupo.
2. Ponemos una contraseña para el conductor.



1. Vemos las imágenes subidas del conductor.

Memorandum de trabajo

Sindicato Señor de San Roque x Imprimir Documento x Imprimir Documento x +

localhost/sissanroque/ws/wsChoferes.php?memorandum&imprimir=65

MEMORANDUM DE TRABAJO

 <p>SINDICATO MIXTO DE TRANSPORTES "SEÑOR DE SAN ROQUE" FUND. 04 DE AGOSTO DE 2001 RESOLUCIÓN SUPREMA Nº 221159 EL ALTO - LA PAZ EL ALTO, 29 de Noviembre del 2022</p>	SEÑOR(A): JULIAN MAMANI CAPHA SOCIO PROPIETARIO PRESENTE REF.: ACTUALIZACIÓN DE MEMORANDUM
---	--

Estimado Compañero:
EL DIRECTORIO DEL SINDICATO MIXTO DE TRANSPORTE "SEÑOR DE SAN ROQUE", en aplicación del reglamento interno le comunicamos que: su **MEMORANDUM DE TRABAJO ESTA ACTUALIZADO, VIGENTE COMO SOCIO PROPIETARIO E INGRESADO EN FECHA 12/04/2019** al Sr.: JULIAN MAMANI CAPHA C.I. 2011546 L.P. con las siguientes moviidades que pertenecen al grupo K LOBOS:

PLACA : 5002-UPF
COLOR : Blanco
MARCA : KimLong
MODELO : 2019
CLASE : MINIBUS

Debiendo, cumplir y respetar el estatuto orgánico y reglamento interno de nuestra institución.
Atentamente,

EL DIRECTORIO!



1. Memorandum de trabajo del conductor.

Sindicato Señor de San Roque x Imprimir Documento x Imprimir Documento x +

localhost/finalsistemasanroque/ws/wsChoferes.php?reporte&imprimir=13

REPORTE DEL CONDUCTOR

 <p>SINDICATO MIXTO DE TRANSPORTES "SEÑOR DE SAN ROQUE" FUND. 04 DE AGOSTO DE 2001 RESOLUCIÓN SUPREMA Nº 221159 EL ALTO - LA PAZ EL ALTO, 19 de Setiembre del 2022</p>	SEÑOR(A): JOSE MENDEZ QUISSPE SOCIO PROPIETARIO PRESENTE REF.: REPORTE CONDUCTOR
---	--

Datos como Conductor del Sindicato:

Identidad	Foto	Nombres	Apellidos	Contacto	Grupo	Ingreso	Sindicato	Vehiculo
789		JOSE	MENDEZ QUISSPE	465465 LOMAS	F Andino	07/08/2022	SEÑOR DE SAN ROQUE	PLACA : 25911HB COLOR : Blanco MARCA : Nissan MODELO: 1994 CLASE : MINIBUS

Debiendo, cumplir y respetar el estatuto orgánico y reglamento interno de nuestra institución.
Atentamente,

EL DIRECTORIO!



1. Reporte del Conductor.

Ingresos, Multas y Aportes.

The screenshot shows the 'Multas-Aportes/Ingresos' interface. At the top left, there is a blue '+ Agregar' button (callout 1). Below it is a table with columns: DOCUMENTO DE IDENTIDAD, NOMBRE Y APELLIDOS, MONTO, CONCEPTO, OBSERVACION, and FECHA. The first two rows are highlighted in green, and the last row is highlighted in red. A search bar is located at the top right (callout 3). A second table is visible below the first one.

DOCUMENTO DE IDENTIDAD	NOMBRE Y APELLIDOS	MONTO	CONCEPTO	OBSERVACION	FECHA
6987939	Franz Guillermo Quiso Lopez	+ 50.00 + 500.00 + 30.00 Total: 380.00	Multa Ingreso Aporte	multa Ingreso kind Asistencia	08 Noviembre 2022 08 Noviembre 2022 08 Noviembre 2022
9252117	Alberto Quiso Lopez	+ 20.00 + 200.00 Total: 220.00	Multa Ingreso	multa Ingreso	29 Noviembre 2022 11 Noviembre 2022
422400	Andres Villica Hillaquita	-6.00			
2011546	Julian Mamani Capa	-6.00			

1. Agregar nueva multa, aporte e ingresos.

2. Listado en una tabla de los usuarios o conductores de registro de ingresos, y aportes.

3. Cuadro de texto buscar persona, a la persona que desee buscar mediante su documento de identidad o nombre de dicho conductor.

The screenshot shows the 'Financiero' modal form. It has a search field for 'Personas' (callout 1) with the value '(12345) FERNANDO MAMANI MAMANI'. Below the search field are dropdown menus for 'Concepto' (Ingresos) and 'Monto' (30). There is also a text field for 'Observación' (Asistencia). At the bottom, there are 'Cancelar' and 'Agregar' buttons (callout 3). A callout 2 points to the 'Agregar' button.

1. Buscar conductor

2. Boton cancelar.

3. Boton Agregar.

Asistencia de Conductores

CONDUCTOR	PLACA	NEA	DETALLES	TIPO	FECHA
Franz Guillermo Quispe Apeza	668-SDY	768 ROJO	MINIBUS Nissan Beige	E	06/10/2022 10:56:50

1. Agregar asistencia.
2. Listado en una tabla de los conductores asistidos.
3. Cuadro de texto buscar persona, a la persona que desee buscar mediante su documento de identidad o nombre de dicho conductor.
4. Botón de eliminar.

1048-GSI (Angel Vilca Gironda)

04:11:29 PM

1. Buscar con numero de placa
2. Boton cancelar.
3. Boton Agregar.

Vehiculos

The screenshot shows a web application interface for managing vehicles. On the left is a sidebar menu with options like 'Inicio', 'Multas-Aportes/Ingresos', 'Personas', 'Asistencias', 'Vehiculos', 'Datos', 'Reportes', and 'Mi Perfil'. The main area displays a table of vehicles with columns for 'PLACA', 'LINEA', 'DETALLES', and actions. A search bar is located at the top right. Five orange arrows with numbers 1 through 5 point to specific elements: 1 points to the 'Agregar' button, 2 points to a row in the table, 3 points to the search bar, 4 points to a 'Modificar' button, and 5 points to an 'eliminar' button.

PLACA	LINEA	DETALLES	Acciones
1003-YTK	698 ROJO	MINIBUS(Nissan) Plomo	Modificar
1003-YTX	439 NEGRO	MINIBUS(Nissan) Plomo	Modificar eliminar
1048-GSI	698 ROJO	MINIBUS(Toyota) Blanco	Modificar eliminar
1090-KHX	490 ROJO	MINIBUS(Nissan) Blanco	Modificar eliminar
115-RBF	698 NEGRO	TRUFIS(Jinchen) Blanco	Modificar eliminar
1160-ZCH	698 NEGRO	TRUFIS(Jimbei) Beigle	Modificar eliminar
1206-YZL	402 ROJO	MINIBUS(Toyota) Blanco	Modificar eliminar

1. Agregar Vehiculos.
2. Listado en una tabla de vehiculos.
3. Cuadro de texto buscar vehiculo, mediante placa.
4. Boton de modificar vehiculo.
5. Botón de eliminar.

The screenshot shows the 'Agregar Vehiculo' modal form overlaid on the vehicle list. The form contains fields for 'Placa' (with value '12'), 'Modelo', 'Linea', 'Clase', 'Marca', and 'Color', each with a dropdown arrow. There is also an 'Imagen' field with a camera icon and a 'Habilitado' checkbox. At the bottom of the form are two buttons: 'Cancelar' and 'Agregar'. Two orange arrows with numbers 1 and 2 point to these buttons.

1. Boton cancelar.
2. Boton Agregar.

Grupos

The screenshot shows a web application interface for managing groups. On the left is a sidebar menu with options like 'Inicio', 'Multas-Aportes/Ingresos', 'Personas', 'Asistencias', 'Vehículos', 'Datos', 'Reportes', and 'Mi Perfil'. The main area displays a table of groups with columns for '#', 'NOMBRE', and actions. A search bar is located at the top right. Five orange arrows point to specific elements: 1 points to the 'Agregar' button, 2 points to the table, 3 points to the search bar, 4 points to the 'modificar' button, and 5 points to the 'eliminar' button.

#	NOMBRE	modificar	eliminar
1	A PIONEROS	modificar	eliminar
2	B LEONES	modificar	eliminar
3	C INTOCABLES	modificar	eliminar
4	D ESCORPION	modificar	eliminar
5	E CORRECAMINOS	modificar	eliminar
6	F ANDINO	modificar	eliminar
7	G GENIALES	modificar	eliminar
8	H ACADEMICOS	modificar	eliminar

1. Agregar Grupos.
2. Listado en una tabla de Grupos.
3. Cuadro de texto buscar grupo, mediante nombre del grupo.
4. Boton de modificar grupo.
5. Botón de eliminar grupo.

The screenshot shows the 'Agregar Grupo' modal form overlaid on the main interface. The form has a text input field for 'Nombre' and a 'Habilitado' checkbox. At the bottom of the modal are two buttons: 'Cancelar' and 'Agregar'. Two orange arrows point to these buttons: 1 points to 'Cancelar' and 2 points to 'Agregar'.

1. Boton cancelar.
2. Boton Agregar.

Lineas

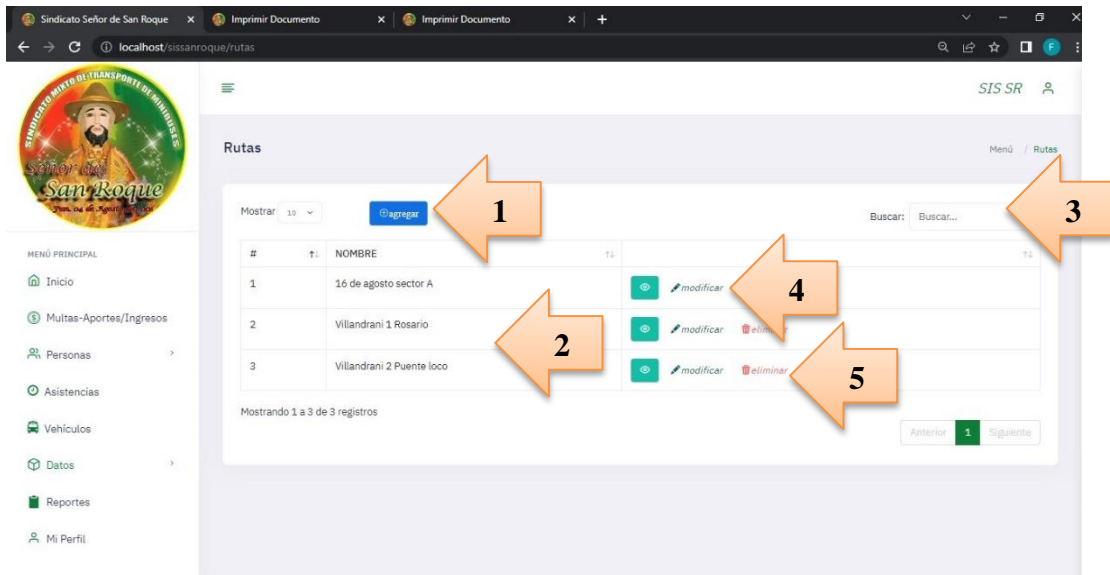
The screenshot shows a web application interface for managing bus lines. On the left is a sidebar menu with options like 'Inicio', 'Multas-Aportes/Ingresos', 'Personas', 'Asistencias', 'Vehiculos', 'Datos', 'Reportes', and 'Mi Perfil'. The main area is titled 'Lineas' and contains a table of bus lines. At the top of the table area, there is a 'Mostrar' dropdown set to '10', an 'Agregar' button (callout 1), and a search box labeled 'Buscar: Buscar...' (callout 3). The table has columns for '#', 'NOMBRE', and 'RUTA'. Each row includes 'modificar' and 'eliminar' buttons (callout 4). The table shows several entries, including 'ROJO' and 'NEGRO' lines with various routes like 'Villandran 1 Rosario' and '16 de agosto sector A'. At the bottom of the table area, there are 'Anterior', '1', '2', and 'Siguiete' buttons (callout 5).

1. Agregar Lienas.
2. Listado en una tabla de Lineas.
3. Cuadro de texto buscar linea, mediante nombre de la linea.
4. Boton de editar linea.
5. Botón de eliminar línea.

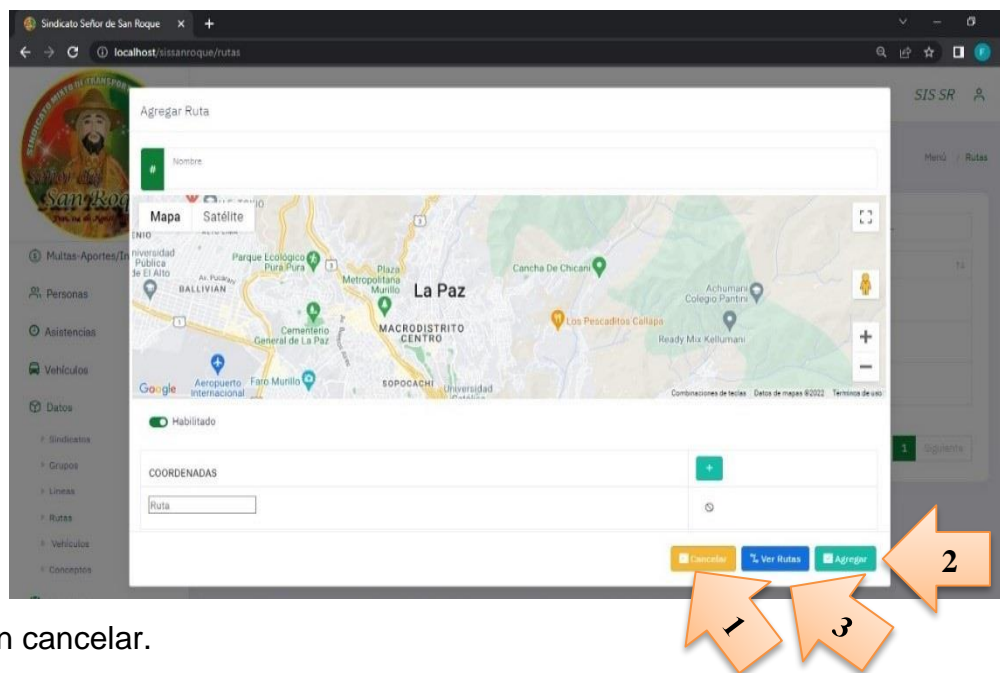
The screenshot shows the 'Agregar Linea' modal form overlaid on the 'Lineas' table. The form has a title 'Agregar Linea' and a subtitle 'Agregar ruta a la linea.'. It contains four input fields: '# Número de linea', 'N Nombre de la linea', 'Rutas' (with a plus icon), and 'Habilitado' (with a checked checkbox). At the bottom of the form are two buttons: 'Cancelar' and 'Agregar' (callout 2). The background shows the 'Lineas' table with entries like '402 ROJO', '402 NEGRO', '439 NEGRO', etc.

1. Boton cancelar.
2. Boton Agregar.

Rutas

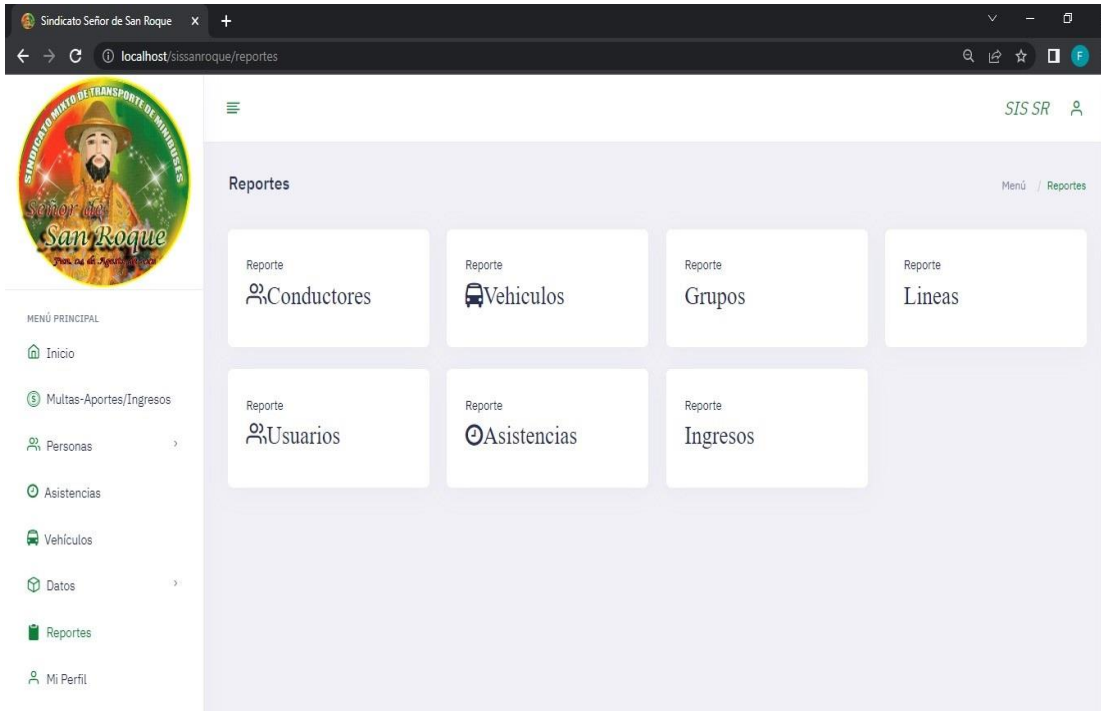


1. Agregar Ruta.
2. Listado en una tabla de Rutas.
3. Cuadro de texto buscar ruta, mediante nombre de la ruta.
4. Boton de editar ruta.
5. Botón de eliminar ruta.



1. Boton cancelar.
2. Boton Agregar.
3. Ver ruta

Imprimir Reportes



1. Imprimir Reportes Conductor, Vehiculos, Grupos, Lineas, Usuarios, Asistencias, Ingresos

MANUAL TÉCNICO

Sindicato

Señor de San Roque



**SISTEMA WEB PARA EL
SEGUIMIENTO Y CONTROL DE
SOCIOS PROPIETARIOS DE
MOVILIDADES**

1. Introducción

En el presente manual técnico se explicara los pasos necesarios para cualquier usuario de cómo va poder realizar la instalación del aplicativo y hacer el uso del Sistema de información web para el seguimiento y control de socios propietarios de movilidades.

2. Objetivo

Guiar al Usuario con este manual técnico para el uso del sistema de forma correcta.

3. Requerimientos del Sistema

- Requerimiento de Hardware

Se Necesita un Ordenador como una computadora o laptop.

Memoria RAM: mínimo 4 Gigabytes

Espacio de disco duro mínimo: 5Gb

- Requerimiento de Software

Sistema Operativo Windows o Linux.

4. Herramientas Utilizadas para el desarrollo

Gestor de Base de datos MySQL

Servidor Web Apache

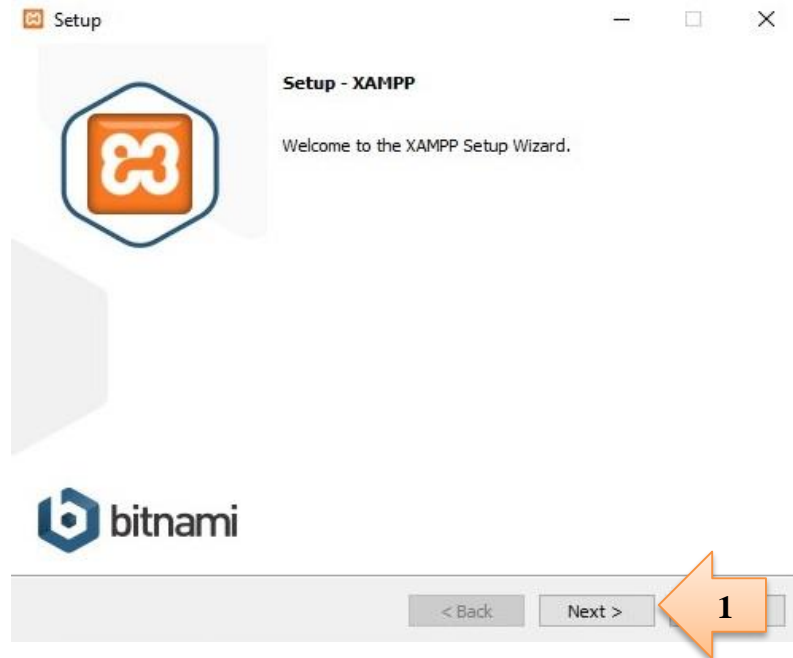
Lenguaje de Programación PHP

Las versiones de PHP recomendable para instalar a un servidor.

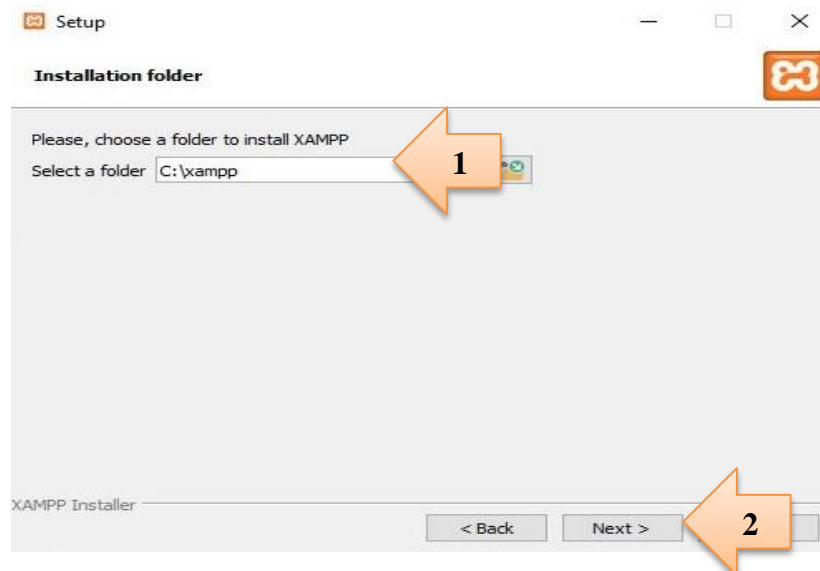
- **PHP 7.1**
- **PHP 7.2**
- **PHP 7.3**
- **PHP 7.4**
- **PHP 8.0**
- **PHP 8.1**

5. Instalación de Web Server

Entrar a cualquier navegador desde una computadora y escribir el siguiente enlace para poder descargar xampp Win32: <https://xampp.uptodown.com/windows/versions> una vez descargado se guardara en descargas y ejecutarlo el archivo xampp7.4



1. Presionamos Next.



1. Re direccionamos a la unidad C:xampp.

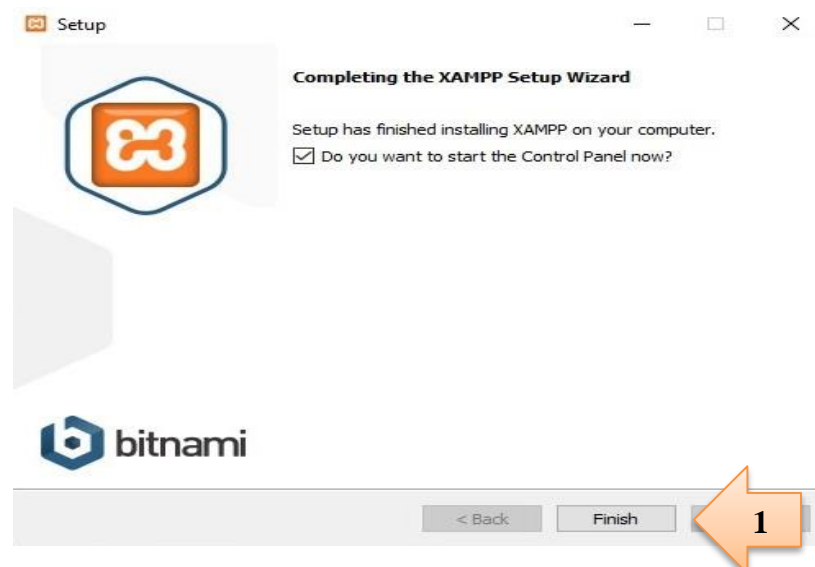
2. después presionamos next.



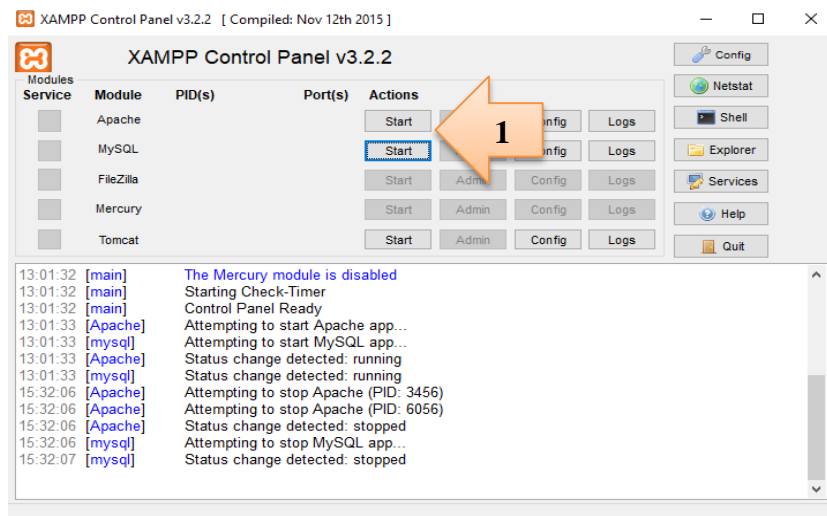
1. Seleccionamos permitir acceso



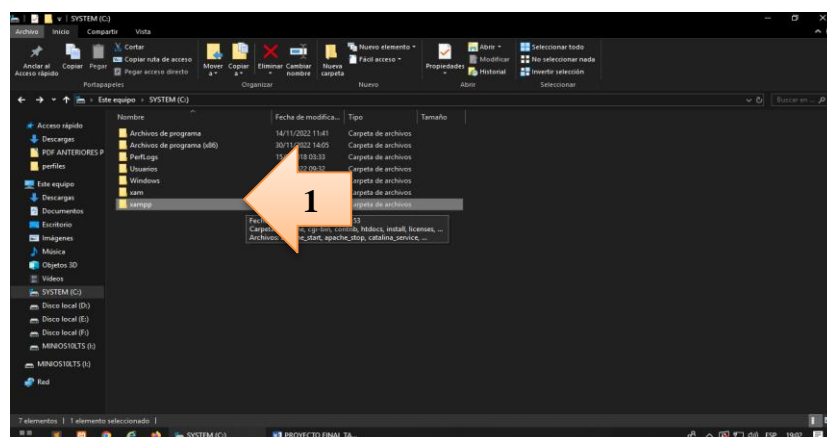
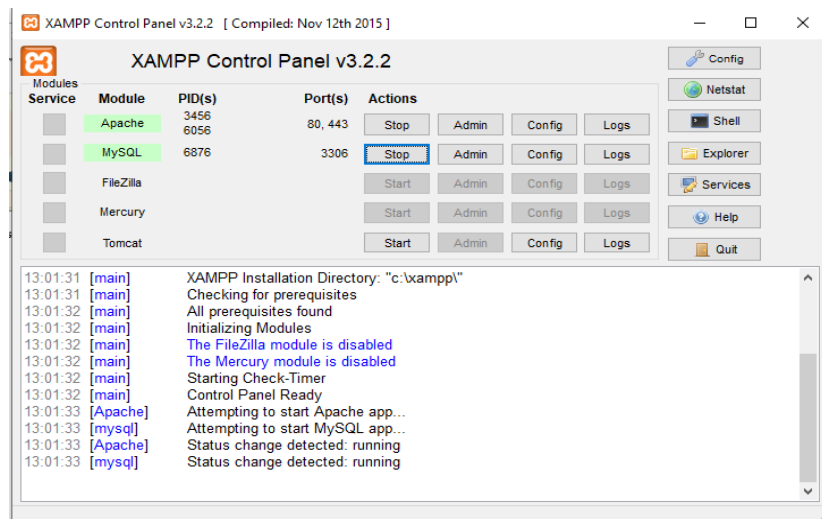
Dejamos instalar xampp



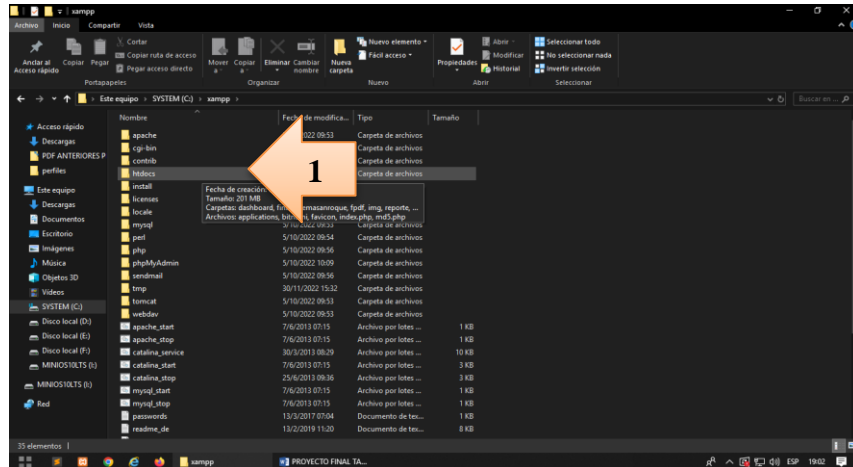
1. Por ultimo seleccionamos finish



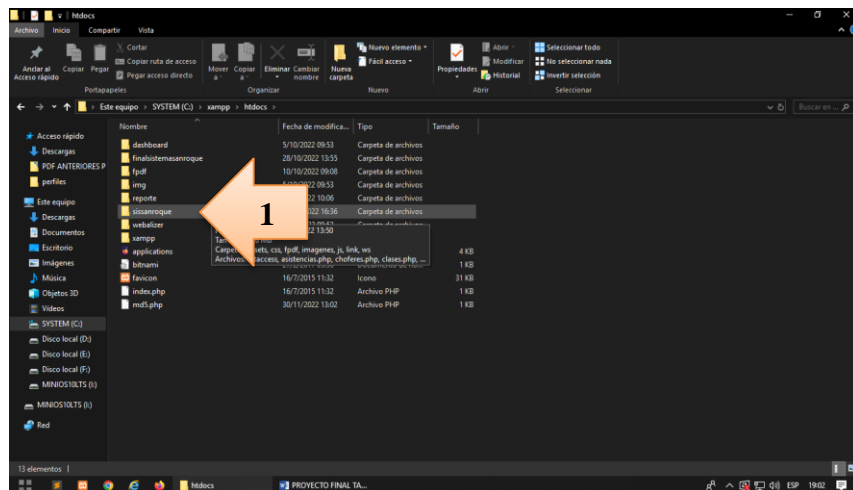
1. Damos en Start Apache, Start MySQL



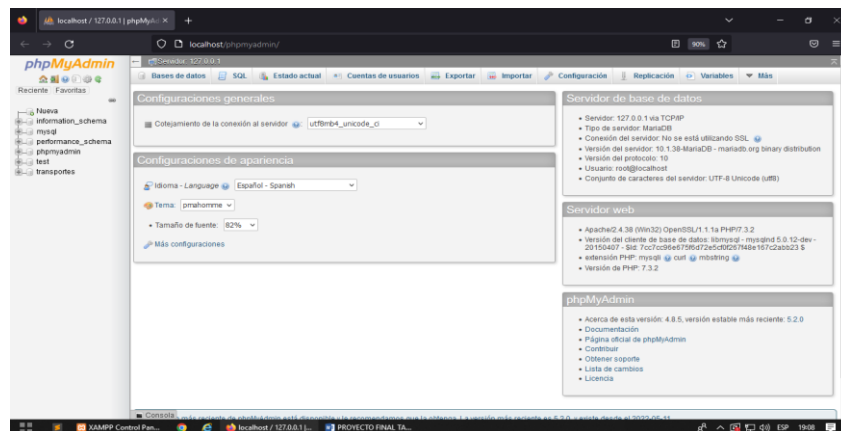
1. Entramos a xampp



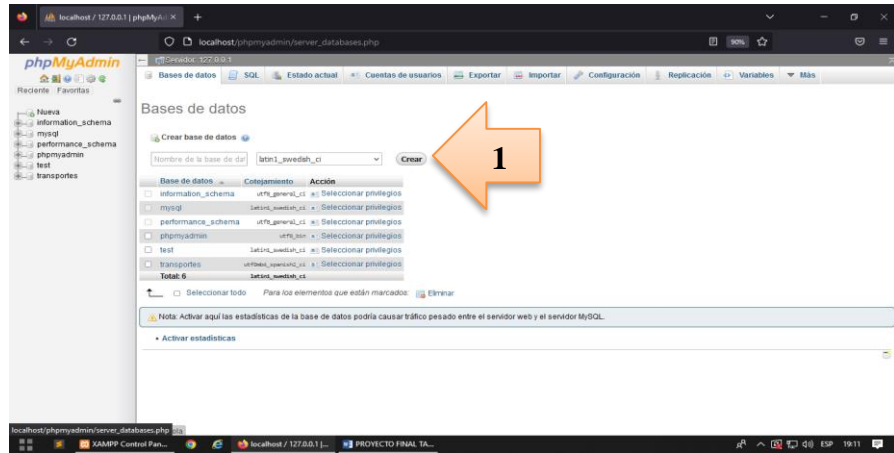
1. Entramos a htdocs



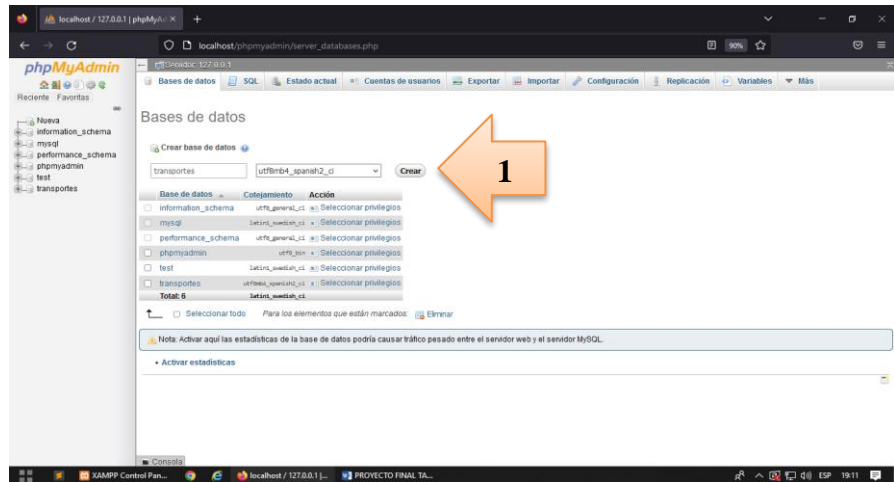
1. Adentro de htdocs lo pegamos la carpeta sissanroque lo que se proporcione, como vemos en la siguiente imagen



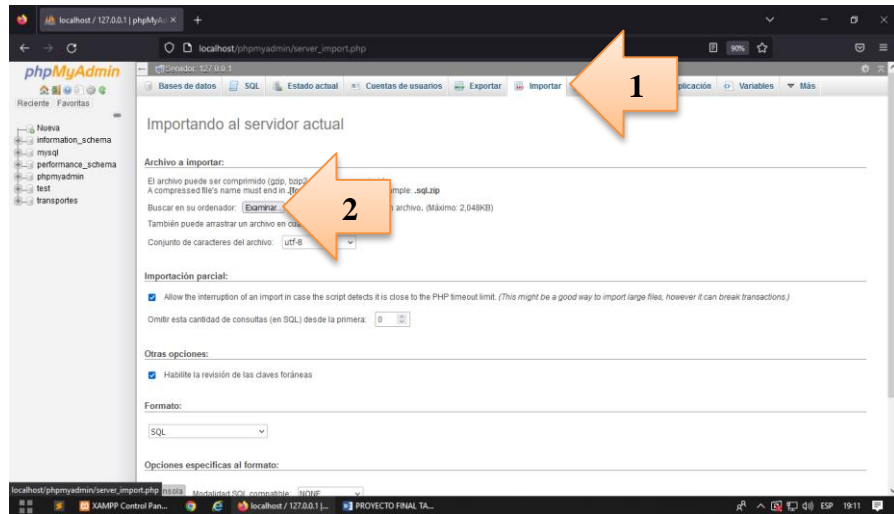
1. Entramos a un navegador y copiar la url <http://localhost/phpmyadmin/> y tiene que mostrarnos la siguiente pantalla.



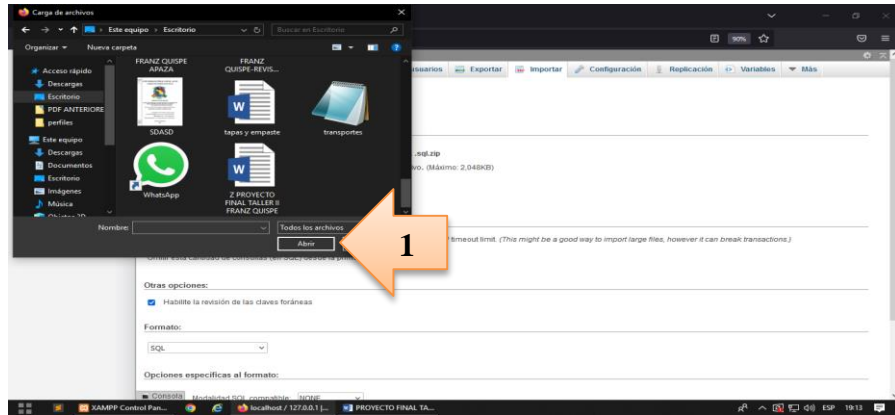
1. Creamos la base de datos con el nombre de transportes utf8 mb4



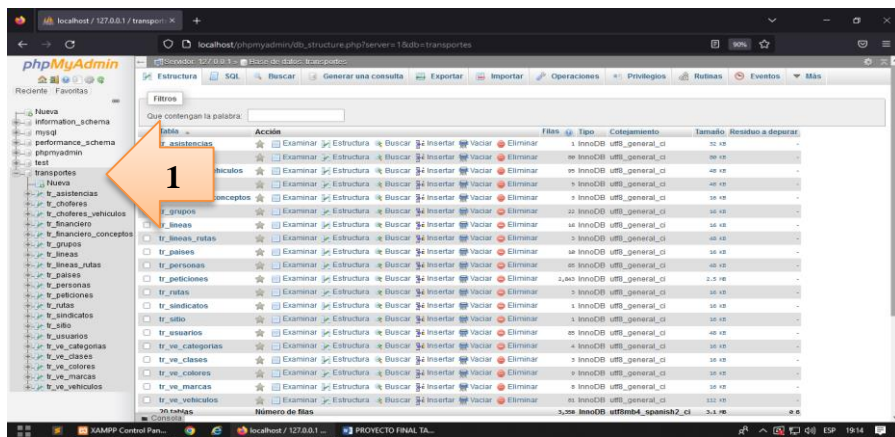
1. Seleccionamos crear



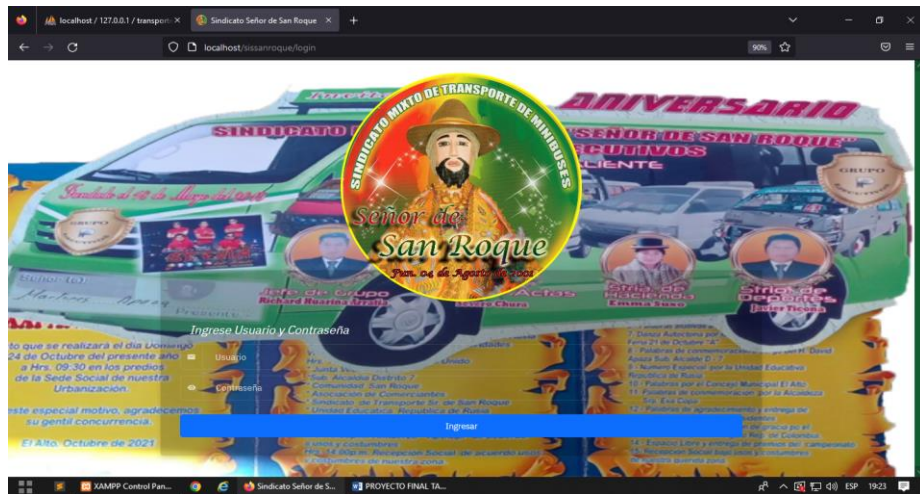
1. Seleccionamos importar
2. Después apretamos en examinar



1. Seleccionamos el archivo transportes y aemos clic en abrir



1. Y ya tenemos la base de datos creada de trasportes



Por ultimo entramos a un navegador con la url: <http://localhost/sissanroque/login>

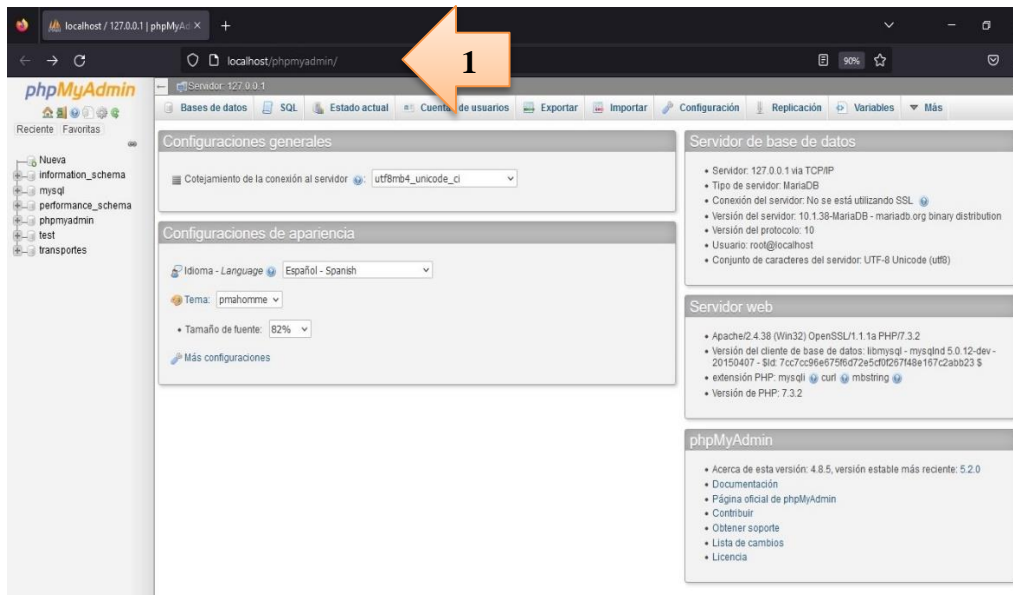
Y nos mostrar la siguiente ventana lo cual debemos ingresar:

Usuario: Cedula de identidad → 85974558

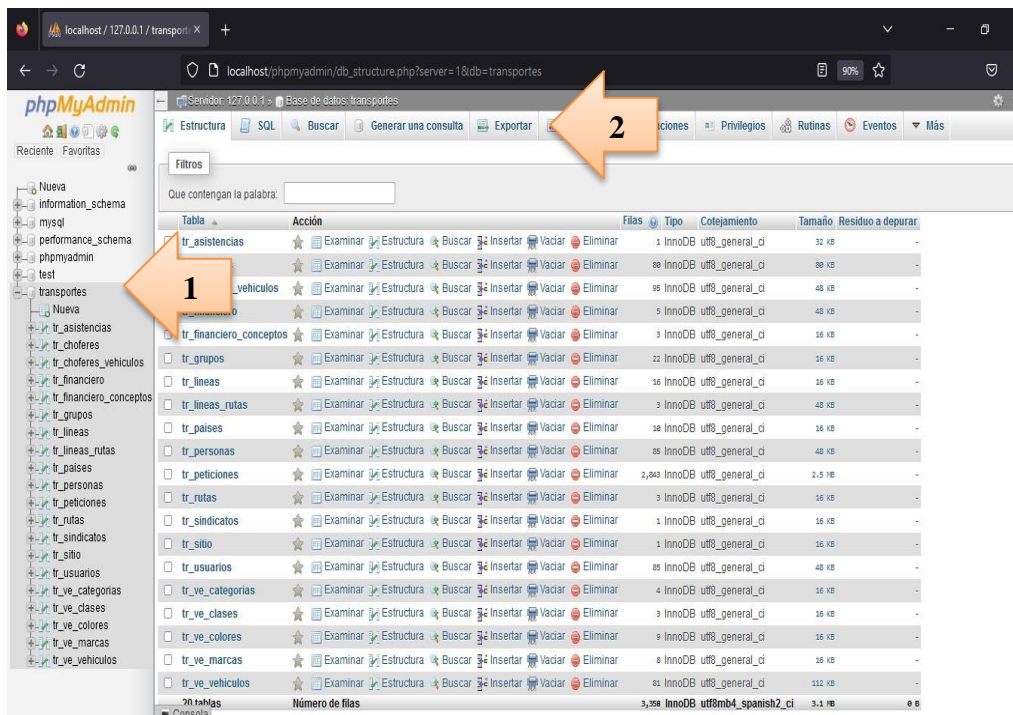
Password: Primer nombre y Placa de Movilidad → Franz-2541-WSD

6. Backup

Demostramos como a ser un respaldo de todo el archivo de los registrados conductores.

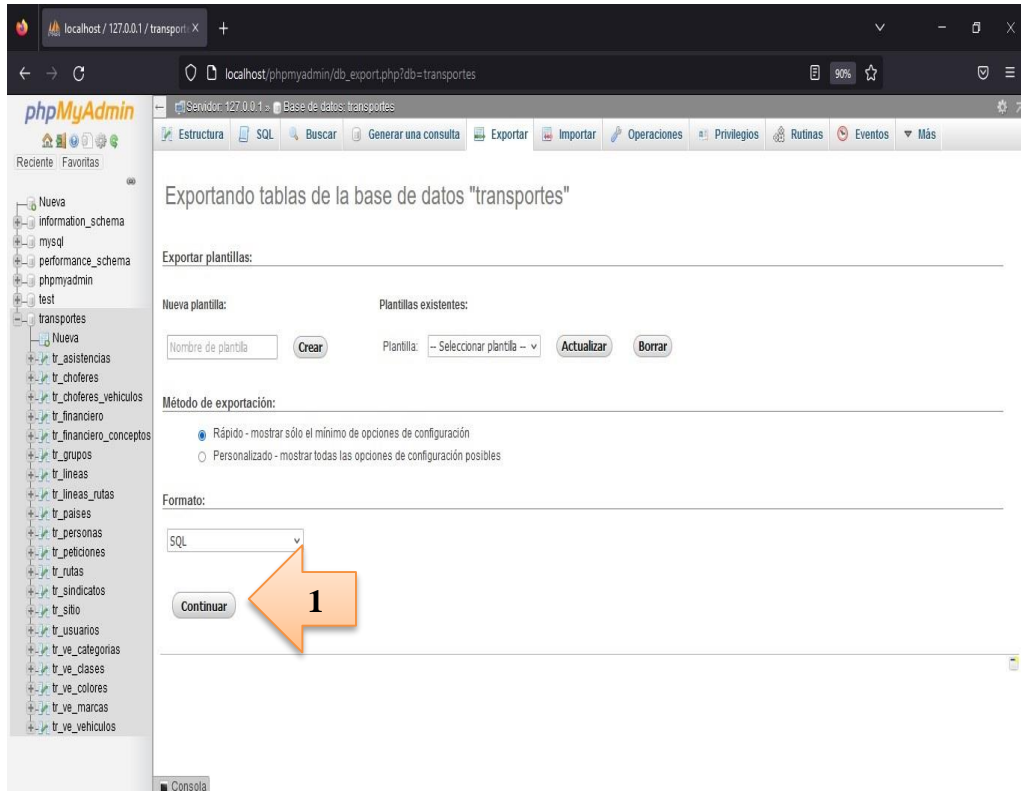


1. Entrar al Navegador y escribir <http://localhost/phpmyadmin/>

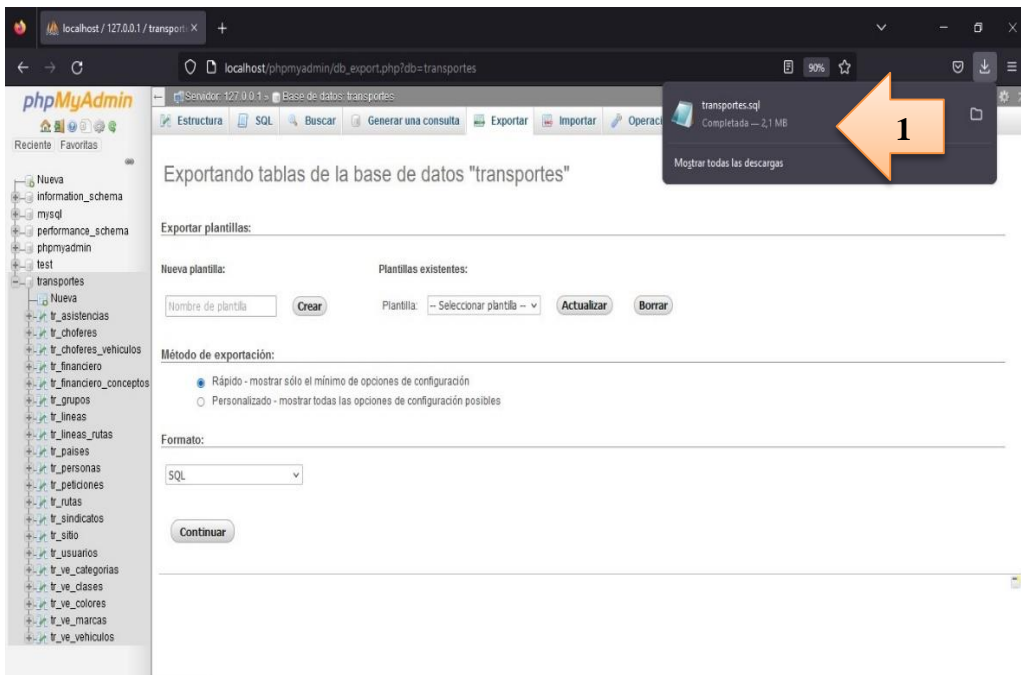


1. Luego hacer clic en la base de datos transportes

2. Luego vamos a la opción exportar



1. Le damos continuar y se descargara el archivo transportes.sql



1. El archivo descargado transportes.sql
2. Buscar en la carpeta descargas con el nombre transportes.sql

ANEXO E

AVALES

AVAL DE CONFORMIDAD

(TUTOR METODOLÓGICO)

El Alto, 17 de noviembre de 2022

Señor:
M. Sc. David Carlos Mamani Quispe
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 WEB PARA EL SEGUIMIENTO Y CONTROL DE SOCIOS PROPIETARIOS DE MOVILIDADES
CASO: SINDICATO "SEÑOR DE SAN ROQUE"
MODALIDAD: PROYECTO DE GRADO
Univ. Franz Guillermo Quispe Apaza
Registro Universitario: 11005100
Cedula de Identidad: 6987939

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. Mañsol Arguedas Balladares

TUTOR METODOLÓGICO
TALLER DE GRADO II

AVAL DE CONFORMIDAD

(TUTOR ESPECIALISTA)

El Alto, 15 de noviembre de 2022

Señor:
Ing. Marisol Arguedas Valladares

TUTOR METODOLÓGICO
TALLER DE GRADO II
Presente. –

REF. AVAL DE CONFORMIDAD

Distinguido director de carrera:

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

TITULO: SISTEMA WEB PARA EL SEGUIMIENTO Y CONTROL DE SOCIOS
PROPIETARIOS DE MOVILIDADES
CASO: SINDICATO "SEÑOR DE SAN ROQUE"
MODALIDAD: PROYECTO DE GRADO
Univ. Franz Guillermo Quispe Apaza
Registro Universitario: 11005100
Cedula de Identidad: 6987939

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. Margarita Bernarda López Mariaca
TUTOR ESPECIALISTA

AVAL DE CONFORMIDAD

(TUTOR REVISOR)

El Alto, 15 de noviembre de 2022

Señor:
Ing. Marisol Arguedas Valladares

TUTOR METODOLÓGICO
TALLER DE GRADO II

Presente. –

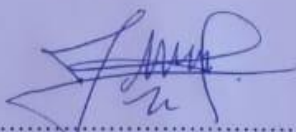
REF. AVAL DE CONFORMIDAD

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

TITULO: SISTEMA WEB PARA EL SEGUIMIENTO Y CONTROL DE SOCIOS PROPIETARIOS DE MOVILIDADES
CASO: SINDICATO "SEÑOR DE SAN ROQUE"
MODALIDAD: PROYECTO DE GRADO
Univ. Franz Guillermo Quispe Apaza
Registro Universitario: 11005100
Cedula de Identidad: 6987939

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. Zara Yujra Cama
TUTOR REVISOR

AVAL DE CONFORMIDAD

SINDICATO "SEÑOR DE SAN ROQUE"

El Alto, 16 de noviembre de 2022

Señor:

M. Sc. David Carlos Mamani Quispe

DIRECTOR DE CARRERA

CARRERA INGENIERÍA DE SISTEMAS

Presente. –

REF. AVAL DE CONFORMIDAD

Distinguido Ingeniero por intermedio de la presente hago llegar un saludo cordial y deseándole éxitos en sus labores cotidiana.

El Sindicato Señor de San Roque, a través del Secretario General de la institución en uso de las atribuciones, felicita la implementación del sistema, al Univ. Franz Guillermo Quispe Apaza con cedula de identidad 6987939lp, tengo a bien para comunicarle mi conformidad del proyecto de grado "**SISTEMA WEB PARA EL SEGUIMIENTO Y CONTROL DE SOCIOS PROPIETARIOS DE MOVILIDADES CASO: SINDICATO "SEÑOR DE SAN ROQUE"**", 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.


David Carlos Mamani Quispe
SECRETARIO GENERAL
SINDICATO MIXTO DE TRANSPORTES
"SEÑOR DE SAN ROQUE"

