



**UNIVERSIDAD
NACIONAL
DE LOJA**



Área de la Energía, las Industrias y los Recursos Naturales No Renovables

CARRERA DE INGENIERÍA EN SISTEMAS

**“TECNIFICACIÓN DEL SISTEMA MUNICIPAL DE
ESTACIONAMIENTO ROTATIVO TARIFADO
(SIMERT) DE LA CIUDAD DE LOJA, MÓDULO
CONECTIVIDAD DE LA RED DE DATOS DEL
SIMERT UTILIZANDO LA TECNOLOGÍA GPRS”**

TESIS DE GRADO PREVIA A LA
OBTENCIÓN DEL TÍTULO DE
INGENIEROS EN SISTEMAS.

AUTORES:

*César Danilo Calle Loja
Julia Vanessa Morocho Guadalima*

DIRECTOR:

Ing. Hernán Leonardo Torres Carrión, Mg. Sc.

**Loja-Ecuador
2014**

CERTIFICACIÓN DEL DIRECTOR

Ingeniero

Hernán Leonardo Torres Carrión, Mg. Sc.

**COORDINADOR DE LA CARRERA DE INGENIERÍA EN SISTEMAS DE LA
UNIVERSIDAD NACIONAL DE LOJA**

CERTIFICA:

Que el presente trabajo de titulación elaborado previo a la obtención del Título de Ingenieros en Sistemas, titulado **“TECNIFICACIÓN DEL SISTEMA MUNICIPAL DE ESTACIONAMIENTO ROTATIVO TARIFADO (SIMERT) DE LA CIUDAD DE LOJA, MÓDULO CONECTIVIDAD DE LA RED DE DATOS DEL SIMERT UTILIZANDO LA TECNOLOGÍA GPRS”**, realizado por los egresados **César Danilo Calle Loja** y **Julia Vanessa Morocho Guadalima**, cumple con los requisitos establecidos por las normas generales para la graduación en la Universidad Nacional de Loja, tanto en aspectos de forma como de contenido.

Por lo tanto autorizo proseguir los trámites legales para su presentación y defensa.

Loja, 01 de Julio del 2014.



Ing. Hernán Leonardo Torres Carrión, Mg. Sc.
DIRECTOR DE TESIS



AUTORÍA

Nosotros, César Danilo Calle Loja y Julia Vanessa Morocho Guadalima declaramos ser autores del presente trabajo de tesis y eximimos expresamente a la Universidad Nacional de Loja y a sus representantes jurídicos de posibles reclamos o acciones legales por el contenido de la misma.

Adicionalmente acepto y autorizo a la Universidad Nacional de Loja, la publicación de nuestra tesis en el Repositorio Institucional-Biblioteca Virtual.

Autor: Julia Vanessa Morocho Guadalima

Firma: 

Cédula: 0704389675

Fecha: 28 de Julio del 2014.

Autor: César Danilo Calle Loja

Firma: 

Cédula: 1104899404

Fecha: 28 de Junio del 2014.

CARTA DE AUTORIZACIÓN DE TESIS POR PARTE DE LOS AUTORES, PARA LA CONSULTA, REPRODUCCIÓN PARCIAL O TOTAL Y PUBLICACIÓN ELECTRÓNICA DEL TEXTO COMPLETO.

Nosotros **César Danilo Calle Loja** y **Julia Vanessa Morocho Guadalima** declaramos ser autores de la tesis titulada: **“TECNIFICACIÓN DEL SISTEMA MUNICIPAL DE ESTACIONAMIENTO ROTATIVO TARIFADO (SIMERT) DE LA CIUDAD DE LOJA, MÓDULO CONECTIVIDAD DE LA RED DE DATOS DEL SIMERT UTILIZANDO LA TECNOLOGÍA GPRS”** como requisitos para optar al grado de: **Ingenieros en Sistemas**; autorizamos al Sistema Bibliotecario de la Universidad Nacional de Loja para que con fines académicos, muestre al mundo la producción intelectual de la Universidad, a través de la visibilidad de su contenido de la siguiente manera en el Repositorio Digital Institucional:

Los usuarios pueden consultar el contenido de este trabajo en el RDI, en las redes de información del país y del exterior, con las cuales tenga convenio la Universidad. La Universidad Nacional de Loja, no se responsabiliza por el plagio o copia de la tesis que realice un tercero. Para constancia de esta autorización, en la ciudad de Loja a los 28 días del mes de Julio del dos mil catorce.

Firma: 

Autor: **César Danilo Calle Loja**

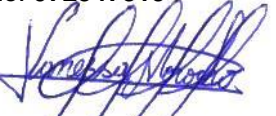
Cédula: **1104899404**

Email: **cesar.c.jtam@gmail.com**

Dirección: **Cdla. Daniel Álvarez Burneo, Av. Benjamín Carrión y Miguel Morelos.**

Teléfono: **072547819**

Celular: **0996955594**

Firma: 

Autor: **Julia Vanessa Morocho Guadalima**

Cédula: **0704389675**

Email: **juvamogu@gmail.com**

Dirección: **Cuidada de Machala Calles Gran Colombia entre la Onceava y Décima Oeste.**

Teléfono: **072967281**

Celular **0968611327**

DATOS COMPLEMENTARIOS

Director de Tesis:

Ing. Hernán Leonardo Torres Carrión, Mg. Sc.

Tribunal de Grado:

Ing. Alex Vinicio Padilla Encalada, Mg. Sc.

Ing. Edwin Rene Guamán Quinche, Mg. Sc.

Ing. Luis Roberto Jácome Galarza, Mg. Sc.

AGRADECIMIENTO

Quiero expresar mi más sincero agradecimiento al Ing. Hernán Leonardo Torres Carrión, Mg. Sc. por su importante aporte y participación activa en el desarrollo de esta tesis. Debo destacar, por encima de todo, su disponibilidad y paciencia. No cabe duda que su participación ha enriquecido el trabajo realizado y, además, ha significado el surgimiento de una sólida amistad. Agradezco a Dios por haberme acompañado y guiado a lo largo de mi carrera, por ser mi fortaleza en los momentos de debilidad y por brindarme una vida llena de aprendizajes, experiencias y sobre todo felicidad.

Le doy gracias a mis padres César y Martha por apoyarme en todo momento, por los valores que me han inculcado, y por haberme dado la oportunidad de tener una excelente educación en el transcurso de mi vida. Sobre todo por ser un excelente ejemplo de vida a seguir.

A mis hermanos por ser parte importante de mi vida y representar la unidad familiar. A Darwin y Lenin por ser un ejemplo de desarrollo profesional a seguir.

César Danilo Calle Loja

Al culminar con éxito este trabajo, me es muy grato expresar mi más sincero agradecimiento a las autoridades y catedráticos de la carrera de Ingeniería en Sistemas de la Universidad Nacional de Loja, quienes con su significativa labor educativa nos ayudan a descubrir y abrir las puertas del conocimiento, contribuyendo así a la formación profesional en el campo de la informática.

De manera muy especial al Ing. Hernán Leonardo Torres Carrión, Mg. Sc., quién con su acertada dirección y orientación supo guiarme a la exitosa culminación del presente proyecto investigativo.

Finalmente, a todas y cada una de las personas que con su desinteresado apoyo lograron motivarme para obtener el resultado final de mucho esfuerzo, la graduación.

Julia Vanessa Morocho Guadalima

DEDICATORIA

Dedico este trabajo a mis padres Cesar Augusto Calle Briceño y Martha Margarita Loja Loja, que son la figura más importante de mi vida, ellos me dieron la vida, amor, educación y sus sanos consejos, supieron formarme como una persona de bien, preparada para los retos que pone la vida.

A mis hermanos Darwin Calle y Lenin Calle que siempre me han brindado su apoyo incondicional en todo momento, juntos hemos compartido los mejores momentos de mi vida.

A mi familia que siempre ha estado pendiente de mis pasos, han sido luz en mi camino, me han mostrado los senderos del bien, de los triunfos y de los éxitos.

A mis amigos incondicionales, que son la familia que yo elegí en la vida, gracias a su amistad y sus consejos he alcanzado todas mis metas, ustedes son testigos y parte de mis éxitos.

Y sobre todo dedico este trabajo a Dios, que por su gracia he podido alcanzar todas mis metas y sueños.

César Danilo Calle Loja

Este trabajo va dedicado primeramente a Dios por ser quién ha estado a mi lado en todo momento dándome las fuerzas necesarias para llegar a esta meta.

A mis queridos papitos Rosa Enidth Guadalima Tacuri y José Leonardo Morocho Correa porque me supieron dar su ejemplo digno de amor, superación, fortaleza y entrega haciendo de mi lo que hoy en día soy y me siento extremadamente orgullosa, a mis hermanos por su apoyo y la hermandad que me demuestran.

A mí querida abuelita Marina que ha sido mi segunda madre por su amor, sus consejos y sus oraciones que están presentes en mi vida.

A mi angelito bello la razón de mi vida, mi hijo Anthony Alejandro Yaguana quien ha estado presente en todo momento para la culminación de este proyecto que con su inocencia y su alegría iluminan mis días dándome su paciencia y comprensión. Y a toda mi familia junto a las personas que me apoyaron en el transcurso del desarrollo de este proyecto.

Julia Vanessa Morocho Guadalima

CESIÓN DE DERECHOS

César Danilo Calle Loja y Julia Vanessa Morocho Guadalima, autores intelectuales del presente proyecto de fin de carrera, autorizan a la universidad Nacional de Loja, al Área de Energía, las industrias y los Recursos Naturales no Renovables y por ende a la carrera de Ingeniería en Sistemas hacer uso del mismo en lo que estime conveniente.



César Danilo Calle Loja

1104899404



Julia Vanessa Morocho Guadalima

0704389675



a. Título

“TECNIFICACIÓN DEL SISTEMA MUNICIPAL DE ESTACIONAMIENTO ROTATIVO TARIFADO (SIMERT) DE LA CIUDAD DE LOJA, MÓDULO CONECTIVIDAD DE LA RED DE DATOS DEL SIMERT UTILIZANDO LA TECNOLOGÍA GPRS”

b. Resumen

El presente trabajo de titulación se lo ha elaborado con el fin de realizar la conectividad de la red de datos del Sistema Municipal De Estacionamiento Rotativo Tarifado (SIMERT) del Cantón Loja, utilizando la tecnología GPRS; cuya intención es potenciar la ejecución de los procesos que realiza diariamente el personal operativo y administrativo del SIMERT con los registros de estacionamiento vehicular, lo que permite una adecuada organización y fiabilidad de la información.

Para el desarrollo de este proceso de investigación, se ha utilizado métodos, técnicas y se ha creado una adecuada metodología, iniciando con un análisis de requerimientos, diseño, implementación y medición de resultados, la misma que nos ha permitido cumplir con las actividades de cada una de las fases de manera organizada y determinar los diferentes problemas existentes para realizar consecuentemente una solución apropiada.

Dando como resultado la implementación de la arquitectura cliente-servidor para llevar a cabo la conectividad de la red de datos del SIMERT, a través de la implementación de un servidor y creación de una aplicación sockets para el mismo denominada "SERVER-SIMERT", garantizando la seguridad en la comunicación durante el envío y recepción de la información de los registros de estacionamiento vehicular del SIMERT; con el uso de la tecnología GPRS bajo el protocolo TCP/IP, direccionamiento IP y utilizando la clase ServerSockets dentro de la aplicación.

El servidor será una distribución Linux con el sistema operativo Ubuntu Server Lucid Lynx 10.04 bajo la plataforma libre LAMP, el cual será configurado correctamente para que funcione como servidor de correo, servidor web, servidor de base de datos y servidor de aplicaciones, utilizando la base de datos Mysql v. 5.5.2., el Phpmyadmin v. 5.1.72, Postfix v. 2.7 para el servicio de correos y el apache 2.4.2. Para la creación de la aplicación sockets denominada "SERVER-SIMERT", se ha utilizado el lenguaje de programación Java bajo la plataforma virtual IDE NetBeans 8.0, el conector para la base de datos -java 5.1.6 -bin.jar y las librerías Jasypt 1.9.1, Commons-dbcp-1.4 y Commons-pool-1.6.

Cumpliendo así con cada uno de los objetivos planteados al inicio del proyecto, los cuales se describen detalladamente en la discusión.

Summary

This research work has been developed in order to do the network connectivity data from the Municipal Parking Rotary Tariffed System (SIMERT) of Loja Canton using GPRS technology; whose purpose is to improve the implementation of the processes that staff of SIMERT carries out daily with the recording of the parking, which allow the appropriate organization and reliability of information.

For the development of this research process; methods, techniques were used, and an appropriate methodology was created, beginning with requirements analysis, design, implementation and results measurement, which has let us to fulfill the activities of each one of the phases in an organized way, and to determine the existing problems to give an appropriate solution.

Resulting in the implementation of the client-server architecture to perform the network connectivity data SIMERT, through the implementation of a server and the creation sockets, application named "SERVER-SIMERT" ensuring communication security for sending and receiving information from the records of vehicular parking SIMERT; using GPRS technology under the TCP / IP protocol, addressing IP and using ServerSockets class within the application.

The server will be a Linux distribution with Ubuntu Server Lucid Lynx 10.04 operating system under the free platform LAMP, which will be configured properly to work as a mail server, web server, data base and application server, using the Mysql database. 5.5.2., The Phpmyadmin v. 5.1.72, Postfix v. 2.7 for the postal service and apache 2.4.2. For the creation of the sockets application named "SERVER-SIMERT" it was used Java programming language under the IDE NetBeans 8.0 virtual platform, the connector database java-5.1.6-bin.jar and libraries Jasypt 1.9.1, Commons-dbc-1.4 and Commons-pool-1.6.

Thus fulfilling each of the goals set at the beginning of the project, which are described in detail in the discussion¹.

¹ Ver Anexo 17: Certificado de la traducción del Resumen.

ÍNDICE DE CONTENIDOS

CERTIFICACIÓN DEL DIRECTOR	II
AUTORÍA	III
CARTA DE AUTORIZACIÓN DE TESIS POR PARTE DE LOS AUTORES, PARA LA CONSULTA, REPRODUCCIÓN PARCIAL O TOTAL Y PUBLICACIÓN ELECTRÓNICA DEL TEXTO COMPLETO.	IV
AGRADECIMIENTO	V
DEDICATORIA	VI
CESIÓN DE DERECHOS	VII
a. Título	VIII
Summary	X
ÍNDICE DE CONTENIDOS	XI
ÍNDICE DE FIGURAS	XVI
ÍNDICE DE TABLAS	XVIII
c. Introducción	20
d. Revisión de literatura	22
1. Capítulo I: Antecedentes del Sistema Municipal de Estacionamiento Rotativo Tarifado (SIMERT)	22
1.1. Introducción	22
1.2. Descripción	22
1.3. Operatividad del SIMERT	23
1.3.1. Estacionamiento	23
1.3.2. Emisión de citaciones o multas	24
1.3.3. Inmovilizaciones	24
1.3.4. Proceso administrativo	25
2. Capítulo II: SERVIDOR	25
2.1. Definición	25
2.2. Arquitectura cliente- servidor para la conectividad de la red de datos del SIMERT	25
3. Capítulo III: Aplicación Sockets	27
3.1. Descripción de la aplicación sockets denominada “SERVER-SIMERT”	27
3.2. GPRS	28



3.3. Redes y protocolo TCP / IP	29
3.4. Sockets.....	29
3.4.1. Definición	29
3.4.2. Propiedades inherentes a los sockets	29
3.4.3. Modelo de comunicaciones con Sockets en Java	30
3.5. Incorporación de librerías para la aplicación de red.....	34
3.5.1. Librería Jasypt	34
3.6. Librería commons-dbcp y commons-pool	36
e. Materiales y métodos.....	38
1. MÉTODOS.....	38
1.1. Método Deductivo	38
1.2. Método Inductivo	38
1.3. Método Científico Experimental.....	38
2. TÉCNICAS	39
2.1. Observación	39
2.2. Entrevista.....	39
2.3. Encuesta	40
3. METODOLOGÍA PARA LA CONECTIVIDAD DE LA RED DE DATOS DEL SIMERT.....	40
3.1. Fase 1: Análisis de requerimientos	40
3.2. Fase 2: Diseño de la infraestructura para la conectividad de la red de datos. 41	
3.3. Fase 3: Implementación	42
3.4. Fase 4: Medición de resultados.	43
f. Resultados.....	44
1. FASE 1: ANÁLISIS DE REQUERIMIENTOS	44
1.1. Recopilación de información confiable de la estructura organizacional del Sistema Integrado Municipal de Estacionamiento Rotativo Tarifado (SIMERT)..	44
1.2. Recopilación de información sobre el perfil técnico y actividades que realiza el personal del SIMERT.	47
1.3. Recopilar información sobre los recursos que actualmente posee el SIMERT. 49	
1.4. Recopilar información acerca de los proveedores que brinden el servicio de GPRS.	50

1.5. Análisis técnico.....	51
1.6. Análisis de seguridad.....	53
1.7. Analizar las aplicaciones y Herramientas de Desarrollo.	54
1.8. Definir la topología que se utilizará para la conectividad de la red de datos del SIMERT.	54
1.9. Análisis de los requerimientos funcionales y no funcionales.	55
1.9.1.Requerimientos Funcionales	55
1.9.2.Requerimientos No Funcionales	56
2. FASE 2: DISEÑO DE LA INFRAESTRUCTURA PARA LA CONECTIVIDAD DE LA RED DE DATOS.	57
2.1. Prototipado Inicial.....	57
2.2. Diseño del Modelo de casos de uso.....	61
2.3. Descripción de los casos de uso.....	61
2.3.1.Caso de uso 001: Iniciar servidor	61
2.3.2.Caso de uso 002: Cambiar Contraseña.....	64
2.3.3.Caso de uso 003: Salir Servidor	66
2.3.4.Caso de uso 004: Visualizar envío y recepción de tramas.	68
2.4. Diseño del Modelo de procesos o robustez.....	74
2.4.1.Caso de uso 001: Iniciar servidor	74
2.4.2.Caso de uso 002: Cambiar Contraseña.....	75
2.4.3.Caso de uso 003: Salir Servidor	76
2.5. Diseño del Modelo de secuencia.	77
2.5.1.Caso de uso 001: Iniciar servidor	77
2.5.2.Caso de uso 002: Cambiar Contraseña.....	78
2.5.3.Caso de uso 003: Salir del Servidor	79
2.6. Diseño de la Arquitectura cliente-servidor para la conectividad de la red de datos del SIMERT.....	80
2.6.1.Simbología de la arquitectura cliente – servidor	81
3. FASE 3: IMPLEMENTACIÓN.....	84
3.1. Creación del servidor SIMERT.....	84
3.1.1.Función del servidor.....	84
3.1.2.Plataforma para el Servidor.	84
3.1.3.Creación de la cuenta para el acceso al Servidor	85
3.1.4.Selección de Paquetes para el servidor.....	86



3.1.5. Instalación De Entorno Gráfico del servidor.	86
3.1.6. Configuraciones para la creación del servidor	87
3.1.6.1. Configuración de tarjetas de Red.	87
3.1.6.2. Configuración del Servidor como servidor de Aplicaciones	88
3.1.6.3. Configuración del Servidor como servidor de Dominios (DNS).....	88
3.1.6.4. Configuración del Servidor como servidor de base de datos.	88
3.1.6.4.1. Configuración de roles en phpmyadmin.	88
3.1.6.5. Configuración del Servidor SIMERT como servidor web Apache y su VirtualHost con Nalp	90
3.1.6.6. Configuración del Servidor SIMERT como Servidor de Correo.....	91
3.1.6.6.1. Instalación de Postfix.....	91
3.1.6.6.3. Autenticación de correo	93
3.1.6.6.4. Utilización del certificado adecuado.	94
3.1.6.6.5. Añadir un remitente nuevo en nuestra cuenta de gmail.....	94
3.1.6.6.6. Puesta en funcionamiento.....	95
3.1.7. Configuración física del Router	96
3.1.8. Configuración del Proxy	96
3.2. Codificación o generación de código Java para la aplicación sockets denominada “SERVER-SIMER” para el servidor.	97
3.2.1. Requerimientos de hardware y software para la instalación y utilización de la aplicación “server-SIMERT”.	97
3.2.1.1. Requerimientos de hardware	97
3.2.1.2. Requerimientos de software	97
3.2.2. Estándares para el código de la aplicación “SERVER-SIMERT”.	98
3.2.3. Arquitectura de la aplicación Sockets “SERVER-SIMERT”	101
3.2.3.1. CAPA VISTA	102
3.2.3.2. CAPA CONTROLADOR	103
3.2.3.3. CAPA MODELO	103
3.3. Simulación en el Packet Tracer de la arquitectura cliente – servidor para la conectividad de la red de datos del SIMERT.	105
3.3.1. Realizar el subneteo.	106
3.3.2. Asignar los Ip’s para las computadoras del SIMERT.....	107
4. FASE 4: MEDICIÓN DE RESULTADOS.	108
4.1. PRUEBAS DEL SERVIDOR	108



4.1.1. Pruebas de carga al servidor	108
4.2. PRUEBAS DE LA APLICACIÓN SOCKETS “SERVER-SIMERT”	111
4.2.1. Pruebas Unitarias	111
4.2.1.1. Validación del acceso a la aplicación “SERVER-SIMERT” a través del nombre de usuario y contraseña.....	111
4.2.1.2. Validación de la trama “Registro enviar zonas”	114
4.2.2. Pruebas de Validación.	116
4.2.2.1. Iniciar Servidor.....	116
4.2.2.3. Salir del Servidor.....	118
4.2.2.4. Visualización de las tramas recibidas acerca de los registros de estacionamiento vehicular del SIMERT.....	119
4.2.3. Prueba de Accesibilidad, funcionalidad y usabilidad.	120
4.2.3.1. Análisis de Prueba: Administrador del Servidor.....	120
g. Discusión.....	134
1. Desarrollo de la propuesta alternativa	134
2. Valoración Técnica Económica Ambiental	138
2.1. Valoración Técnico Económica	138
2.2. Valoración Ambiental	141
h. Conclusiones.....	142
i. Recomendaciones	144
j. Bibliografía	145
k. Anexos	147

ÍNDICE DE FIGURAS

Figura 1: Insertar librerías para la aplicación Sockets denominada "SERVER-SIMERT"	34
Figura 2: Organigrama Estructural para el SIMERT.....	45
Figura 3: Prototipado Interfaz "Iniciar Servidor"	57
Figura 4: Prototipado interfaz "Servidor SIMERT"	58
Figura 5: Prototipado Interfaz "Cambiar Usuario y Contraseña"	59
Figura 6: Prototipo Interfaz "Salir del Servidor"	60
Figura 7: Interfaz "Iniciar Servidor"	61
Figura 8: Interfaz "Servidor SIMERT"	62
Figura 9: Interfaz "Cambiar Usuario y Contraseña"	64
Figura 10: Interfaz "Salir del Servidor"	66
Figura 11: Visualización de la trama "Registro enviar zonas a los controladores"	68
Figura 12: Diagrama de robustez "Iniciar Servidor"	74
Figura 13: Diagrama de robustez "Cambiar Usuario y Contraseña"	75
Figura 14: Diagrama de robustez "Salir del Servidor"	76
Figura 15: Diagrama de secuencia "Iniciar Servidor"	77
Figura 16: Diagrama de secuencia "Cambiar nombre de usuario y contraseña"	78
Figura 17: Diagrama de secuencia "Salir del Servidor"	79
Figura 18: Arquitectura Cliente - Servidor para la conectividad de la red de datos del SIMERT	80
Figura 19: Configuración de usuario y contraseña para el acceso al servidor.....	85
Figura 20: Ingreso del nombre de usuario para el acceso al servidor.....	85
Figura 21: Selección de paquetes para la creación del servidor.....	86
Figura 22: Pantalla principal para el servidor	86
Figura 23: Pantalla Principal de phpmyadmin.	89
Figura 24: Crear un nuevo usuario para el acceso al MYSQL	89
Figura 25: Ingreso de datos del usuario Mysql.....	89
Figura 26: Selección de Privilegios	90
Figura 27: Creación de VirtualHost	90
Figura 28: Configuración de correo.....	92
Figura 29: Ingreso del nombre para el sistema de Correos "simertnet"	92
Figura 30: Configuración del Nuevo Remitente	94
Figura 31: Configuración de Nombre del remitente	95
Figura 32: Elección de SMTP	95
Figura 33: Comprobación de envío de correo	96
Figura 34: Arquitectura Cliente-Servidor con MCV para la aplicación sockets "SERVER-SIMERT"	101
Figura 35: Interfaz gráfica "Iniciar Servidor"	102
Figura 36: Capa Vista.....	103
Figura 37: Capa Controlador.....	103
Figura 38. Capa Modelo.....	104

Figura 39: Arquitectura Cliente-Servidor para la conectividad de la red de datos del SIMERT en el Packet Tracer.	106
Figura 40: Comando de ejecución de Carga con Siege.....	109
Figura 41: Resultados de la prueba de carga con Siege.	110
Figura 42: Respuesta de Aprobación para el acceso a la aplicación "SERVER-SIMERT".	113
Figura 43: Respuesta de error para el acceso a la aplicación "SERVER-SIMERT"....	113
Figura 44: Respuesta de Aprobación de la trama "Registro enviar zonas de estacionamiento".	115
Figura 45: Respuesta de error de la trama "Registro enviar zonas de estacionamiento".	116
Figura 46: Análisis de la ejecución de la aplicación sockets "SERVER-SIMERT".	121
Figura 47: Análisis del acceso y finalización de la aplicación sockets "SERVER-SIMERT".....	121
Figura 48: Análisis del envío y recepción del registro "Enviar zonas a los controladores".	122
Figura 49: Análisis del envío y recepción del registro "Ingresar cliente exonerado o recarga".	123
Figura 50: Análisis del envío y recepción del registro "Ingresar multa por recarga"....	123
Figura 51: Análisis del envío y recepción del registro "Validar código de tarjeta".....	124
Figura 52: Análisis del envío y recepción del registro "Modificar espacios de tarjeta".	125
Figura 53: Análisis del envío y recepción del registro "Ingresar personal del municipio".	125
Figura 54: Análisis del envío y recepción del registro "Ingresar multa por tarjeta".	126
Figura 55: Análisis del envío y recepción del registro "Ingresar código de tarjeta".	127
Figura 56: Análisis del envío y recepción del registro "Ingresar nueva tarjeta".	127
Figura 57: Análisis del envío y recepción del registro "Ingresar cliente recarga estacionado".	128
Figura 58: Análisis del envío y recepción del registro "Ingresar datos posicionales". .	129
Figura 59: Análisis del envío y recepción del registro "Eliminar controlador de sesiones".	129
Figura 60: Análisis del envío y recepción del registro "Ingresar sincronización de multas".	130
Figura 61: Análisis de la velocidad en la transmisión de datos.....	131
Figura 62: Análisis de los requerimientos de la aplicación sockets "SERVER-SIMERT".....	132
Figura 63: Análisis de la interfaz gráfica de la aplicación sockets "SERVER-SIMERT".	133

ÍNDICE DE TABLAS

TABLA I: CÓDIGO PARA CREAR EL SOCKETS DEL SERVIDOR.....	31
TABLA II: CÓDIGO PARA ACEPTAR EL CLIENTE.....	31
TABLA III: CÓDIGO PARA OBTENER LA SALIDA Y ENTRADA DEL CLIENTE	32
TABLA IV: CÓDIGO PARA LA LECTURA DE TRAMAS	33
TABLA V: CÓDIGO PARA CERRAR EL SOCKET.....	33
TABLA VI. MÉTODO DE ENCRIPCIÓN.....	35
TABLA VII: MÉTODO DE DESENCRIPCIÓN	35
TABLA VIII. MÉTODO POOL DE CONEXIONES.....	37
TABLA IX: PROVEEDORES QUE BRINDAN EL SERVICIO GPRS.....	50
TABLA X: COMPARACIÓN DE LOS LENGUAJES DE PROGRAMACIÓN	51
TABLA XI: COMPARACIÓN DE LOS S.O. PARA SERVIDORES	52
TABLA XII: REQUERIMIENTOS FUNCIONALES.....	55
TABLA XIII: REQUERIMIENTOS NO FUNCIONALES	56
TABLA XIV: DESCRIPCIÓN CASO DE USO 001”INICIAR SERVIDOR”	62
TABLA XV. DESCRIPCIÓN CASO DE USO 002 “CAMBIAR CONTRASEÑA”	64
TABLA XVI: DESCRIPCIÓN CASO DE USO 003 “SALIR SERVIDOR”	66
TABLA XVII: DESCRIPCIÓN CASO DE USO 004 “VISUALIZAR ENVÍO Y RECEPCIÓN DE TRAMAS.”	68
TABLA XVIII: SIMBOLOGÍA DE LA ARQUITECTURA CLIENTE-SERVIDOR	81
TABLA XIX: COMANDO DE INSTALACIÓN DE INTERFAZ	86
TABLA XX: CONFIGURACIÓN DE LA TARJETA DE RED.....	87
TABLA XXI: CONFIGURACIÓN DE PROXY.....	88
TABLA XXII: COMANDO DE INSTALACIÓN PHPMYADMIN	88
TABLA XXIII: COMANDO DE REINICIO DE VIRTUALHOST	91
TABLA XXIV: COMANDO DE INSTALACIÓN DE POSTFIX Y DEPENDENCIAS	91
TABLA XXV: COMANDO DE EDICIÓN DE ARCHIVO MAIN.CF.....	92
TABLA XXVI: LÍNEAS DE CONFIGURACIÓN DE MAIN.CF	93
TABLA XXVII: AUTENTICACIÓN DE CORREO	93
TABLA XXVIII: COMANDO PARA CAMBIO DE PERMISOS.....	93
TABLA XXIX: COMANDO PARA LA CREACIÓN DE ARCHIVOS HASH.....	93
TABLA XXX: COMANDO PARA INSTALAR CERTIFICADOS	94
TABLA XXXI: COMANDO PARA AÑADIR CERTIFICADOS EQUIFAX.....	94
TABLA XXXII: COMANDO DE VISUALIZACIÓN DE ENVÍO DE CORREO	95
TABLA XXXIII: COMANDO DE ENVÍO DE CORREO.....	95
TABLA XXXIV: CONFIGURACIÓN FÍSICA DEL ROUTER	96
TABLA XXXV: CONFIGURACIÓN DE PROXY	96
TABLA XXXVI: CÓDIGO PARA LA CONEXIÓN CON LA BASE DE DATOS.	104
TABLA XXXVII: REALIZAR EL SUBNETEO.....	106
TABLA XXXVIII: CANTIDAD DE BITS.....	106
Tabla XXXIX: DESCRIPCIÓN DE SUBNETEO	107
TABLA XL: ASIGNACIÓN DE IPS	107



TABLA XLI: COMANDO DE INSTALACION DE SIEGE	108
TABLA XLII: COMANDO DE EJECUCION DE SIEGE	109
TABLA XLIII: DESCRIPCIÓN DE RESULTADOS DE CARGA CON SIEGE	110
TABLA XLIV: CÓDIGO FUENTE DEL MÉTODO “TESTVALIDARUSUARIOCONTRASEN”	111
TABLA XLV: CÓDIGO FUENTE PARA REALIZAR EL TESTCONFIRMATRAMA1.	114
TABLA XLVI: CASO DE PRUEBA - INICIAR SERVIDOR	117
TABLA XLVII: CASO DE PRUEBA - CAMBIAR NOMBRE DE USUARIO Y CONTRASEÑA.....	117
TABLA XLVIII: CASO DE PRUEBA - SALIR DEL SERVIDOR.....	118
TABLA XLIX. CASO DE PRUEBA - VISUALIZACIÓN DE LAS TRAMAS RECIBIDAS ACERCA DE LOS REGISTROS DE ESTACIONAMIENTO VEHICULAR DEL SIMERT.....	119
TABLA L: VALORACIÓN ECONÓMICA DE RECURSOS HUMANOS	138
TABLA LI: VALORACIÓN ECONÓMICA DE RECURSOS TÉCNICOS	139
TABLA LII: VALORACIÓN ECONÓMICA DE RECURSOS MATERIALES	140
TABLA LIII: VALORACIÓN ECONÓMICA DE RECURSOS DE SERVICIOS	141
TABLA LIV: TOTAL DE RECURSOS	141

c. Introducción

El Sistema Municipal De Estacionamiento Rotativo Tarifado (SIMERT) de la ciudad de Loja tiene como objetivo “conceder el derecho a todos los usuarios de utilizar la vía pública en forma organizada y ordenada, a través de la generación de una oferta permanente y continua de espacios libres de estacionamiento”, entrando en vigencia a partir del 3 de Mayo del 2002².

Actualmente el SIMERT no utiliza tecnología avanzada que potencie su funcionamiento y servicio a la ciudadanía lojana, afrontando problemas como: falta de seguridad y agilidad en el almacenamiento y recopilación de información acerca de los registros de estacionamiento vehicular ya que se lo realiza de forma manual en hojas impresas y falta de algún tipo de conexión entre los diferentes departamentos del SIMERT que facilite y asegure el control y administración de la información.

La parte administrativa y personal operativo del SIMERT, se han dado cuenta que se está volviendo muy difícil llevar la gestión de los registros de estacionamiento vehicular y obtener información verídica de los vehículos estacionados diariamente de forma manual, puesto que al transcurrir los años existe una mayor demanda de usuarios, la información crece cada día, los procesos de recuperación de la misma son más complejos y las hojas se ven expuestas al deterioro y destrucción por factores ambientales y descuido de las personas que los custodian.

Como solución a estos problemas se han creado cuatro módulos desarrollando diferentes sistemas informáticos para cubrir los procesos que conforman el SIMERT esto con el objetivo de realizar la Tecnificación del mismo. El presente proyecto específicamente se contempla en llevar a cabo el módulo denominado “Conectividad de la red de datos del Sistema Municipal De Estacionamiento Rotativo Tarifado (SIMERT) utilizando la tecnología GPRS” a través de la implementación de la arquitectura cliente-servidor, implementación de un servidor, desarrollo de una aplicación sockets denominada “SERVER - SIMERT” bajo la arquitectura Modelo Vista Controlador (MVC) y uso de la clase ServerSockets; facilitando las tareas de búsqueda y almacenamiento, organizando la información, integrando los datos entre los diferentes departamentos del SIMERT y proporcionando niveles de seguridad tanto en la comunicación como para el

² Memorando N°715-SGM-2012g. Ordenanza que regula el Sistema Municipal De Estacionamiento Rotativo Tarifado del Cantón Loja, “SIMERT”.

acceso al servidor y su aplicación. Cumpliendo así con los objetivos planteados tales como: realizar un análisis de requisitos para llevar a cabo la conectividad de la red de datos entre las diferentes subáreas del SIMERT, realizar el diseño de la arquitectura cliente-servidor para establecer la comunicación, implementar la arquitectura cliente-servidor para la conectividad de la red de datos del SIMERT y realizar las pruebas pilotos para verificar la conectividad en la red de datos del SIMERT.

La metodología creada para la conectividad de la red de datos permitió una planificación adecuada de las diferentes tareas a ejecutarse para el cumplimiento de los objetivos anteriormente mencionados a través de las siguientes fases como: Fase 1: Análisis de requerimientos, donde se ha recopilado toda la información para llevar a cabo el proyecto con el uso de métodos y técnicas; Fase 2: Diseño de la infraestructura para la conectividad de la red de datos, donde se ha realizado los diferentes modelados para la visión global del proyecto; Fase 3: implementación, donde se ha creado el código fuente y se ha realizado las debidas configuraciones tanto en el servidor como en la aplicación haciendo uso del análisis y diseño; finalmente la Fase 3: Medición de resultados, donde se aplicaron diferentes pruebas en la aplicación y en el servidor para el funcionamiento y calidad de los mismos.

La estructura del informe inicia con un resumen del Proyecto de Fin de Carrera que condensa los aspectos más importantes del trabajo realizado, seguidamente una breve introducción que informa al lector sobre el contenido del informe, a continuación se presenta el desarrollo del marco teórico que está compuesto por toda la información de las herramientas utilizadas para la construcción del proyecto.

A continuación del marco teórico se explica los métodos y técnicas aplicadas, así como la metodología que se ha creado para llevar a cabo la conectividad de la red de datos del SIMERT.

El informe finaliza con la discusión de resultados, las conclusiones que nos permite dar el cumplimiento de los objetivos planteados y los resultados obtenidos. Luego las recomendaciones que expresan sugerencias para qué, en un futuro mejorar la conectividad de la red de datos del SIMERT, así también consideraciones que se debe tomar en cuenta para el desarrollo de proyectos de igual similitud. Al igual que la bibliografía que es la lista de recursos consultados y finalmente los anexos en donde se detalla información complementaria y también necesaria para la ejecución del presente proyecto.

d. Revisión de literatura

1. Capítulo I: Antecedentes del Sistema Municipal de Estacionamiento Rotativo Tarifado (SIMERT)

1.1. Introducción

Los sistemas de estacionamiento rotativo tarifado crean un mecanismo que apunta a generar la rotación de vehículos estacionados; permite establecer los horarios de estacionamiento y determinan zonas específicas para estos efectos.

En el Ecuador algunas ciudades utilizan este sistema como una forma de organizar el tráfico y solucionar los problemas de tiempo excesivo de transportación, operación ineficiente y costosa, zonas sin servicio, congestión y parque vehicular contaminante.

1.2. Descripción

El Sistema Municipal De Estacionamiento Rotativo Tarifado (SIMERT), se creó mediante ordenanza el dieciocho de Marzo del dos mil dos, con el objetivo de devolverles a todos los usuarios el derecho a utilizar la vía pública en forma organizada y ordenada, a través de la generación de una oferta permanente y continua de espacios libres para estacionamiento y reducir la contaminación ambiental provocada por el flujo continuo de vehículos

Este sistema funciona en la ciudad de Loja desde el 3 de Mayo del 2002 y desde Enero del 2008 viene funcionando como un área independientemente de la Unidad de Tránsito. Sus oficinas están situadas en el edificio de la Policía Municipal, en las calles Bernardo Valdivieso y Colón esq. Luego de la reestructuración, la dependencia cuenta con 50 personas entre controladores, inspectores y administrativos. El personal que labora en esta área, a más de velar por el correcto uso de las tarjetas, colabora en la seguridad de los vehículos, ya que cuenta con un sistema de comunicación permanente que les permite solicitar ayuda en caso de ser necesario.

El Horario es de lunes a viernes de 08:00 a 19:00 y el sábado de 09:00 a 13:00 [3].

1.3. Operatividad del SIMERT.

La operatividad del SIMERT, se resume en cuatro flujos claramente establecidos que hacen referencia a las acciones que generan los ingresos para el sistema.

- Estacionamiento
- Emisión de Citaciones
- Inmovilizaciones
- El proceso administrativo

1.3.1. Estacionamiento

En este proceso intervienen los tres factores que son:

- El usuario,
- El controlador,
- Y la tarjeta prepago del SIMERT donde:

Usuario: Es la persona que utiliza el SIMERT.

Controlador: Es la persona autorizada por parte de la Municipalidad para que cumpla las funciones de inspector. Cada controlador tendrá asignada un área específica en la cual, podrá realizar múltiples registros dependiendo de los vehículos que se estacionen en dicha área.

Tarjeta prepago: es el único requisito para que el usuario pueda hacer uso del SIMERT (ver Anexo 9).

El tiempo máximo de estacionamiento continuo es de 3 horas según la zona ya sea el cordón interior y el cordón exterior

La tarifa para la utilización del SIMERT se fija en veinticinco centavos de dólar por cada sesenta minutos, recaudados a través de un tipo de tarjeta prepago. La tarjeta A que tiene un costo de 1.50 dólares con un período de duración de 90 días contados a partir de la fecha de su primer uso. La fracción de tiempo mínima de uso para este tipo de tarjeta es de 30 minutos.

Actualmente la utilización del SIMERT se resume en los siguientes pasos:

POR PARTE DEL USUARIO:

- Comprar la tarjeta del SIMERT.
- Estacionar el vehículo en el lugar señalado y deseado.

- Indicar en la tarjeta la fecha exacta de uso del estacionamiento, bajo el formato mes y día; adicionalmente deberá llenar el campo de la hora, indicando la hora exacta de llegada.
- Luego procede a señalar los recuadros indicando el tiempo que va a ser uso del estacionamiento,
- El usuario finalmente procederá a ubicar la tarjeta en la parte frontal del parabrisas para que sea vista por el controlador.

POR PARTE DEL CONTROLADOR E INSPECTORES

Los inspectores y controladores serán las personas autorizadas por el Municipio para hacer cumplir las normas relacionadas con el SIMERT, quienes tendrán asignadas áreas específicas en las cuales podrán registrar cada uno de los vehículos con los respectivos tiempos de estacionamiento, esta información se la lleva manualmente en hojas impresas denominadas: “fichas de control diarias” (ver Anexo 8).

1.3.2. Emisión de citaciones o multas.

El usuario tiene un tiempo de “gracia” de 15 minutos, pero si el tiempo sobrepasado es de 16 minutos a 30 minutos o más, el controlador extenderá una citación al usuario.

La citación mínima tiene un costo de USD \$3 y deberá ser cancelada en 48 horas por parte del usuario en la Oficina de Recaudaciones del SIMERT.

Las citaciones son emitidas al usuario por parte de los controladores.

1.3.3. Inmovilizaciones

Se procederá a inmovilizar el vehículo y emitir una citación por el controlador al usuario por infracción en los siguientes casos: si el tiempo sobrepasado es de 31 a 60 minutos, la multa es de 5 dólares. De 61 a 120 minutos la multa es de 10 dólares y si es mayor de 121 minutos la multa es de 20 dólares.

Para proceder a des inmovilizar el vehículo, el usuario deberá cancelar en forma inmediata estos valores en las oficinas de Recaudaciones del SIMERT.

El tiempo de inmovilización será tarifado como uso normal del SIMERT. Igual sanción y procedimiento si el usuario hiciere el uso del SIMERT sin utilización de la tarjeta prepago.

La inmovilización y des inmovilización de vehículos son realizadas por los controladores.

1.3.4. Proceso administrativo

La parte administrativa del SIMERT realiza las siguientes actividades:

- Control del personal operativo que cumpla con el horario y la zona establecida de trabajo.
- Administrar los horarios y lugares de trabajo para los controladores.
- Atender sugerencias y reclamos por parte de los usuarios y empleados.
- Brindar información a los usuarios que utilizan el SIMERT.
- Modificación de multas emitidas por los controladores en casos necesarios.
- Imprimir y generar reportes de lo siguiente:
 - Ingresos y egresos del SIMERT diaria, semanal, mensual y anualmente.
 - Reporte de las multas recaudadas diariamente.
 - Reporte de las ventas de tarjetas SIMERT realizadas diariamente.
 - Reporte de tasas de rotación, ocupación y estacionamiento calculadas a través de las fichas de control diarias que registran los controladores acerca de la información de los registros de estacionamiento vehicular.

2. Capítulo II: SERVIDOR

2.1. Definición

Un servidor es un tipo de software que realiza ciertas tareas en nombre de los usuarios. Los usuarios se conectan a un servidor pueden acceder a programas, archivos y otra información del servidor. Algunos servidores manejan solamente correo o solamente archivos, mientras que otros hacen más de un trabajo, ya que un mismo ordenador puede tener diferentes programas de servidor funcionando al mismo tiempo.

Los servidores se conectan a la red mediante una interfaz que puede ser una red verdadera o mediante conexión vía línea telefónica o digital.

El servidor a implementar realizará varias funciones como servidor de correos, servidor de aplicaciones, servidor web y servidor de base de datos. Potenciando la seguridad en la información y agilidad en las diversas actividades que realiza diariamente el SIMERT.

2.2. Arquitectura cliente- servidor para la conectividad de la red de datos del SIMERT

A continuación se describirá la arquitectura cliente – servidor como parte del proyecto denominado “Tecnificación del sistema municipal de estacionamiento rotativo tarifado

(SIMERT) de la ciudad de Loja, módulo conectividad de la red de datos del SIMERT utilizando la tecnología GPRS”.

Como se muestra en la (Figura 18) el proyecto general Tecnificación del Sistema Municipal de Estacionamiento Rotativo Tarifado consta de cuatro módulos desarrollando diferentes aplicaciones como: el módulo de administración para el sistema de la información a través del desarrollo de una aplicación web que abarca la parte administrativa del SIMERT, módulo software para el control vehicular y geolocalización del personal operativo a través del desarrollo de una aplicación móvil Android que utilizarán los controladores para realizar los procesos de control vehicular y emisión de multas de este Sistema, módulo de servicios en aplicaciones móviles a través del desarrollo de una aplicación de recargas web multiplataforma que abarca la parte administrativa de recargas y una aplicación móvil la cual está destinada a las personas que hacen uso del servicio de estacionamiento vehicular y finalmente el módulo conectividad de la red de datos del SIMERT utilizando la tecnología GPRS a través de la implementación de un servidor y desarrollo de una aplicación sockets denominada “SERVER-SIMERT” para el servidor.

Este proyecto se centra en la implementación del servidor para el SIMERT y desarrollo de una aplicación sockets para el mismo denominado “SERVER-SIMERT”.

El servidor será configurado e instalado en el centro de cómputo del SITU en conjunto con la aplicación sockets, la cual tendrá como usuario final al administrador del servidor quien será la persona encargada de utilizar dicha aplicación, este software permitirá recibir, enviar, guardar, eliminar, modificar, consultar y visualizar la información en tramas acerca de los procesos de control de estacionamiento vehicular recibida por los controladores. Esta información a su vez será utilizada por la administración de recargas web y la administración del SIMERT.

Como canal de comunicación entre el servidor y los clientes que serán los controladores se ha hecho uso de la tecnología GPRS bajo el protocolo TCP/IP y direccionamiento IP y la clase ServerSockets para la aplicación.

2.3. Correo Electrónico

2.3.1. SMTP

SMTP (simple mail transfer protocolo, protocolo simple de transferencia de correo), se basa en una entrega punto a punto, un cliente SMTP se contacta con el servidor SMTP

del host destino para entregarle directamente el correo electrónico, este nos da seguridad en la entrega al receptor ya que espera que sea guardado con éxito. Funciona conjuntamente con los protocolo POP o IMAP [1].

2.3.2. MUA

MUA (Mail User Agent, Agente de usuario de correo) MUA (Mail User Agent, Agente de usuario de correo) es un programa que permite a un usuario, como mínimo, leer y escribir mensajes de correo electrónico. Los programas MUA pueden ser gráficos, como Mozilla Mail, o pueden tener una interfaz basada en texto sencilla, como Mutt o Pine

2.3.2.1. MTA Agente de transporte de correo Postfix

Postfix que es un Agente de Transporte de Correo (MTA) de código abierto, o lo que es lo mismo, un programa informático para el enrutamiento y envío de correo electrónico [5].

Fue creado como alternativa a Sendmail, buscando un servidor que fuera más rápido, fácil de administrar y seguro, desarrollado por Wietse Venema durante una estancia en el Centro de Investigación Thomas J. Watson de IBM y fue conocido con el nombre de VMailer e IBM Secure Mailer, siendo distribuido al gran público por primera vez a mediados de 1999. Desde entonces sigue en proceso de desarrollo y mejora de una forma activa.

Se Distribuye bajo Licencia Pública IBM v.1.0, que a pesar de ser una licencia de software libre, resulta incompatible con la Licencia GPL.

Postfix es de hecho, el MTA que se usa por defecto en muchos sistemas operativos derivados de UNIX, entre ellos, GNU/Linux. Con capacidad de manejar altos volúmenes de correo.

3. Capítulo III: Aplicación Sockets.

3.1. Descripción de la aplicación sockets denominada “SERVER-SIMERT”

La aplicación sockets denominada “SERVER-SIMERT” ha sido creada para potenciar la seguridad e integridad en la recepción, envío y gestión de la información de los registros de estacionamiento vehicular correspondientes al SIMERT de la ciudad de Loja.

Para el desarrollo de aplicación sockets “SERVER-SIMERT” se ha utilizado el lenguaje de programación java bajo plataforma virtual de Java NETBEANS IDE 7.0.4 que es un entorno de desarrollo integrado libre, hecho principalmente para el lenguaje de programación Java que proporciona una arquitectura de aplicación fiable y flexible. Esta aplicación permite al usuario “administrador del servidor” acceder al sistema a través de su autenticación con el nombre de usuario y contraseña las mismas que serán validadas, en caso de ser correctas la aplicación sockets se ejecuta para establecer la conexión con los clientes a través de una dirección IP y número de puerto. Los clientes que son los controladores utilizan la aplicación móvil Android para el envío y recepción de los registros de estacionamiento vehicular hacia el servidor. Los datos de los registros de estacionamiento vehicular llegan y salen del servidor a través de tramas.

Como seguridad para el acceso de los controladores al servidor, las tramas llegan encriptadas y serán des encriptadas por una clave definida tanto para el cliente como para el servidor, luego serán validadas por los siguientes parámetros: número de trama, inicio y fin de trama con caracteres especiales y número de datos. Finalmente se valida el nombre de usuario y contraseña del controlador junto con la mac del dispositivo móvil Android.

Una vez que los controladores han accedido al servidor se procede a cargar las zonas de trabajo y se realiza los diferentes procesos como consultar, guardar, modificar y eliminar de la base datos; la información de los registros de estacionamiento vehicular que se reciban en cada una de las tramas, hasta que el controlador finalice su horario de trabajo y cierre la conexión con el servidor.

3.2. GPRS

GPRS es una tecnología que comparte el rango de frecuencias de la red GSM utilizando una transmisión de datos por medio de “paquetes”. La conmutación de paquetes es un procedimiento más adecuado para transmitir datos, antes los datos se habían transmitido mediante conmutación de circuitos, procedimiento más adecuado para la transmisión de voz.

Como canal de comunicación entre los clientes que serán los controladores y el servidor se utilizará la tecnología GPRS, este canal se comparten entre los distintos usuarios

dinámicamente, de modo que un usuario sólo tiene asignado un canal cuando se está realmente transmitiendo datos.

GPRS utiliza el protocolo de comunicación TCP/IP que hace posible la transferencia de datos entre redes de ordenadores, hace referencia a dos protocolos: el conocido como Protocolo de Control de Transmisión o (TCP) y el llamado Protocolo de Internet (IP), para la comunicación se lo realiza mediante este protocolo [6].

3.3. Redes y protocolo TCP / IP

Se han desarrollado diferentes familias de protocolos para comunicación por red de datos para los sistemas UNIX. El más ampliamente utilizado es el Internet Protocol Suite, comúnmente conocido como TCP / IP.

Es un protocolo DARPA que proporciona transmisión fiable de paquetes de datos sobre redes. El nombre TCP / IP Proviene de dos protocolos importantes de la familia, el Transmission Control Protocol (TCP) y el Internet Protocol (IP). Todos juntos llegan a ser más de 100 protocolos diferentes definidos en este conjunto. Linux tiene integrado en su código el protocolo TCP/IP, actualmente la implementación se lo denomina NET-4 [2].

3.4. Sockets

3.4.1. Definición

Es el objeto básico en toda comunicación a través de Internet, bajo el protocolo TCP. Esta clase proporciona métodos para la entrada/salida a través de streams que hacen la lectura y escritura a través de sockets muy sencilla.

3.4.2. Propiedades inherentes a los sockets

Las propiedades de un socket dependen de las características del protocolo en el que se implementan. El protocolo más utilizado es TCP (Transmission Control Protocol); una alternativa común a éste es UDP (User Datagram Protocol).

Cuando se implementan con el protocolo TCP, los sockets tienen las siguientes propiedades:

- Son orientados a la conexión.

- Se garantiza la transmisión de todos los octetos sin errores ni omisiones.
- Se garantiza que todo octeto llegará a su destino en el mismo orden en que se ha transmitido.

Estas propiedades son muy importantes para garantizar la corrección de los programas que tratan la información.

El protocolo UDP es un protocolo no orientado a la conexión. Sólo se garantiza que si un mensaje llega, llegue bien. En ningún caso se garantiza que llegue o que lleguen todos los mensajes en el mismo orden que se mandaron [7]. Esto lo hace adecuado para el envío de mensajes frecuentes pero no demasiado importantes, como por ejemplo, un streaming de audio.

3.4.3. Modelo de comunicaciones con Sockets en Java

Como modelo de comunicación entre los clientes que son los controladores y la aplicación sockets “SERVER-SIMERT” del servidor, se ha utilizado los sockets de la siguiente manera:

Crear el socket servidor.

Para hacer el servidor en java tenemos la clase `ServerSocket`, al instanciarla usaremos el constructor al que se le pasa un número de servicio (de puerto). Este número de puerto puede ser cualquier entero entre 1 y 65535. Los números de 1 a 1023 están reservados para servicios del sistema (como ftp, mail, www, telnet, etc.). Del 1024 en adelante podemos usarlos a nuestro gusto. Lo único es que no puede haber dos servidores atendiendo al mismo puerto/servicio.

Para el cliente tenemos la clase `Socket`. Basta instanciarla indicándole contra que máquina conectarse y el puerto con el que debe conectarse. Debe ser el mismo que el puerto que está atendiendo el servidor, como se muestra en la (Tabla I).

TABLA I: CÓDIGO PARA CREAR EL SOCKETS DEL SERVIDOR

CREAR EL SOCKET
<pre> /* inicializa el Socket servidor*/ ServerSocket sc = null; try { /* crea socket servidor que escuchara el puerto, * por ejemplo, el 3088*/ sc = new ServerSocket(); /* Vincula una dirección local IP y un puerto de protocolo * al socket*/ sc.bind(new InetSocketAddress(PUERTO)); } catch (IOException ex) { /* mensaje a presentar*/ area.append("**ERROR AL ESCUCHAR EL PUERTO: " + ex.toString()+ "**" + "\n"); } </pre>

Aceptar un cliente.

Una vez creado el servidor, le decimos que empiece a atender conexiones de clientes. Para ello llamamos al método `accept ()`. Este método se queda bloqueado hasta que algún cliente se conecta y nos devuelve un `Socket`, que es la conexión con dicho cliente. Podemos aceptar simultáneamente varios clientes, pero para atenderlos necesitaremos programación multitarea, como se muestra en la (Tabla II).

TABLA II: CÓDIGO PARA ACEPTAR EL CLIENTE

ACEPTA VARIOS CLIENTES
<pre> /* mientras sea verdadera la sentencia anterior se esperara que * los clientes que se conecten al servidor*/ while (true) { /*crea socket cliente*/ cliente = new Socket(); try { /*Se acepta una conexión con un cliente. * Esta llamada se queda bloqueada hasta * que se arranque el cliente*/ cliente = sc.accept(); /*Extraemos la ip del cliente conectado*/ ipclientes = cliente.getInetAddress(); /*Obtener la referencia al socket cliente. * Inicia el socket, ahora está esperando una conexión * por parte del cliente*/ area.append("UN CLIENTE SE HA CONECTADO " + "\n"); area.append("CLIENTE: " + ipclientes + "\n"); } catch (IOException ex) { /*presenta mensaje*/ </pre>

```

        area.append("**ERROR EN LA CONEXION DEL CLIENTE: "
                + ex.toString() + "***" + "\n");
    }
}

```

Obtener los InputStream y/o OutputStream del cliente.

Ahora tenemos la conexión con el cliente, lo único que tenemos que hacer es obtener el OutputStream o InputStream con los métodos `getOutputStream ()` o `getInputStream ()`. La clase `OutputStream` nos sirve para enviarle datos al cliente.

La clase `InputStream` nos sirve para leer datos del cliente.

Estos métodos los ubicamos dentro del `while` como indica la tabla anterior

TABLA III: CÓDIGO PARA OBTENER LA SALIDA Y ENTRADA DEL CLIENTE

OBTENER SALIDA Y ENTRADAS DEL CLIENTE
<pre> try { /*setSoLinger() a true hace que el cierre del socket espere * a que el cliente lea los datos, hasta un máximo de 10 * Segundos de espera. Si no ponemos esto, el socket * se cierra inmediatamente y si el cliente no ha tenido * tiempo de leerlos, los datos se pierden*/ cliente.setSoLinger(true, 10); /*Canales de entrada y salida de datos*/ entrada = new DataInputStream(cliente.getInputStream()); salida = new DataOutputStream(cliente.getOutputStream()); } catch (IOException ex) { /*presentar mensaje*/ area.append("**ERROR EN LA ENTRADA Y SALIDA DE DATOS: " + ex.toString() + "***" + "\n") } </pre>

Crear unos InputStream y/o OutputStream más adecuados a nuestras necesidades.

Los métodos de estas dos clases para leer o escribir datos, únicamente envían bytes. Suele ser habitual construir alguna otra clase de entrada/salida de datos que tenga métodos más adecuados:

- Si queremos enviar o recibir tipos normales (enteros, flotantes, strings) tenemos las clases `DataInputStream` y `DataOutputStream`. Estas clases tienen un constructor que admite un `InputStream` y un `OutputStream` respectivamente.
- Si queremos enviar o recibir clases enteras propias nuestras, tenemos las clases `ObjectInputStream` y `ObjectDataOutputStream`. Al igual que las anteriores, usaremos el constructor que admite un `InputStream` y un `OutputStream`.

Leer y escribir datos del y al cliente.

El envío/lectura de datos normales se hace con clases `DataInput` y `DataOutputStream`, No tienen ningún truco especial, basta usar el método adecuado (`writeln ()`, `writeFloat ()`, `readln ()`, etc.). Para strings usaremos los métodos `writeUTF ()` y `readUTF ()`, que envían/leen las cadenas en formato UTF como se muestra en la (Tabla IV).

TABLA IV: CÓDIGO PARA LA LECTURA DE TRAMAS

LEER LAS TRAMAS QUE LLEGAN DE LOS CLIENTES
<pre> /** método para leer la trama que envían los controladores *(clientes) *@return la trama descriptada recibida por el controlador*/ public String leerTramas() { try { /*variable para almacenar la trama*/ String msj = "", msjn = ""; /*lee la trama que se recibe del controlador*/ msj = entrada.readUTF(); /* descripta la trama recibida por el controlador*/ msjn = decrypt(msj, clave); /*retornamos la trama recibida*/ return msjn; } catch (IOException ex) { /*presenta mensaje*/ area.append("**ERROR AL LEER EL MENSAJE: " + ex.toString() + "**" + "\n"); } return null; } </pre>

Cerrar el socket.

Para cerrar un socket hay que llamar a la función `close ()`. Se cierra el socket cuando el mensaje está en blanco o cuando la trama no es válida como se muestra en la (Tabla V).

TABLA V: CÓDIGO PARA CERRAR EL SOCKET

CIERRE DEL SOCKET
<pre> /*comprobamos si el mensaje del cliente está en blanco o vacío*/ if (msj2 == null) { /*cierra la conexión del cliente*/ cliente.close(); /*presenta el mensaje "la trama no existe" en el área de texto*/ area.append("**LA TRAMA NO EXISTE**" + "\n"); } </pre>

3.5. Incorporación de librerías para la aplicación de red.

3.5.1. Librería Jasypt

Para la encriptación y des encriptación en el envío y la recepción de datos desde la aplicación sockets del servidor hacia el dispositivo móvil Android de los controladores y viceversa se hizo uso de la librería **Jasypt**.

La librería jasypt contiene una clase que permite la encriptación y des encriptación mediante la utilización de una clave privada. Es decir esta clave es creada por los desarrolladores y debe constar tanto en la aplicación cliente como en la aplicación sockets del servidor [8].

Una vez que se ha descargado la librería de la página oficial [9]; se realiza lo siguiente dentro de Netbeans se pulsa clic derecho sobre el proyecto y se selecciona la opción “Propiedades”.

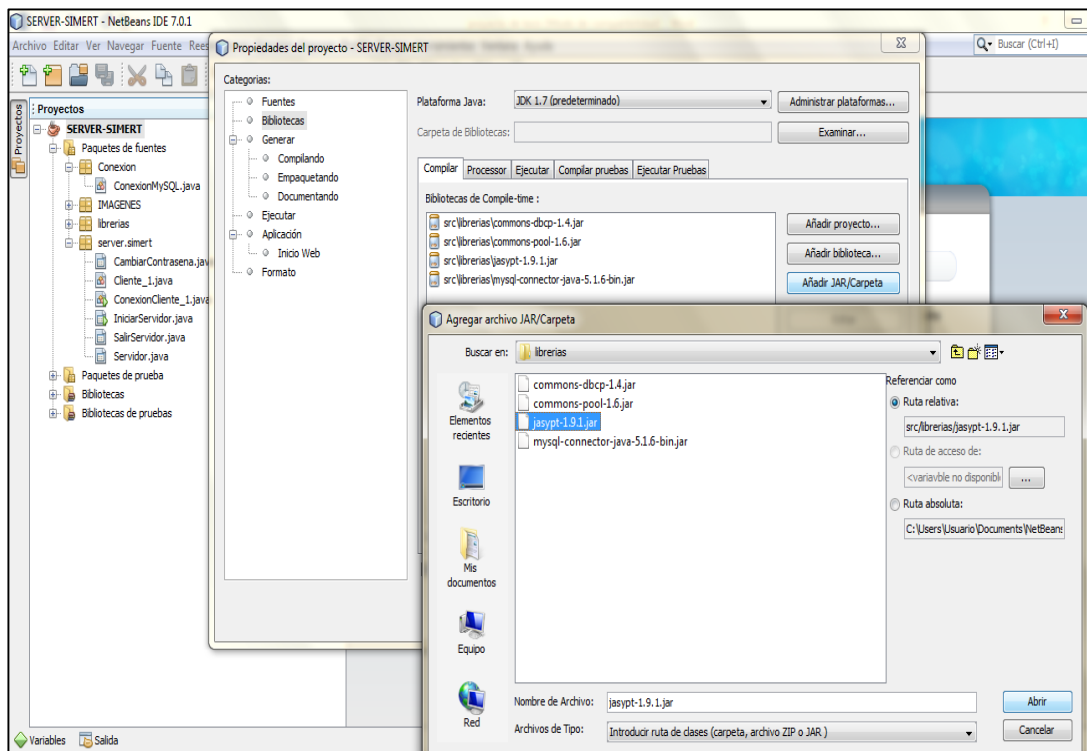


Figura 1: Insertar librerías para la aplicación Sockets denominada "SERVER-SIMERT"

Como se muestra en la (Figura 1) dentro de Propiedades del proyecto se selecciona dentro de Categorías, la opción “Bibliotecas” y damos clic en “Añadir JAR Carpeta”, se busca la dirección en donde se descargó la librería se selecciona y damos clic en “Abrir”. En este caso ya se encuentra instalada.

Luego de realizar lo anteriormente descrito ya se puede hacer uso de la clase de encrypt o decrypt con los respectivos métodos para tal objetivo como se muestra en la (Tabla VI) y en la (Tabla VII).

Método de encriptación

TABLA VI. MÉTODO DE ENCRIPCIÓN

MÉTODO ENCRYPT
<pre>public String encrypt(String trama, String clave) { StandardPBESStringEncryptor s = new StandardPBESStringEncryptor(); s.setPassword(clave); return s.encrypt(trama); }</pre>

Como se muestra en la Tabla VI, el método de encriptación recibe dos parámetros String trama que son los datos del registro de control a encriptarse y String clave que es la cadena que se debe definir tanto en el cliente como en el servidor, es decir la clave privada definida.

Método de desencriptación

TABLA VII: MÉTODO DE DESENCRIPTACIÓN

MÉTODO DECRYPT
<pre>public String decrypt(String trama, String clave) { StandardPBESStringEncryptor s = new StandardPBESStringEncryptor(); s.setPassword(clave); String devuelve = ""; try { devuelve = s.decrypt(trama); } }</pre>

```
    } catch (Exception e) {  
        area.append("**ERROR AL DESENCRIPTAR LA TRAMA: "  
                + e.toString() + "**" + "\n");  
    }  
    return devuelve;  
}
```

Como se muestra en la Tabla VII, el método de desencriptación recibe dos parámetros String trama que son los datos del registro de estacionamiento vehicular a desencriptar y String clave que es la cadena fijada en el cliente y servidor como clave de desencriptación.

3.6. Librería commons-dbcp y commons-pool

Estas librerías se utilizan para hacer un pool de conexiones que es una clase java que tiene abiertas varias conexiones a la base de datos. Cuando alguien necesita una conexión a la base de datos, en vez de abrirla directamente con DriverManager.getConnection (), se la pide al pool usando su método pool.getConnection (). El pool coge una de las conexiones que ya tiene abierta, la marca como que alguien la está usando para no dársela a nadie más y la devuelve. La siguiente llamada a este método pool.getConnection (), buscará una conexión libre para marcarla como ocupada y la devolverá... y así sucesivamente.

Con la librería de apache commons-dbcp podemos implementar un pool de conexiones, esta librería necesita a su vez la librería commons-pool y el driver de nuestra base de datos mysql-connector-java-5.1.6.jar [4]. Una vez que se ha descargado la librería commons-dbcp y commons-pool de la página oficial www.commons.apache.org y el driver de nuestra base de datos de la página oficial www.dev.mysql.com; y se aplica el siguiente método de la (Tabla VIII).

TABLA VIII. MÉTODO POOL DE CONEXIONES

MÉTODO POOL DE CONEXIONES A LA BASE DE DATOS
<pre>public void inicializaDataSource() { BasicDataSource basicDataSource = new BasicDataSource(); basicDataSource.setDriverClassName("com.mysql.jdbc.Driver"); basicDataSource.setUsername("root"); basicDataSource.setPassword("toor1"); basicDataSource.setUrl("jdbc:mysql://127.0.0.1/SIMERT"); dataSource = basicDataSource; }</pre>

Como se muestra en la (Tabla VIII), en el método pool de conexiones tenemos que hacer un *new* de esa clase y pasarle los parámetros adecuados de nuestra conexión con los métodos *set ()* disponibles para ello como:

- Con *setDriverClassName()*: el nombre de la clase que es el driver de nuestra conexión a la base de datos, MySQL en nuestro ejemplo
- Con *setUsername ()*: un nombre de usuario válido para la base de datos (root).
- Con *setPassword ()*: el password de base de datos para dicho usuario (toor1), con *setUrl ()*: la cadena de conexión con la base de datos, en nuestro caso en el mismo ordenador en que va a correr nuestro ejecutable (localhost) y el nombre de la base de datos (SIMERT).

e. Materiales y métodos

En el proceso de desarrollo del presente trabajo de titulación se ha utilizado técnicas de recolección de información, métodos y una adecuada metodología para realizar la conectividad de la red de datos del SIMERT y cumplir los objetivos propuestos, las mismas que se detallan a continuación:

1. MÉTODOS

El presente trabajo de titulación se sustenta en los siguientes métodos:

1.1. Método Deductivo

El método deductivo parte de datos generales verdaderos para llegar a hechos particulares, siendo así en el presente proyecto se ha utilizado este método para analizar la situación actual en cuanto a la comunicación y servicio del Sistema Municipal De Estacionamiento Rotativo Tarifado (SIMERT) para determinar los problemas generales existentes, con ello llegar al problema específico o particular y plantear la problemática de la presente tesis, organizando la información que será utilizada para el desarrollo del proyecto.

1.2. Método Inductivo

Con el método inductivo se obtiene conclusiones generales de hechos particulares, siendo así se ha utilizado este método para que a partir del análisis de los problemas encontrados en la comunicación y servicio del Sistema Municipal De Estacionamiento Rotativo Tarifado (SIMERT) se pueda determinar cuál es la situación general que origina estos inconvenientes y enfocarse directamente en resolver dicho problema.

1.3. Método Científico Experimental

La aplicación más completa de la investigación científica porque permite establecer con toda claridad el principio de relación causa – efecto. Consiste en provocar voluntariamente una situación que se quiere estudiar, este método ha permitido realizar las tareas planificadas.

2. TÉCNICAS

Entre las técnicas que se han utilizado para obtener información de cada uno de los entes del Sistema Municipal De Estacionamiento Rotativo Tarifado (SIMERT) en el presente trabajo de titulación son las siguientes:

2.1. Observación

Es una técnica que consiste en observar atentamente el fenómeno, hecho, tomar información y registrarla para su posterior análisis.

La observación es un elemento fundamental de todo proceso investigativo; en ella se apoya el investigador para obtener el mayor número de datos, entre las clases de observación tenemos:

- **Observación Directa:**

Es aquella en la cual el investigador se pone en contacto personalmente con el hecho o fenómeno que trata de investigar.

- **Observación Indirecta:** cuando el investigador entra en el conocimiento del hecho o fenómeno observado a través de las observaciones realizadas anteriormente por otra persona, (Libros, revistas, etc.)

Dentro del proyecto se ha utilizado la observación en los siguientes casos:

- Visualizar los procesos que se realizan con los registros de estacionamiento vehicular. Para ello se procedió a identificar las personas que intervienen en el caso de estudio como son: Coordinador, Secretaria y Personal Administrativo, seguidamente se observó las actividades que realizan cada una de estas personas, como procesos de ingreso, reportes, cálculos de tasas, búsqueda de la información, vigilancia y control del personal operativo, sin utilizar herramientas informáticas.
- La observación del fenómeno estudiado facilitó la obtención de datos, los mismos que han sido utilizados en la fase de análisis de requerimientos.

2.2. Entrevista

Para el proyecto se ha realizado la entrevista a las personas involucradas en la problemática; como a la Ingeniera encargada de la Técnica de la Unidad de Tránsito (Ver Anexo 5) con el fin de identificar cualquier necesidad de Hardware y Software actuales y futuros, los mismos que han servido para sustentar el presente trabajo de titulación.

2.3. Encuesta

Se ha realizado la encuesta a las personas involucradas en la problemática; como son: a la Coordinadora del SIMERT (Ver Anexo 1), a la Secretaria (Ver Anexo 2), a la Ingeniera encargada de la Técnica de la Unidad de tránsito (Ver Anexo 3) y por último a los Controladores (Ver Anexo 4) obteniendo información como el perfil técnico de cada uno de ellos, operaciones y procesos que se llevan a cabo y el listado de los requerimientos.

Mediante esta técnica también se ha podido detectar el grado de satisfacción de las personas que laboran dentro del SIMERT como la Ingeniera encargada de la Técnica de la Unidad de tránsito, con respecto a la aplicación desarrollada e identificar si cumple o no con cada uno de los requerimientos establecidos al inicio.

3. METODOLOGÍA PARA LA CONECTIVIDAD DE LA RED DE DATOS DEL SIMERT.

Para realizar la conectividad de la red de datos del SIMERT se ha desarrollado una metodología para realizar la conectividad de la red de datos del SIMERT donde se ha planteado las siguientes fases, que permiten cumplir con cada una de las actividades de manera organizada y determinar principalmente los diferentes problemas existentes para realizar consecuentemente una solución apropiada:

- Fase 1: Análisis de requerimientos
- Fase 2: Diseño de la infraestructura para la conectividad de la red de datos.
- Fase 3: Implementación.
- Fase 4: Medición de Resultados

3.1. Fase 1: Análisis de requerimientos

El primer paso para llevar a cabo la conectividad de la red de datos es realizar una recopilación de información confiable del Sistema Municipal de Estacionamiento Rotativo Tarifado (SIMERT).

Para obtener toda esta información se hizo uso de la técnicas de encuestas, observación y entrevistas, donde se ha realizado una entrevista al Personal Administrativo del SITU y del SIMERT para obtener información de la Institución como la estructura organizacional, estado actual, sus políticas de operación y sus procedimientos de

administración (Ver sección Resultados, Pág. 44, apartado 1.1. Recopilación de información confiable de la estructura organizacional del Sistema Integrado Municipal de Estacionamiento Rotativo Tarifado (SIMERT)) también se aplicó una encuesta al personal que labora en el SIMERT como: a la Coordinadora del SIMERT (Ver Anexo 1), a la Secretaria (Ver Anexo 2), a la Ingeniera encargada de la Técnica de la Unidad de tránsito (Ver Anexo 3) y por último a los Controladores (Ver Anexo 4) obteniendo información como el perfil técnico de cada uno de ellos, operaciones y procesos que realizan (Ver sección Resultados, Pág. 57, 1.2. Recopilación de información sobre el perfil técnico y actividades que realiza el personal del SIMERT). Se ha utilizado la técnica de observación en las instalaciones del SIMERT y del SITU en conjunto con la aplicación de una entrevista realizada a la Ingeniera encargada de la Técnica de la Unidad de Tránsito (Ver Anexo 5) para identificar cualquier necesidad de Hardware y Software actuales y futuros (Ver sección Resultados, Pág. 59, 1.3. Recopilar información sobre los recursos que actualmente posee el SIMERT), y por último se aplicó una entrevista a los diferentes proveedores que brindan el servicio de la tecnología GPRS obteniendo las diferentes cotizaciones de: CLARO (Ver Anexo 11), CNT (Ver Anexo 12) y MOVISTAR (Ver Anexo 13) para la contratación del mismo.

A partir de la información recolectada se ha determinado los requerimientos funcionales y no funcionales, así como también se ha elaborado un análisis técnico, análisis de seguridad, análisis de las aplicaciones y herramientas para el desarrollo, concluyendo con la definición de la topología a utilizar (Ver sección Resultados, Pág. 54, apartado 1.8. Definir la topología que se utilizará para la conectividad de la red de datos del SIMERT).

Esta fase es uno de los puntos claves en el desarrollo del diseño, identificando posibles problemas desde el inicio y uno de los más difíciles de obtener.

3.2. Fase 2: Diseño de la infraestructura para la conectividad de la red de datos.

Una vez claros los requerimientos para la conectividad de la red de datos del SIMERT, se elaboró los prototipos de las interfaces de la aplicación (Ver sección Resultados, Pág. 57, apartado 2.1. Prototipado Inicial) los cuales fueron presentados a la Ingeniera encargada de la Técnica de la Unidad de Tránsito como un punto de partida, quién emitió las sugerencias pertinentes.

Luego se ha procedido a modelar el sistema de información en términos de funciones y datos desde la perspectiva del usuario. Para ello se ha realizado los siguientes modelos:

- Modelo de la base de datos para el almacenamiento de la información de los registros de estacionamiento vehicular del SIMERT en el servidor: Aquí se identifican las entidades que componen el sistema de información, sus atributos y relaciones entre éstas (Ver Anexo 6).
- Modelo de casos de uso (Ver sección Resultados, Pág.61, apartado 2.2. Diseño del modelo de casos de uso), en correspondencia a los requerimientos establecidos. Para diseñar este modelo inicialmente se realizó las modificaciones en las interfaces de la aplicación para continuar con la descripción de los casos de uso, identificando la secuencia de actividades con el respectivo curso normal de eventos y cursos alternos (Ver sección Resultados, Pág. 61, apartado 2.3. Descripción de los casos de uso).
- Modelo de procesos (Ver sección Resultados, Pág. 74, apartado 2.4. Diseño del Modelo de procesos o robustez)
- Modelo de secuencia: en los cuales se ha mostrado las funciones visibles por el usuario, determinando la jerarquía de menús y submenús que el sistema ofrecerá al usuario administrador del servidor (Ver sección Resultados, Pag.77, apartado 2.5. Diseño del Modelo de secuencia).

También se ha realizado el diseño de la arquitectura cliente – servidor para la conectividad de la red de datos del SIMERT y la descripción de su simbología (Ver sección Resultados, Pág. 80, apartado 2.6. Diseño Arquitectura Cliente-Servidor), definiendo una topología de red general que satisfaga los requerimientos de usuario. Donde los principales dispositivos a utilizar son los switch para mejorar el rendimiento en la red, los routers para crear segmentos únicos de red que permiten la comunicación entre segmentos basados en direccionamiento IP y finalmente como canal de comunicación entre los clientes y el servidor para el envío y recepción de la información de los registros de estacionamiento vehicular del SIMERT, se ha hecho uso de la tecnología GPRS bajo el protocolo TCP/IP, direccionamiento IP y la clase ServerSockets dentro de la aplicación.

3.3. Fase 3: Implementación

Durante la implementación se ha utilizado especialmente la fase del análisis y la fase del diseño.

En esta fase se aplica la implementación de todos los diseños para establecer la conectividad de la red de datos del SIMERT. Donde se ha empezado por la creación e instalación del servidor SIMERT (Ver sección Resultados, Pág. 84, apartado 3.1. Creación del servidor SIMERT), luego se ha realizado las debidas configuraciones par el servidor (Ver sección Resultados, Pág.87, apartado 3.1.6. Configuraciones para la creación del servidor SIMERT).

Una vez configurado y creado el servidor se ha generado todo el código fuente para construir la aplicación “SERVER-SIMERT” (Ver sección Resultados, Pág. 97, apartado 3.2. Codificación o generación de código Java para la aplicación sockets denominada “SERVER-SIMER” para el servidor), bajo la arquitectura MVC (modelo vista controlador) (Ver sección Resultados, Pág. 101, apartado 3.2.3. Arquitectura de la aplicación sockets “SERVER-SIMERT”).

Finalmente se ha realizado la simulación en el packet tracer de la arquitectura cliente/servidor con la debida configuración de los equipos (Ver sección Resultados, Pág. 105, apartado 3.3.Simulación en el Packet Tracer de la arquitectura cliente/servidor para la conectividad de la red de datos del SIMERT)

3.4. Fase 4: Medición de resultados.

En esta última fase es necesario realizar una evaluación del trabajo realizado en el SIMERT, donde se ha evaluado la utilización de los servicios y por defecto se estará evaluando la conformidad, la aceptación por parte de los usuarios hacia la nueva implementación, realizando las respectivas pruebas de funcionalidad y verificar que la infraestructura HW/SW permita un correcto fluido de datos entre todas las áreas del SIMERT. Se ha realizado las pruebas de carga al servidor (Ver sección Resultados, Pág. 108, apartado 4.1.1. Pruebas de carga al servidor), se ha realizado las pruebas a la aplicación sockets “SERVER-SIMERT” (Ver sección Resultados, Pág. 111, apartado 4.2. Pruebas de la aplicación sockets “SERVER-SIMERT”).

f. Resultados

El uso de la metodología creada para la conectividad de la red de datos del SIMERT, permitió llevar a cabo nuestro proyecto fin de carrera por fases, abarcando todo el ciclo de vida del proyecto, descritas cada una a continuación.

1. FASE 1: ANÁLISIS DE REQUERIMIENTOS

En esta fase inicial de análisis se elaboró un documento correspondiente a la Especificación de Requisitos según el formato que provee el estándar IEEE 830 (ver Anexo 7), para determinar las funciones con las que cumplirá la aplicación.

La fase de análisis de requerimientos se ha enfocado en la obtención de los objetos que están directamente e indirectamente relacionados con la problemática. A continuación detallamos las actividades realizadas en esta etapa:

1.1. Recopilación de información confiable de la estructura organizacional del Sistema Integrado Municipal de Estacionamiento Rotativo Tarifado (SIMERT).

Para la recopilación de esta información se ha realizado una entrevista al personal administrativo del Sistema Integrado de Transporte Urbano (SITU) y del Sistema Integrado Municipal de Estacionamiento Rotativo Tarifado (SIMERT) para obtener información de la institución como la estructura organizacional, estado actual, sus políticas de operación y sus procedimientos de administración.

El Sistema Integrado de Transporte Urbano –SITU- para la ciudad de Loja, implementado por el municipio, dentro de las competencias de descentralización que le fueron transferidas por la Dirección Nacional de Tránsito, trata de ordenar el tránsito, apoyándose en los artículos de la Ordenanza de la Unidad Municipal de Tránsito, Transporte Terrestre y Seguridad Vial, (UMTTTSV); obligando a las personas a utilizar este servicio. Esta institución cuenta con un orgánico estructural donde se encuentra el Sistema Integrado Municipal de Estacionamiento Rotativo Tarifado (SIMERT)

Actualmente el Sistema Integrado Municipal de Estacionamiento Rotativo Tarifado (SIMERT) funciona como un área independientemente de la Unidad de Tránsito. Se

encarga del control y vigilancia del estacionamiento vehicular y del uso correcto de las tarjetas para el servicio de estacionamiento, apoyándose también en los artículos de la Ordenanza de la Unidad Municipal de Tránsito, Transporte Terrestre y Seguridad Vial, (UMTTTSV). Esta institución no cuenta con un orgánico estructural y para asegurar su funcionamiento se propone el siguiente esquema con sus funciones como se muestra en la (figura 2).

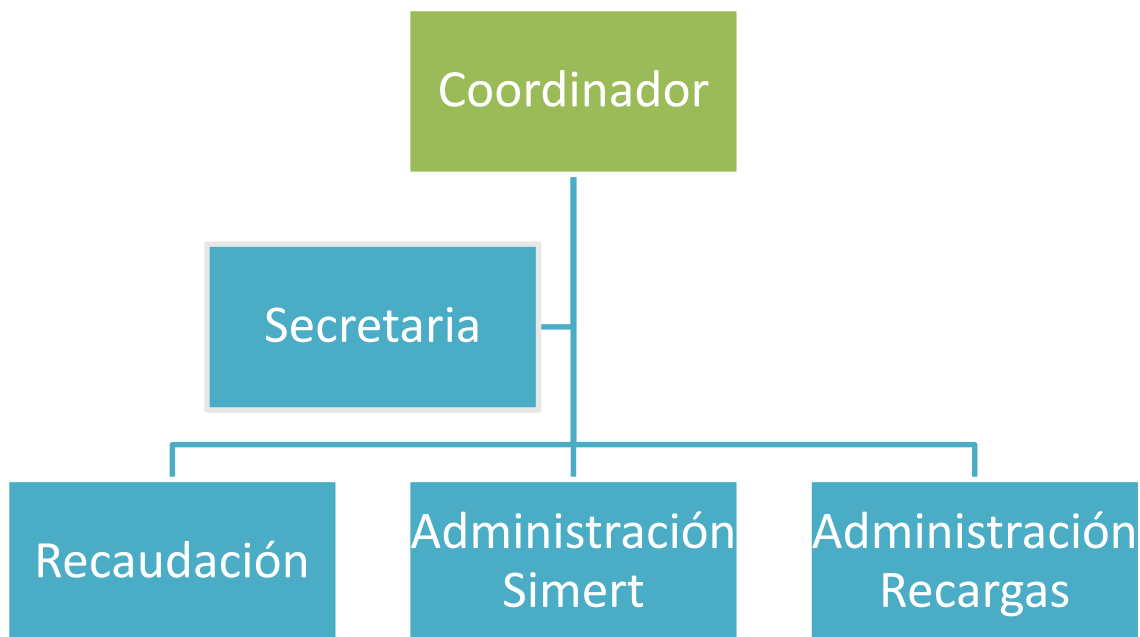


Figura 2: Organigrama Estructural para el SIMERT.

COORDINADOR

Tendrá todos los privilegios en el sistema:

- Administración de usuarios.
- Modificación de multas en casos necesarios.
- Configuración del sistema.
- Visualizar a los controladores a través de la página web.
- Reportar los ingresos y egresos del SIMERT diaria, semanal, mensual y anualmente.
- Reportar las tasas de rotación, respeto y ocupación.

SECRETARIA

- Atender las sugerencias de los usuarios y empleados del SIMERT.
- Brindar información a los usuarios del SIMERT.
- Imprimir reportes de los procesos generados por el SIMERT.

RECAUDACIÓN

Donde se realizará el cobro de multas y venta de tarjetas

• COBRO DE MULTAS:

- Gestión de multas cobradas y las no cobradas.
- Visualizar el listado de los clientes que han infringido la ordenanza.
- Reporte de las multas recaudadas diariamente.

• VENTA DE TARJETAS:

- Administrar clientes de tarjetas SIMERT y recargas web.
- Venta de tarjetas y recargas SIMERT.
- Reporte de las ventas realizadas diariamente.
- Los usuarios adquirirán la tarjeta SIMERT en algún punto autorizado de venta.
- Marcar el tiempo de estacionamiento en la tarjeta que ocupará el vehículo cada que necesite un espacio dentro de una zona permitida.

ADMINISTRACIÓN DEL SIMERT

- Administrar el personal operativo.
- Administrar las zonas de estacionamiento.
- Administrar horarios y lugares de trabajo.
- Visualizar y controlar al personal operativo mediante una página web.

PERSONAL OPERATIVO

- Registrará los vehículos que harán uso del estacionamiento.
- Verificar que las tarjetas prepago no sean falsas y que no hayan sido adulteradas.
- Verificar que los códigos de los adhesivos sean los correctos.
- Controlar el tiempo de estacionamiento automáticamente.
- Emitirá las multas calculadas por el sistema.

ADMINISTRACIÓN DE RECARGAS WEB

- Registrarse en el Sistema y abonar una cantidad de dinero.
- Activará el servicio a través de un teléfono inteligente

- Podrá conocer los espacios de estacionamiento disponibles en cada zona.
- Desactivará el servicio de la misma manera que los activó.
- Conocerá el total de tiempo estacionado y su equivalencia en dólares para ser descontado.
- Conocerá el saldo de la recarga electrónica.

1.2. Recopilación de información sobre el perfil técnico y actividades que realiza el personal del SIMERT.

En esta actividad se realizaron varias encuestas al personal que labora en el SIMERT como: a la Coordinadora (Ver Anexo 1), a la Secretaria (Ver Anexo 2), a la Ingeniera encargada de la Técnica de la Unidad de Tránsito (Ver Anexo 3), por último a los Controladores (Ver Anexo 4) quienes establecerán comunicación con el servidor, con la finalidad de conocer las operaciones y procesos que realizan dentro de la institución y el perfil técnico de cada uno, es decir la capacidad para la utilización de equipos informáticos y aplicaciones de software.

Luego de haber realizado el análisis de las encuestas, se pudo conocer las tareas que se realiza en el SIMERT. Se constató que todas las actividades se las lleva en hojas manualmente y son las siguientes:

Registro de estacionamiento de vehículos

Esta actividad la realiza el personal operativo que son los controladores quienes ingresan los datos de cada vehículo estacionado en hojas impresas con un formato establecido por el SIMERT denominadas “Ficha de control” (Ver Anexo 8). Donde los datos a ingresar son: nombre del controlador, fecha, placa del vehículo, hora de llegada, tiempo de estacionamiento, hora de control, número de tarjeta, hora de inmovilización, hora de orden de inmovilización, número de citación, dirección y observación, toda esta información se las ingresa manualmente y se las archiva en una carpeta; al finalizar el trabajo estas hojas se las entregan al coordinador del SIMERT.

Registro de asignación de zonas a los controladores.

Esta actividad la realiza el coordinador del SIMERT quien se encarga de asignar diariamente las zonas de trabajo a los controladores con sus respectivos horarios y calles, los datos se ingresan en hojas impresas con el formato establecido por el

SIMERT y se archivan en carpetas, también se encarga de controlar al personal operativo que cumpla con su trabajo en la zona asignada para ello se moviliza al lugar.

Registro de tasas de rotación, respeto y ocupación.

Esta actividad la realiza el Coordinador y la Secretaria del SIMERT donde se calcula manualmente lo siguiente: la tasa de ocupación como a la media diaria del cociente entre la cantidad total de horas de estacionamiento (ocupadas) relevadas y la cantidad total de lugares de estacionamiento, en la zona que se considere. La tasa de rotación como la ocupación media en el día, provocada únicamente por los vehículos que estacionan menos de un determinado tiempo. Y la tasa de respeto que se define como el porcentaje de horas de estacionamiento (pagan la tarifa correspondiente), con respecto al total de las horas de estacionamientos ocupadas por vehículos estacionados. Estos procesos los realizan en hojas de Excel y en hojas impresas con el formato establecido por el SIMERT y se las archiva en carpetas.

Soporte Técnico de los equipos y mantenimiento de la red del Sistema Municipal de Estacionamiento Rotativo Tarifado (SIMERT).

Esta actividad es realizada por la Ingeniera de Telecomunicaciones encargada de la Técnica de la Unidad de Tránsito, quién se encarga del soporte técnico, administración de la red de la UMTTTSV, monitoreo de la Unidad de transporte y taxi ejecutivo, asesoría en temas tecnológicos y emisión de reportes e informes técnicos.

En cuanto al perfil técnico de las personas involucradas como los Controladores, la Secretaria y Coordinadora del SIMERT, no poseen la debida capacitación para la manipulación de equipos informáticos con versiones actuales.

Como solución a la problemática de la información recopilada anteriormente es factible realizar la “Tecnificación del Sistema Municipal de Estacionamiento Rotativo Tarifado” que consta de cuatro módulos desarrollando diferentes aplicaciones como: el módulo de administración para el sistema de la información a través del desarrollo de una aplicación web que abarca la parte administrativa del SIMERT, módulo software para el control vehicular y geolocalización del personal operativo a través del desarrollo de una aplicación móvil Android que utilizarán los controladores para realizar los procesos de control vehicular y emisión de multas de este Sistema, módulo de servicios en aplicaciones móviles a través del desarrollo de una aplicación de recargas web

multiplataforma que abarca la parte administrativa de recargas y una aplicación móvil la cual está destinada a las personas que hacen uso del servicio de estacionamiento vehicular y finalmente el módulo conectividad de la red de datos del SIMERT utilizando la tecnología GPRS a través de la implementación de un servidor y desarrollo de una aplicación sockets denominada “SERVER-SIMERT” para el servidor.

Este proyecto se centra en realizar la conectividad de la red de datos del SIMERT utilizando la tecnología GPRS a través de la implementación de un servidor y desarrollo de una aplicación sockets para el mismo denominado “SERVER-SIMERT”, garantizando la seguridad en la comunicación durante el envío y recepción de la información de los registros de estacionamiento vehicular del SIMERT; con el uso de la tecnología GPRS bajo el protocolo TCP/IP, direccionamiento IP y utilizando la clase ServerSockets dentro de la aplicación.

Además se realizará las capacitaciones necesarias dirigida a la Secretaria, Coordinadora y Controladores del SIMERT para la manipulación de equipos informáticos incluyendo las aplicaciones de software con el fin de lograr éxito en la comunicación bajo la arquitectura cliente – servidor.

1.3. Recopilar información sobre los recursos que actualmente posee el SIMERT.

En esta actividad se ha utilizado la técnica de observación en las instalaciones del SIMERT y del SITU en conjunto con la aplicación de una entrevista realizada a la Ingeniera encargada de la Técnica de la Unidad de Tránsito (Ver Anexo 5) para identificar cualquier necesidad de Hardware y Software actuales y futuros, con la finalidad de conocer los recursos tecnológicos con los que cuenta actualmente la Institución.

Luego de haber realizado el análisis de la observación y encuesta, se ha llegado a la conclusión que el SITU no cuenta con un Centro de Cómputo porque está en proceso de equipamiento, solo tiene una sala de monitoreo donde utiliza un rack cerrado de pared en el cual se encuentra ubicado 4 servidores, el servidor web en el cual se alojan las páginas para el sistema de rastreo de buses, el servidor proxy, el servidor de base de datos para el rastreo de buses, el servidor backup para el rastreo de buses y 3 pc's para el monitoreo; posee políticas de seguridad y hace uso del firewall en el proxy para

mantener la confidencialidad de la información. Utiliza la red GPRS del proveedor de Movistar para el rastreo de bus.

En cuanto al SIMERT no cuenta con un Centro de Cómputo por la falta de recursos e infraestructura ya que no cuenta con un lugar estable, por lo tanto todas las actividades que realiza el SIMERT se las hace manualmente sin control alguno.

Como solución a la problemática que presenta el SIMERT se ha optimizado los recursos que posee actualmente dicha institución, realizando el presente proyecto denominado “Tecnificación del Sistema Municipal de Estacionamiento Rotativo Tarifado, módulo conectividad de la red de datos del SIMERT utilizando la red GPRS” a través de la implementación de un servidor y creación de una aplicación sockets para el mismo, el cual será ubicado bajo las instalaciones del SITU en la sala de monitoreo, para la seguridad y control de la información de los registros de estacionamiento vehicular. También se ha recomendado la creación de un Centro de Cómputo para el SIMERT, en un lugar estable para el buen funcionamiento del mismo.

1.4. Recopilar información acerca de los proveedores que brinden el servicio de GPRS.

En esta actividad se realizaron visitas a los proveedores de telefonía móvil, donde se obtuvieron diferentes cotizaciones de CLARO (Ver Anexo 11), de CNT (Ver Anexo 12) y MOVISTAR (Ver Anexo 13) para la adquisición de la tecnología GPRS.

A continuación se ha elaborado una tabla comparativa de los diferentes proveedores del servicio GPRS con el fin de contratar uno de ellos.

TABLA IX: PROVEEDORES QUE BRINDAN EL SERVICIO GPRS

SERVICIO GPRS (GLOBAL PACKET RADIO SERVICE)					
Proveedores	Velocidad	Tarifa Básica (P/Simm)	MEGAS INCLUIDAS	Tarifa Básica MB	Tarifa Básica Final MB MENSUAL
CNT	Buena	Gratuito	1000 MB	\$ 19.99	\$ 19.99
CLARO	Excelente	Gratuito	1000 MB	\$ 19 +imp (HSPA+)	\$ 21.28

MOVISTAR	Buena	Gratuito	1000 MB	\$ 20 +imp	\$ 22.40
-----------------	--------------	----------	---------	---------------	----------

De acuerdo al análisis realizado según la tabla comparativa de los proveedores que brindan el servicio GPRS.

Por el presupuesto, velocidad y puesto a prueba, se recomienda al proveedor de claro para la adquisición del servicio GPRS.

1.5. Análisis técnico.

Como parte de la solución propuesta por este proyecto de Tesis es realizar la conectividad de la red de datos del SIMERT utilizando la tecnología GPRS, por lo tanto se ha implementado un servidor junto con el desarrollo de una aplicación sockets denominada “SERVER-SIMERT” para el servidor.

A continuación se muestra una tabla comparativa de las tecnologías más importantes para desarrollar aplicaciones sockets.

TABLA X: COMPRACIÓN DE LOS LENGUAJES DE PROGRAMACIÓN

	C	C++	Java	Python
Orientado a Objetos	No	Si	Si	Si
Plataforma	Ninguna	Ninguna	Java Virtual Machine Netbeans Ide 7.0.4	Ninguna
Nivel de Conocimiento	Básico	Intermedio	Avanzado	Básico
Seguridad de Tipos	No	No	Si	Si

Análisis Técnico de las herramientas utilizadas en la aplicación “SERVER-SIMERT”

De acuerdo al análisis realizado anteriormente en la (Tabla X), para construir la aplicación sockets para el servidor denominada “SERVER-SIMERT”; por conocimiento en la utilización de librerías y código se utilizará el lenguaje de programación Java en su versión jdk 1.8.0., bajo la plataforma virtual Netbeans IDE 8.0 por la facilidad de uso

en los componentes para el desarrollo de la interfaz gráfica. Utilizando el conector para Mysql -java 5.1.6 -bin.jar, las librerías Commons-pool-1.6 y Commons-logging-1.4 para el pool de conexiones, la librería Jasypt-1.9.1 para la encriptación y des encriptación de datos y el Enterprise Architect v 8.0 para la modelación del sistema.

La aplicación sockets permitirá establecer la conexión entre el servidor y los clientes que serán los controladores, visualizando el envío, recepción y gestión de la información de los registros de estacionamiento vehicular a través de las tramas.

A continuación se muestra una tabla comparativa de las plataformas para los servidores que cumpla con las funciones de envío de correos, almacenamiento de base de datos y de aplicaciones.

TABLA XI: COMPARACIÓN DE LOS S.O. PARA SERVIDORES

PLATAFORMA	SISTEMA OPERATIVO	LICENCIA	COSTO	SOPORTE TECNICO	SISTEMA DE INFRAESTRUCTURA DE INTERNET
Microsoft Windows	Windows Server 2008	SI	SI	SI	Wamp
Linux	Ubuntu Server 10.04	NO	NO	SI	Lamp
Linux	Centos	NO	NO	SI	Lamp

Análisis de las plataformas para el servidor.

Según el análisis realizado acerca de las diferentes plataformas para los servidores como se muestra en la (Tabla XI), por sus características, conocimiento y usabilidad se implementará el servidor Ubuntu Sever Lucid Links 10.04 que es una versión de Linux con el sistema de infraestructura LAMP, el mismo que será ubicado e instalado en la sala de monitoreo del SITU. El mismo que se ha configurado correctamente para que funcione como servidor web, servidor de correo, servidor de base de datos y servidor de aplicaciones, utilizando la base de datos Mysql v. 5.5.2., el Phpmyadmin v. 5.1.72 para la administración de la base de datos, Postfix v. 2.7 para el servicio de correos y el apache 2.4.2. Potenciando la ejecución de los procesos que realiza diariamente el personal operativo y administrativo del SIMERT con los registros de estacionamiento vehicular, lo que permite una adecuada organización y fiabilidad de la información.

1.6. Análisis de seguridad.

Para realizar la conectividad de la red de datos del SIMERT, se debe realizar un análisis de seguridad tanto para la implementación del servidor como para la aplicación sockets del servidor y del cliente.

Tanto para la aplicación cliente como para el servidor se realizará a través de tramas, el envío y recepción de información de los registros de estacionamiento vehicular, las mismas que serán encriptadas y des encriptadas a través de una clave tanto para el cliente como para el servidor haciendo uso de la librería Jasypt de Java.

La aplicación Cliente necesita claves de acceso para poder conectarse al servidor. Si los controladores desean conectarse con el servidor enviarán la trama correspondiente para su identificación, donde la aplicación del servidor validará el usuario y contraseña recibida y luego enviará la trama de respuesta correspondiente.

Se utilizará la base de datos Mysql para el almacenamiento de la información en el servidor, utilizando un pool de conexiones a través de las librerías commons-pool y commons-dbcp de java.

Para la comunicación entre los clientes y el servidor, se utilizará un protocolo seguro de comunicaciones a través de la red, el protocolo TCP/IP, el GPRS, el direccionamiento IP y la clase ServerSockets dentro de la aplicación.

El servidor se ubicó en la sala de monitoreo del SITU, el cual posee políticas de seguridad para que solo personas autorizadas tengan acceso al mismo. Este servidor también se encuentra configurado correctamente, es decir, se configuró solo los servicios y protocolos a usarse, ya que un servidor mal configurado es un potencial riesgo de caída del sistema. Además para garantizar la seguridad de la información e integridad del servidor, este se ubico detrás de un servidor proxy que utiliza firewall.

También el software fue correctamente instalado, como el Sistema Operativo y la aplicación sockets del servidor, es decir que no tenga ningún hueco de seguridad, los mismos que pueden ser corregidos por los parches de actualización de las aplicaciones.

1.7. Analizar las aplicaciones y Herramientas de Desarrollo.

De acuerdo al análisis técnico realizado anteriormente, para realizar la conectividad de la red de datos del SIMERT, se requiere de un servidor y el desarrollo de una aplicación sockets para el servidor.

Donde el servidor debe ser escalable, confiable y tener un buen desempeño, para que funcione como servidor web, servidor de correo, servidor de base de datos y servidor de aplicaciones. La plataforma bajo la cual debe correr es LINUX, ya que se necesita que ésta no tenga costo alguno, por lo tanto se utilizará uno de las distribuciones de Debian que será Ubuntu server 10.04 como sistema operativo para el servidor con el sistema de infraestructura LAMP, para la base de datos el Mysql v. 5.5.2. Y para su administración Phpmyadmin v. 5.1.72, Postfix v. 2.7 para el servicio de correos y apache 2.4.2.

También se definen las herramientas que serán utilizadas para el desarrollo de la aplicación sockets del servidor denominada “SERVER-SOCKETS”, donde necesitamos una herramienta de desarrollo como el lenguaje de programación Java el cual permite la utilización de librerías para la creación de aplicaciones sockets, bajo la plataforma de Netbeans IDE 7.0.1 porque es un producto libre y gratuito sin restricciones de uso, además está hecho principalmente para el lenguaje de programación Java e incluye un número importante de módulos para extenderlo. Se ha utilizado la librería -java 5.1.6 -bin.jar como conector para la base de datos, las librerías Commons-pool-1.6 y Commons-dbcp-1.4 para realizar un pool de conexiones con la base de datos, la librería Jasypt-1.9.1 para la encriptación y des encriptación de datos y el Enterprise Architect v 8.0 para la modelación del sistema.

Para la comunicación entre los clientes y el servidor, se utilizará un protocolo seguro de comunicaciones a través de la red, el protocolo TCP/IP, direccionamiento IP con la tecnología GPRS y la clase ServerSockets dentro de la aplicación.

1.8. Definir la topología que se utilizará para la conectividad de la red de datos del SIMERT.

Se ha utilizado la topología estrella para realizar la conectividad de la red de datos del SIMERT utilizando la red GPRS dentro de la arquitectura cliente-servidor (Ver sección

Resultados, Pág.80, apartado 2.6. Diseño de la Arquitectura cliente-servidor para la conectividad de la red de datos del SIMERT), porque esta conectividad abarca una red LAN (Local Área Network) y este tipo de topología son las más populares y empleadas para esta red. La topología estrella es más tolerante, esto quiere decir que si una computadora se desconecta o si se le rompe el cable solo esa computadora es afectada y el resto de la red mantiene su comunicación normalmente, también es fácil de reconfigurar, añadir o remover una computadora es tan simple como conectar o desconectar el cable.

1.9. Análisis de los requerimientos funcionales y no funcionales.

Al conocer la situación actual del SIMERT, se determinó los requerimientos funcionales y no funcionales de la aplicación, para lo cual se apoya en las actividades realizadas anteriormente. Estos son anunciados a continuación:

1.9.1. Requerimientos Funcionales

Los requerimientos funcionales con las que debe cumplir el software SERVER-SIMERT, a continuación se nombran en la tabla XII.

TABLA XII: REQUERIMIENTOS FUNCIONALES

REFERENCIA	FUNCIÓN	CATEGORÍA
RF-001	Autenticación del administrador del servidor.	Evidente
RF-002	Cambiar el nombre de usuario y contraseña del Administrador del servidor.	Evidente
RF-003	Escritura y lectura de ficheros	Oculto
RF-004	Visualización del envío y recepción de tramas en tiempos cortos.	Evidente
RF-005	Validación de las tramas	Oculto.
RF-006	Autenticación del Personal Operativo.	Oculto.
RF-007	Almacenamiento de la información de las tramas.	Oculto.
RF-008	Modificación de la información de las tramas.	Oculto.

RF-009	Eliminación de la información de las tramas.	Ocultas.
RF-010	Consultar información de las tramas	Ocultas.
RF-011	Encriptación y des encriptación cada una de las tramas	Ocultas.
RF-012	Establecer la conexión cliente-servidor	Ocultas.

1.9.2. Requerimientos No Funcionales

Los requisitos no funcionales para el software a desarrollarse se describen en la (Tabla XIII).

TABLA XIII: REQUERIMIENTOS NO FUNCIONALES

Referencia	Descripción	Categoría
RNF-001	El sistema funcionará en diferentes plataformas de los Sistemas Operativos Linux, Windows	Evidente
RNF-002	La aplicación será desarrollada en el lenguaje de programación JAVA, JDK 1.8.0, plataforma virtual NETBEANS IDE 8.0	Evidente
RNF-003	El sistema tendrá una interfaz gráfica amigable, donde emitirá cuadros de diálogo cuando se produzca un error y tendrá una navegación fácil desde el mouse y teclado.	Evidente
RNF-006	Mostrar etiquetas claras para campos y botones.	Evidente
RNF-007	No permitir el ingreso de datos erróneos.	Oculto

2. FASE 2: DISEÑO DE LA INFRAESTRUCTURA PARA LA CONECTIVIDAD DE LA RED DE DATOS.

Una vez claros los requerimientos de la red de datos para realizar el diseño se realizara las siguientes actividades:

2.1. Prototipado Inicial.

Para el prototipado de la aplicación “SERVER-SIMERT”, se utilizó la herramienta de software libre Java bajo la plataforma de Netbeans, que permitió diseñar de manera rápida la interfaz de la aplicación. Éste prototipado inicial se muestra a continuación:



Figura 3: Prototipado Interfaz "Iniciar Servidor"

El prototipo que se muestra en la (figura 3) presenta la interfaz para ingresar la aplicación sockets denominada “SERVER-SIMERT”, que consta de lo siguiente:

- En la parte superior el título de la ventana “Iniciar el Servidor”
- En la parte intermedia se encuentra dos campos de textos “usuario” y “contraseña”, para que el administrador del servidor ingrese su nombre y contraseña que son las claves principales para abrir la aplicación.
- En la parte inferior se encuentra un botón abrir donde valida que los campos ingresados como usuario y contraseña sean correctos para abrir la ventana “Servidor SIMERT” y ejecutar la aplicación como se muestra en la (figura 4), también se encuentra un botón cambiar contraseña donde valida que los campos ingresados como usuario y contraseña sean correctos para abrir la ventana “cambiar nombre de usuario y contraseña” como se muestra en la (figura 5).



Figura 4: Prototipado interfaz “Servidor SIMERT”

El prototipo que se muestra en la (figura 5) presenta la interfaz de ejecución del “SERVIDOR SIMERT” que consta de lo siguiente:

- En la parte superior el título de la ventana “SERVIDOR SIMERT”
- En la parte intermedia se encuentra un área de texto donde se visualiza la hora actual, la fecha actual, la dirección IP y el número de puerto del servidor. También se visualizará durante la comunicación las tramas que son los datos de los registros de estacionamiento que serán enviados por los controladores junto con los mensajes de los procesos que se realiza con cada una de ellas; así como también la fecha actual, la hora actual, la dirección IP y número de puerto de los clientes que serán los controladores.
- En la parte inferior se encuentra un botón Salir para abrir la ventana “Salir Servidor” como se muestra en la (figura 6).



Figura 5: Prototipado Interfaz “Cambiar Usuario y Contraseña”

El prototipo que se muestra en la (figura 5) presenta la interfaz “Cambiar Usuario y Contraseña”, que consta de lo siguiente:

- En la parte superior el título de la ventana “Cambiar Usuario y Contraseña”
- En la parte intermedia se encuentra dos campos de textos “usuario” y “contraseña”, para que el administrador del servidor ingrese su nombre y contraseña que son las claves principales que se cambiara, para abrir la aplicación.
- En la parte inferior se encuentra un botón “cambiar” donde valida que los campos ingresados tanto usuario como contraseña tengan mínimo 8 caracteres para modificarse internamente en el fichero autenticación y abrir la ventana “Iniciar Servidor” como se muestra en la (figura 3), también se encuentra un botón “cancelar” para salir de la ventana actual y abrir nuevamente la ventana “Iniciar Servidor”.



Figura 6: Prototipo Interfaz "Salir del Servidor"

El prototipo que se muestra en la (figura 6) presenta la interfaz "Salir del Servidor", que consta de lo siguiente:

- En la parte superior el título de la ventana "Salir del Servidor"
- En la parte intermedia se encuentra dos campos de textos "usuario" y "contraseña", para que el administrador del servidor ingrese su nombre y contraseña que son las claves principales para cerrar la aplicación.
- En la parte inferior se encuentra un botón "salir" donde valida que los campos ingresados como usuario y contraseña sean correctos para cerrar la aplicación sockets "SERVER-SIMERT", también se encuentra un botón "cancelar" para salir de la ventana actual y abrir la ventana "Servidor SIMERT" como se muestra en la (figura 4).

2.2. Diseño del Modelo de casos de uso.

En el siguiente diagrama se muestra los casos de uso de todas las tareas que componen el software "SERVER-SIMERT".

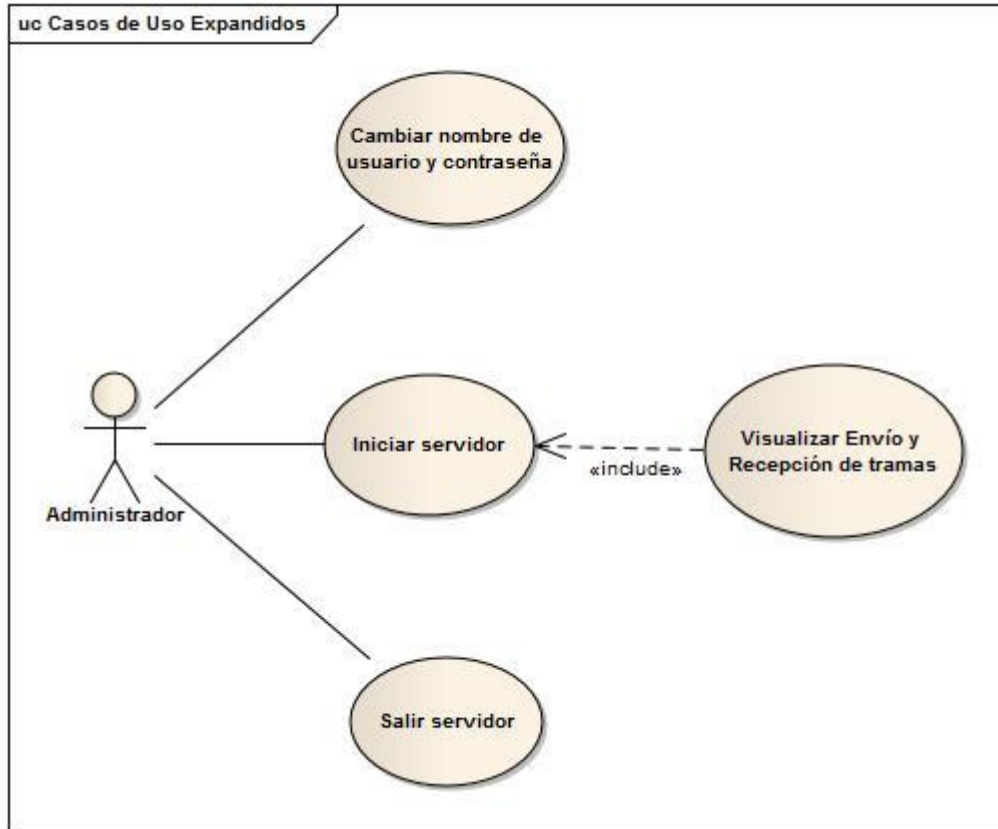


Figura7: Diagrama de casos de uso del software "SERVER-SIMERT".

2.3. Descripción de los casos de uso

2.3.1. Caso de uso 001: Iniciar servidor



Figura 7: Interfaz "Iniciar Servidor"

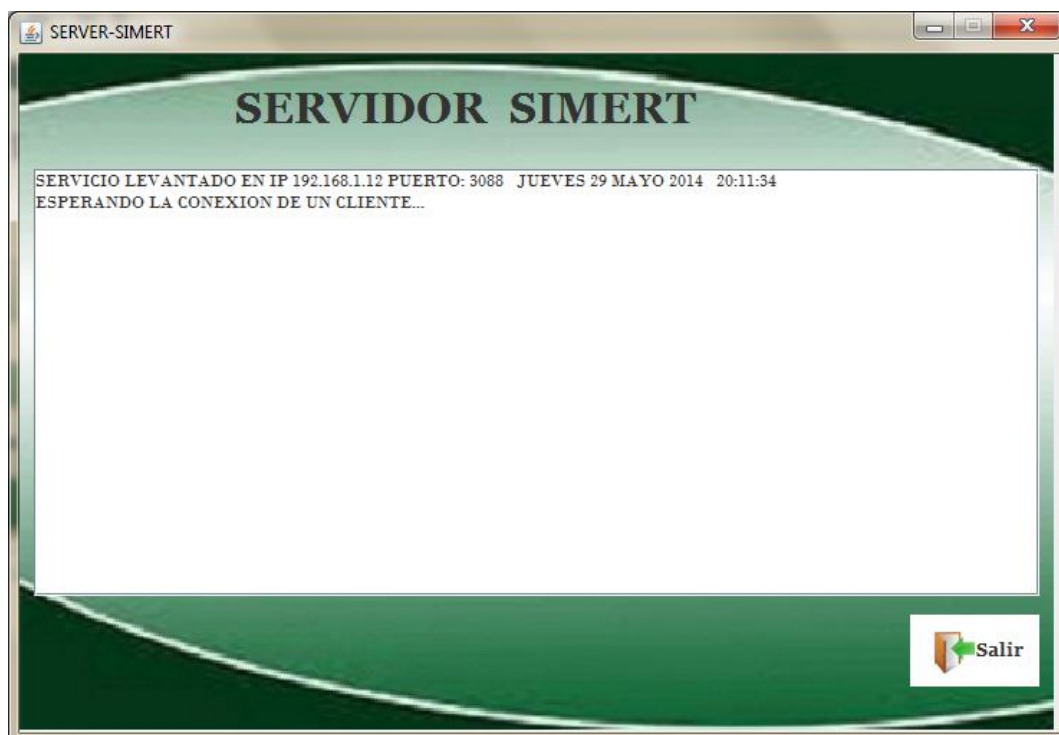


Figura 8: Interfaz “Servidor SIMERT”

TABLA XIV: DESCRIPCIÓN CASO DE USO 001”INICIAR SERVIDOR”

Caso de uso 001	Iniciar servidor
Actor	Administrador del Servidor
Propósito	Ingreso a la aplicación por parte del administrador del servidor.
Resumen	Al ejecutarse la aplicación el administrador observa una interfaz “Iniciar Servidor” como se muestra en la (figura 3); en la cual deberá ingresar el nombre de usuario y contraseña para el acceso al mismo donde le presentará la ventana “SERVIDOR SIMERT” como se muestra en la (figura 4), donde se mantendrá ejecutando la aplicación y estableciendo la comunicación con los clientes que son los controladores.
Tipo	Primario y esencial

Referencias Cruzadas	RF-001, RF-003, RF-012
Pre-condiciones	<ul style="list-style-type: none"> - El administrador debe estar registrado dentro del fichero generado por el sistema. - Que se encuentre en la aplicación y presiona el botón "Iniciar" - El sistema presenta la ventana "Servidor SIMERT"
Post-condiciones	<ul style="list-style-type: none"> - Se inicialice el sistema con sus funcionalidades.
Curso Normal de Eventos	
Acción del Administrador	Respuesta del Sistema
1. Ingresar su usuario y contraseña y presionar el botón "Iniciar".	2. Valida los campos requeridos de usuario y contraseña. 3. Presenta la interfaz "Servidor SIMERT" y visualiza la hora actual, fecha actual, la dirección IP y número de puerto del Servidor que servirán para establecer la comunicación con los controladores.
Curso Alterno de Eventos	
2. Si el usuario no existe presenta el mensaje de error "nombre de usuario incorrecto", si la contraseña no existe presenta el mensaje "contraseña incorrecta" y si ambos parámetros no existen presenta el mensaje de error "nombre de usuario y contraseña incorrectos".	
A. OPCIÓN CAMBIAR CONTRASEÑA	
A1. El administrador presiona el botón "cambiar contraseña"	
A2. El sistema valida los campos ingresados como usuario y contraseña.	
A3. El sistema presenta la ventana "Cambiar usuario y contraseña"	
B. OPCIÓN SALIR	
A1. Cuando el administrador presiona el botón "Iniciar"	

- A2. El sistema presenta la ventana “Servidor SIMERT”
- A3. Donde el administrador presiona el botón “Salir”
- A4. El sistema presenta la ventana “Salir del Servidor” para cerrar definitivamente la aplicación

2.3.2. Caso de uso 002: Cambiar Contraseña



Figura 9: Interfaz “Cambiar Usuario y Contraseña”

TABLA XV. DESCRIPCIÓN CASO DE USO 002 “CAMBIAR CONTRASEÑA”

Caso de uso 002	Cambiar Contraseña
Actor	Administrador del Servidor
Propósito	Cambiar nombre de usuario y contraseña
Resumen	Al ejecutarse la aplicación el administrador observa una interfaz “Iniciar Servidor”; en la cual deberá ingresar el nombre de usuario y contraseña y presionar el botón “cambiar contraseña”. Luego se valida los datos y presenta

	la ventana “Cambiar Usuario y Contraseña” como se muestra en la (figura 5)
Tipo	Primario y esencial
Referencias Cruzadas	RF-001, RF-002, RF-003
Pre-condiciones	<ul style="list-style-type: none"> - El administrador debe estar registrado dentro del fichero generado por el sistema. - Que se encuentre en la ventana “Cambiar Usuario y Contraseña” ingresa los datos de nombre de usuario y contraseña luego presiona el botón “Cambiar”.
Post-condiciones	<ul style="list-style-type: none"> - Se valida los datos como nombre de usuario y contraseña. - Se modifica los datos como nombre de usuario y contraseña.
Curso Normal de Eventos	
Acción del Administrador	Respuesta del Sistema
1. Ingresar dentro de la ventana “Iniciar Servidor” su usuario y contraseña y presiona el botón “Cambiar contraseña”.	2. Valida los campos requeridos de usuario y contraseña.
	3. Presenta la interfaz “Cambiar Usuario y Contraseña”.
4. Ingresar el nuevo usuario y contraseña y presiona el botón “Cambiar”	5. Valida los campos requeridos de usuario y contraseña.
	6. Modifica el nombre de usuario y contraseña.
	7. Presenta nuevamente la interfaz “Iniciar Servidor”
Curso Alternativo de Eventos	

5 Si el usuario no tiene mínimo 8 caracteres presenta el mensaje de error “nombre de usuario mínimo 8 caracteres”, si la contraseña no tiene mínimo 8 caracteres presenta el mensaje “contraseña mínimo 8 caracteres” y si ambos parámetros no tiene mínimo 8 caracteres presenta el mensaje de error “nombre de usuario y contraseña mínimo 8 caracteres”.

A OPCIÓN CANCELAR

- A1. El administrador presión el botón “cancelar”
- A2. El sistema cierra la ventana actual “Cambiar Usuario y Contraseña”.
- A3. El sistema presenta la ventana “Iniciar Servidor”

2.3.3. Caso de uso 003: Salir Servidor



Figura 10: Interfaz “Salir del Servidor”

TABLA XVI: DESCRIPCIÓN CASO DE USO 003 “SALIR SERVIDOR”

Caso de uso 003	Salir Servidor
Actor	Administrador del Servidor
Propósito	Cerrar la aplicación sockets “SERVER-SIMERT”
Resumen	Cuando el administrador ya ha iniciado sesión le presenta la ventana “SERVIDOR-SIMERT” donde se mantiene

	ejecutándose la aplicación, si desea cerrar la aplicación pulsa el botón “Salir” y le presenta la ventana “Salir del Servidor” como se muestra en la (figura 6).	
Tipo	Primario y esencial	
Referencias Cruzadas	RF-001, RF-003	
Pre-condiciones	<ul style="list-style-type: none"> - El administrador debe estar registrado dentro del fichero generado por el sistema. - Que se encuentre en la ventana “Salir del Servidor” ingresa el nombre de usuario y contraseña luego presiona el botón”Salir”. 	
Post-condiciones	<ul style="list-style-type: none"> - Se valida los datos como nombre de usuario y contraseña. - Se cierra la aplicación definitivamente. 	
Curso Normal de Eventos		
Acción del Administrador	Respuesta del Sistema	
1. Presiona el botón “Salir” de la ventana “Servidor SIMERT”.	2. Presenta la interfaz “Salir del Servidor”.	
3. Ingresa el usuario y contraseña y presiona el botón “Salir”	4. Valida los campos requeridos de usuario y contraseña.	
	5. Cierra la aplicación “SERVER-SIMERT”.	
Curso Alterno de Eventos		
<p>4 Si el usuario no existe presenta el mensaje de error “nombre de usuario incorrecto”, si la contraseña no existe presenta el mensaje “contraseña incorrecta” y si ambos parámetros no existen presenta el mensaje de error “nombre de usuario y contraseña incorrectos”.</p>		

A OPCIÓN CANCELAR

- A1. El administrador presión el botón “cancelar”
- A2. El sistema cierra la ventana actual “Salir del Servidor”.
- A3. El sistema presenta la ventana “Servidor SIMERT”

2.3.4. Caso de uso 004: Visualizar envío y recepción de tramas.

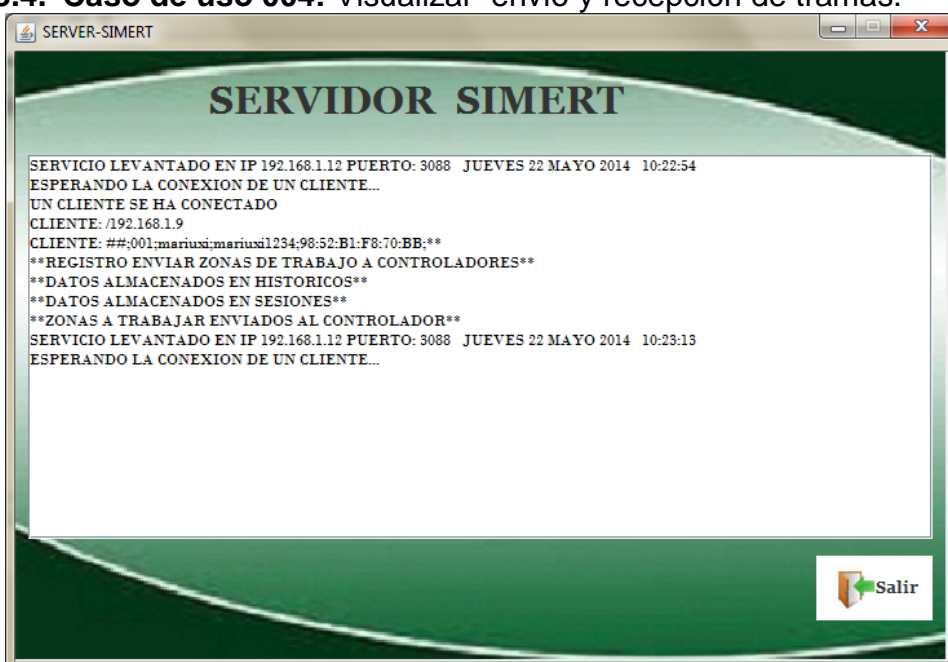


Figura 11: Visualización de la trama “Registro enviar zonas a los controladores”

TABLA XVII: DESCRIPCIÓN CASO DE USO 004 “VISUALIZAR ENVÍO Y RECEPCIÓN DE TRAMAS.”

Caso de uso 004	Visualizar envío y recepción de tramas.
Actor	Controlador
Propósito	Enviar y recibir cada una de las tramas que envían los controladores al servidor y viceversa (ver figura 11). En este caso la trama denominada: “Registro enviar zonas a los controladores” que contiene las zonas de trabajo, los

	tipos de multas y los horarios establecidos para los controladores inicien su jornada de trabajo.	
Resumen	Cuando el administrador ya ha iniciado sesión le presenta la ventana “SERVIDOR-SIMERT” donde se mantiene ejecutándose la aplicación, y estableciendo la comunicación con los clientes que son los controladores quienes envían las tramas sobre los datos de los registros de estacionamiento vehicular, en este caso se realizará el envío y recepción de la trama correspondiente al registro enviar zonas a los controladores	
Tipo	Primario y esencial	
Referencias Cruzadas	RF-004, RF-005, RF-006, RF-007, RF-008., RF-009., RF-010. RF-011., RF-012.	
Pre-condiciones	<ul style="list-style-type: none"> - El administrador debe dejar inicializado la aplicación “SERVER-SIMERT”. - El controlador debe estar registrado en la base de datos. 	
Post-condiciones	<ul style="list-style-type: none"> - El sistema envía las zonas a trabajar si el usuario y contraseña del controlador son correctos. 	
Curso Normal de Eventos		
	Acción del Controlador	Respuesta del Sistema
	1. Envía la trama correspondiente al registro enviar zonas a los controladores.	2. Consulta y valida, el usuario y contraseña del controlador.
		3. Consulta y valida, la mac del dispositivos Tablet del controlador.
		4. Si el usuario, contraseña y mac son correctos realiza lo siguiente:



	<p>4.1. Consulta con el usuario los datos del controlador como: idrol, idempleado, idcontrolador.</p> <p>4.2. Consulta y valida con el usuario, si el controlador ya inicio sesión en el día actual, realizando lo siguiente:</p> <p>4.2.1. Si el usuario ya inicio sesión en ese día, envía la trama de respuesta al controlador que ya inicio sesión.</p> <p>4.2.2. Si el usuario no ha iniciado sesión, realiza lo siguiente:</p> <p>4.2.2.1. Consulta y valida con el usuario, si el controlador ya inicio sesión con la fecha de ayer y realiza lo siguiente:</p> <p>4.2.2.1.1. Si el usuario inicio sesión con la fecha de ayer, realiza lo siguiente:</p> <p>4.2.2.1.1.1. Consulta y valida con el usuario, si el controlador tiene o no zonas asignadas a trabajar en ese día y realiza lo siguiente:</p> <p>4.2.2.1.1.1.1. Si el controlador no tiene zonas a trabajar, envía una trama de respuesta que no tiene zonas a trabajar.</p>
--	---

	<p>4.2.2.1.1.1.2. Si el controlador tiene asignado las zonas a trabajar.</p> <p>Almacena los datos del controlador en la tabla “históricos”.</p> <p>Elimina al controlador de la tabla “sesiones” con la fecha de ayer.</p> <p>Inserta el controlador en la tabla “sesiones” con la fecha actual.</p> <p>Envía la trama de respuesta con las zonas a trabajar y los tipos de multas existentes para que luego las utilice.</p> <p>4.2.2.1.2. Si el usuario no inicio sesión con la fecha de ayer realiza lo siguiente:</p> <p>4.2.2.1.2.1. Consulta y valida con el usuario, si el controlador tiene o no zonas asignadas a trabajar en ese día y realiza lo siguiente:</p> <p>4.2.2.1.2.1.1. Si el controlador no tiene zonas a trabajar, envía una trama de respuesta que no tiene zonas a trabajar.</p> <p>4.2.2.1.2.1.2. Si el controlador tiene asignado las zonas a trabajar.</p>
--	---



	<p>Almacena los datos del controlador en la tabla “históricos”.</p> <p>Elimina al controlador de la tabla “sesiones” con el nombre del controlador.</p> <p>Inserta el controlador en la tabla “sesiones” con la fecha actual.</p> <p>Envía la trama de respuesta con las zonas a trabajar y los tipos de multas existentes para que luego las utilice.</p>
-	<p>5. Si la mac del dispositivo Tablet es incorrecta, se envía la trama de respuesta que el controlador no existe</p>
	<p>6. Si el usuario y contraseña son incorrectos, se envía la trama de respuesta que el controlador no existe</p>
<p>Curso Alterno de Eventos</p>	
<p>1 Trama que envía el controlador. "##;001;mariuxi;mariuxi1234;98:52:B1:F8:70:BB;*" </p> <p>4.2.1 Trama que envía el servidor como respuesta "##;002;3;*" </p> <p>4.2.2.1.1.1 y 4.2.2.1.2.1.1. Trama que envía el servidor como respuesta. "##;002;2;*" </p> <p>4.2.2.1.1.1.2 y 4.2.2.1.2.1.2 Trama que envía el servidor como respuesta. "##;002;1;nombre_controlador;zonastrabajar;vectortiposmulta;*" </p>	

5 y 6 Trama que envía el servidor como respuesta.

"##; 002; 0; **"

	<p>4. Si no es un cliente de recarga, envía la trama de respuesta al controlador que el registro no es de un cliente de recarga y tampoco de un cliente exonerado.</p>
<p>Curso Alterno de Eventos</p>	
<p>1. La trama que envía el controlador</p> <p>"##;003;mariuxi;EPSILON1;20,22-05-2014;10:49;rojo;LBB897;Chevrolet; 98:52:B1:F8:70:BB; **"</p> <p>"##;003;mariuxi;EPSILON1;20,22-05-2014;10:49;blanco;LAG464;suzuki; 98:52:B1:F8:70:BB; **"</p>	

2.4. Diseño del Modelo de procesos o robustez.

2.4.1. Caso de uso 001: Iniciar servidor

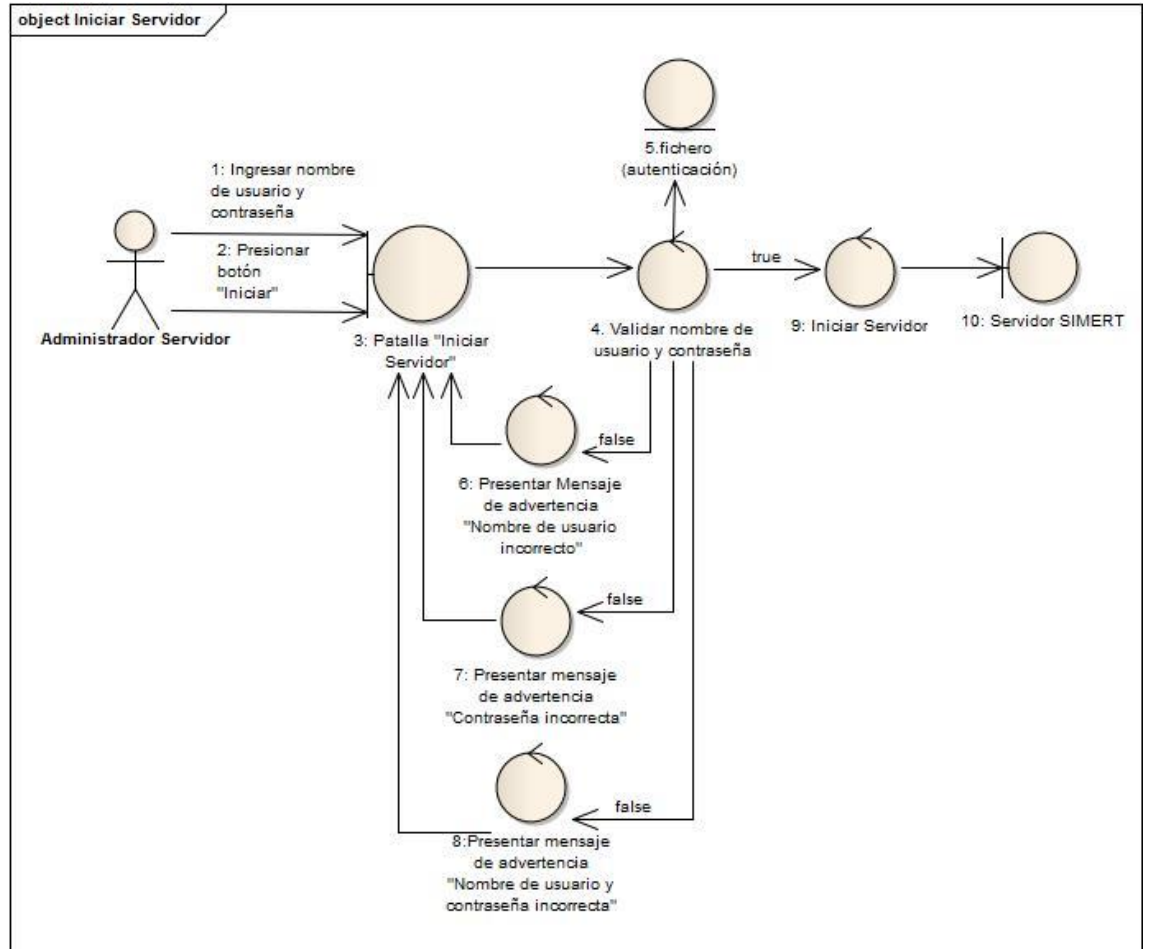


Figura 12: Diagrama de robustez "Iniciar Servidor"

2.4.2. Caso de uso 002: Cambiar Contraseña

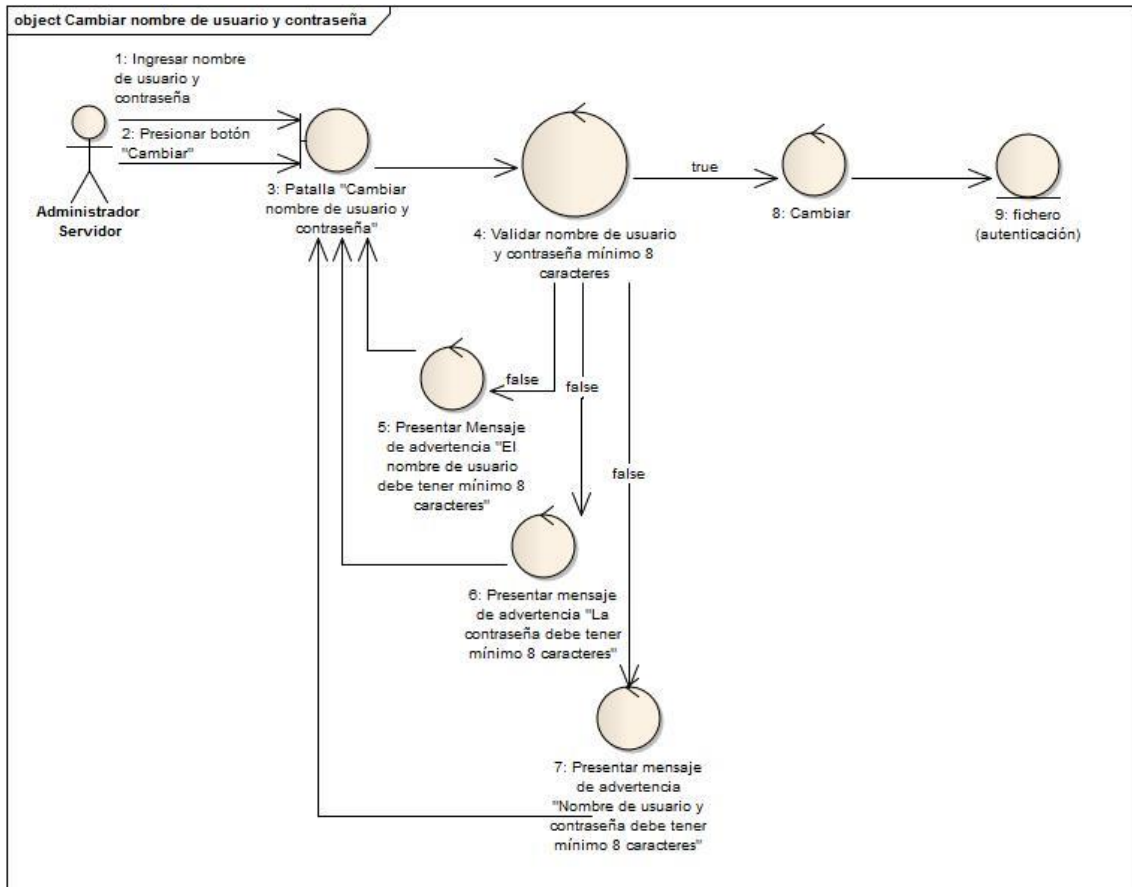


Figura 13: Diagrama de robustez "Cambiar Usuario y Contraseña"

2.4.3. Caso de uso 003: Salir Servidor

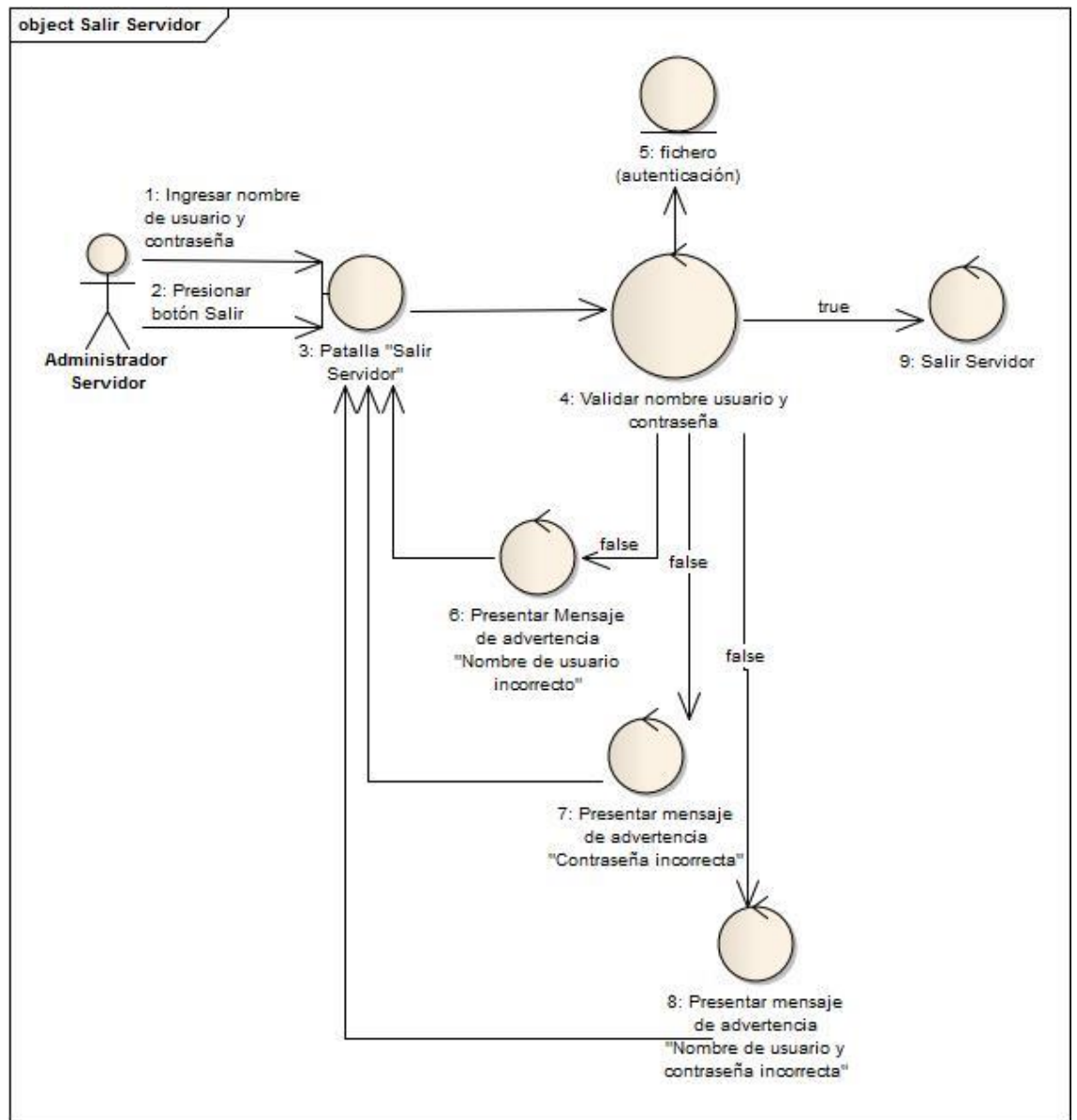


Figura 14: Diagrama de robustez "Salir del Servidor"

2.5. Diseño del Modelo de secuencia.

2.5.1. Caso de uso 001: Iniciar servidor

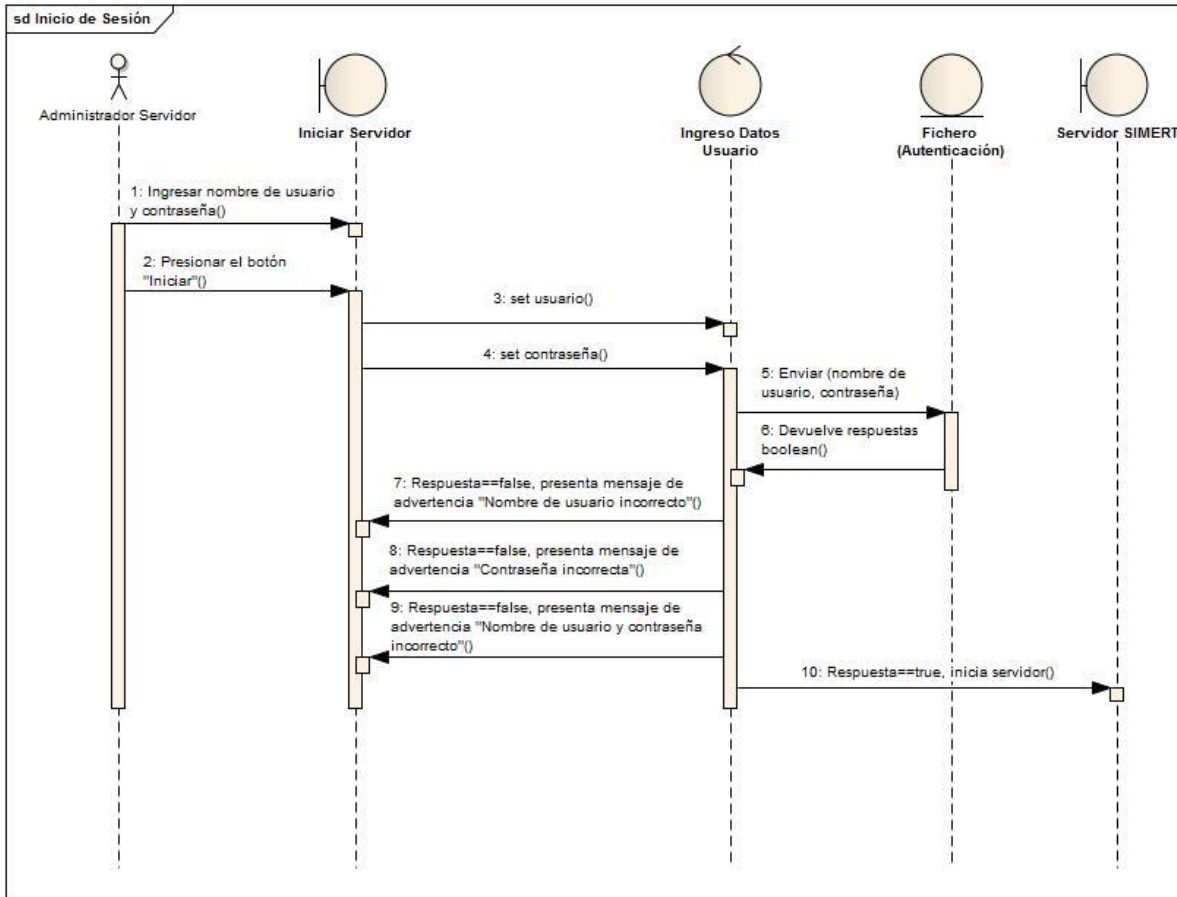


Figura 15: Diagrama de secuencia "Iniciar Servidor"

2.5.2. Caso de uso 002: Cambiar Contraseña

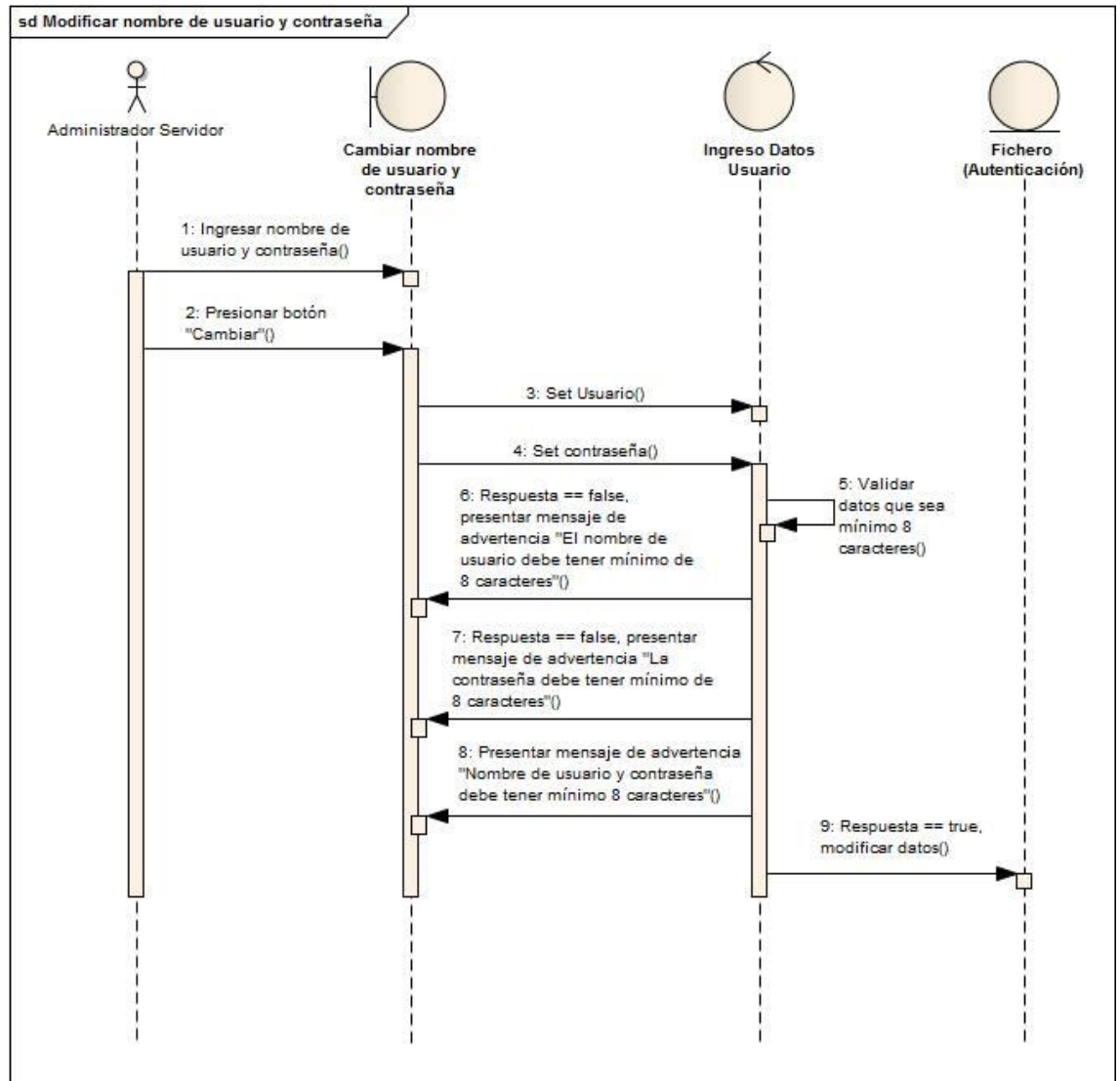


Figura 16: Diagrama de secuencia "Cambiar nombre de usuario y contraseña"

2.5.3. Caso de uso 003: Salir del Servidor

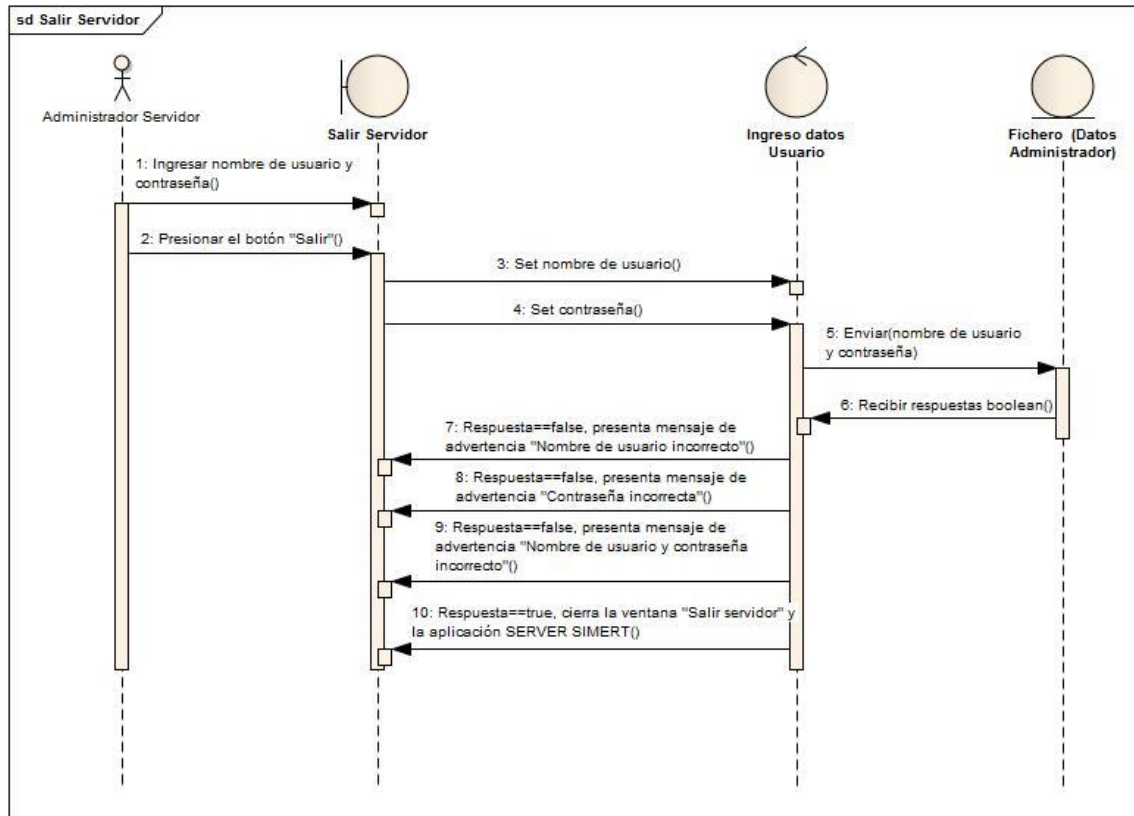


Figura 17: Diagrama de secuencia "Salir del Servidor"

2.6. Diseño de la Arquitectura cliente-servidor para la conectividad de la red de datos del SIMERT.

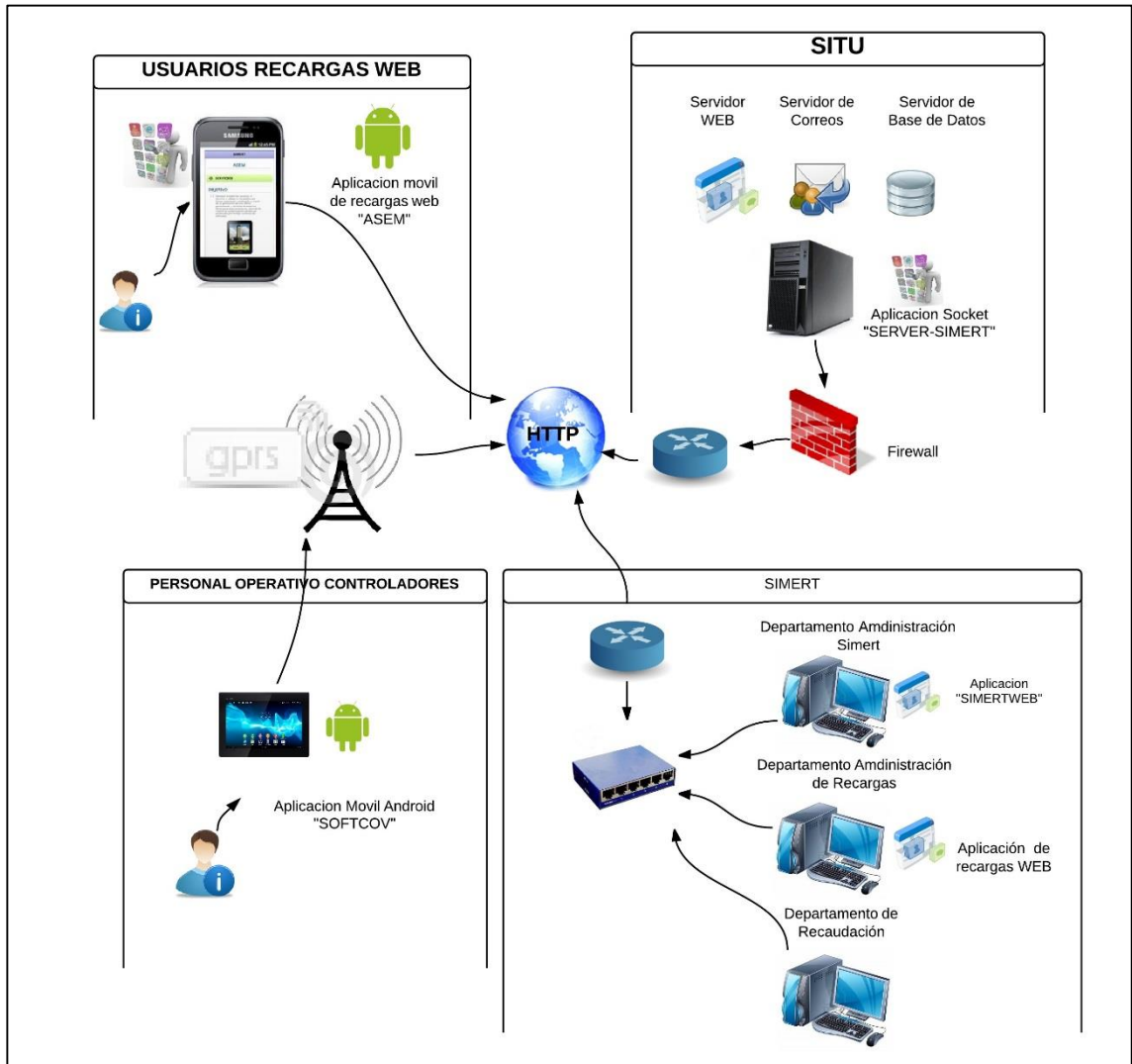


Figura 18: Arquitectura Cliente - Servidor para la conectividad de la red de datos del SIMERT

2.6.1. Simbología de la arquitectura cliente – servidor.

A continuación se detalla la simbología de la arquitectura cliente – servidor:

TABLA XVIII: SIMBOLOGÍA DE LA ARQUITECTURA CLIENTE-SERVIDOR

IMAGEN	DESCRIPCIÓN
	<p>RED GPRS CLARO.</p> <p>La empresa de telecomunicaciones Claro brindara el servicio de GPRS, a través de los chips que cada Tablet utilizara. Se contratara un plan de megas para la trasmisión de datos.</p>
	<p>PROTECCIÓN FIREWALL.</p> <p>Un cortafuego (firewall en inglés) es una parte de un sistema o una red que está diseñada para bloquear el acceso no autorizado, permitiendo al mismo tiempo comunicaciones autorizadas.</p>
	<p>SERVIDOR.</p> <p>Un servidor es un nodo que, formando parte de una red, provee servicios como web, correo y almacenamiento de datos entre otros. Algunos servicios habituales son los servicios de archivos, que permiten a los usuarios almacenar y acceder a los archivos de una computadora y los servicios de aplicaciones, que realizan tareas en beneficio directo del usuario final.</p>
	<p>USUARIO FINAL</p> <p>Persona o personas que van a manipular de manera directa un producto de software.</p>

	<p>BASE DE DATOS</p> <p>Una base de datos o banco de datos es un conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso.</p>
<p>Aplicación Java</p> 	<p>APLICACIÓN JAVA</p> <p>Una aplicación es un tipo de programa informático diseñado como herramienta para permitir a un usuario realizar uno o diversos tipos de trabajos.</p>
	<p>TELÉFONO INTELIGENTE</p> <p>Dispositivo final que interactuara directamente con el usuario final y permitirá el uso y la ejecución de las aplicaciones clientes.</p>
	<p>APLICACIÓN WEB</p> <p>Se denomina aplicación web a aquellas herramientas que los usuarios pueden utilizar accediendo a un servidor web a través de Internet o de una intranet mediante un navegador.</p>
	<p>TABLET</p> <p>Dispositivo final que interactuara directamente con el usuario final y permitirá el uso y la ejecución de las aplicaciones clientes.</p>

	<p>Computadora</p> <p>Dispositivo final que interactuara directamente con el usuario final y permitirá el uso y la ejecución de las aplicaciones clientes.</p>
	<p>Router</p> <p>El encaminador es un dispositivo que permite la conexión a internet como también conectar al mismo tiempo uno o varios equipos o incluso una o varias redes de área local.</p>
	<p>Switch</p> <p>Un switch es un dispositivo de propósito especial diseñado para resolver problemas de rendimiento en la red, debido a anchos de banda pequeños y embotellamientos.</p>
	<p>Internet</p> <p>Internet es un conjunto descentralizado de redes de comunicación interconectadas que utilizan la familia de protocolos TCP/IP.</p>

3. FASE 3: IMPLEMENTACIÓN.

Durante la implementación se utilizó especialmente la fase del análisis y la fase del diseño.

Donde se ha realizado la implementación de todos los diseños así como también la codificación o generación de código Java para la aplicación sockets utilizando la arquitectura Modelo Vista Controlador (MVC), las configuraciones de los equipos de la red, la simulación de la arquitectura cliente-servidor en el Packet Tracer y finalmente las respectivas configuraciones para la creación del servidor SIMERT.

3.1. Creación del servidor SIMERT.

3.1.1. Función del servidor.

El servidor SIMERT proporciona una ubicación central en la red, el mismo que fue ubicado e instalado en la sala de monitoreo del SITU, el cual será configurado correctamente para que funcione como:

Servidor web para el alojamiento de páginas web, servidor de correo para el envío y recepción de correos de los clientes que utilizan el servicio de recargas web, servidor de aplicaciones y servidor de base de datos.

Realizando los procesos como: gestión de la información de los registros de estacionamiento vehicular con la base de datos Mysql, alojamiento de páginas web, almacenamiento de correos electrónicos y aplicaciones en red, cuya intención es potenciar la ejecución de los procesos que realiza diariamente el personal operativo y administrativo del SIMERT con los registros de estacionamiento vehicular, lo que permite una adecuada organización y fiabilidad de la información.

3.1.2. Plataforma para el Servidor.

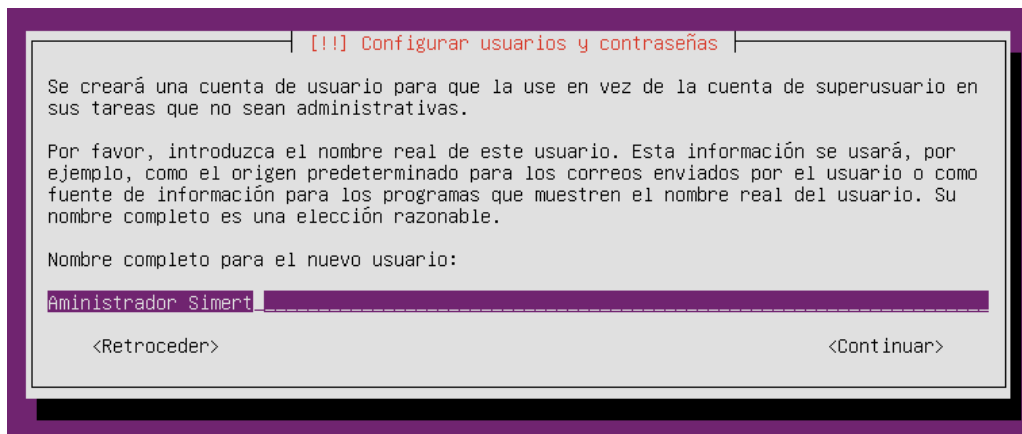
El servidor SIMERT contiene las siguientes características:

- El nombre del servidor es SIMERTNET
- Modelo del servidor es HP Proliant ML 350 G6 E5606
- Procesador: Intel® Xeon® E5606 (4 núcleos, 4.80 GT/s, 2.13 GHz)
- Socket de procesador Socket 1366
- Memoria: Internal memory 4 GB, memoria interna DDR3 máximo 192 GB
- Sistema Operativo Ubuntu Server Edition 10.04 TLS Lucid Lynx
- Base de datos Mysql v. 5.5.2
- Administrar para la base de datos Phpmyadmin v. 5.1.72

- Sistema de infraestructura LAMP
- Postfix v. 2.7 para el almacenamiento de correos
- Apache 2.4.2
- **RED:** 192.168.1.0
- **IP pública:** 190.12.61.30
- **Puerta de enlace:** 192.168.1.99
- **IP Server:** 192.168.1.110
- **DNS:** simernet.sytes.net

3.1.3. Creación de la cuenta para el acceso al Servidor

Se creó el usuario “situ” con el nombre completo de “Administrador SIMERT” para su posterior uso.



[[!]] Configurar usuarios y contraseñas

Se creará una cuenta de usuario para que la use en vez de la cuenta de superusuario en sus tareas que no sean administrativas.

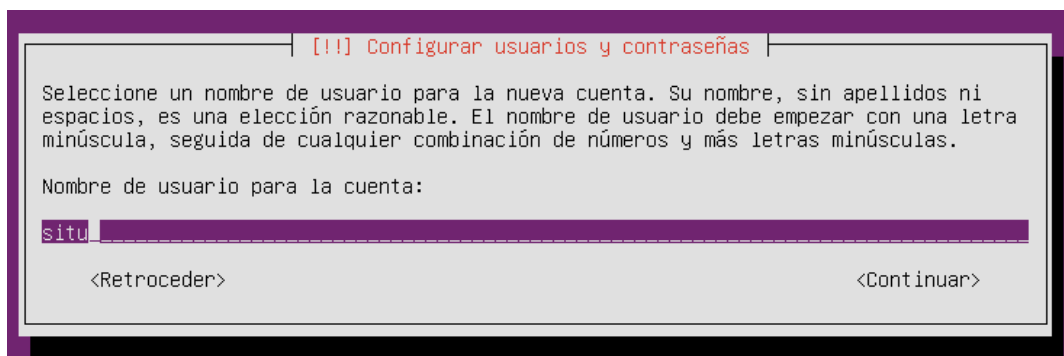
Por favor, introduzca el nombre real de este usuario. Esta información se usará, por ejemplo, como el origen predeterminado para los correos enviados por el usuario o como fuente de información para los programas que muestren el nombre real del usuario. Su nombre completo es una elección razonable.

Nombre completo para el nuevo usuario:

Administrador Simert

<Retroceder> <Continuar>

Figura 19: Configuración de usuario y contraseña para el acceso al servidor



[[!]] Configurar usuarios y contraseñas

Seleccione un nombre de usuario para la nueva cuenta. Su nombre, sin apellidos ni espacios, es una elección razonable. El nombre de usuario debe empezar con una letra minúscula, seguida de cualquier combinación de números y más letras minúsculas.

Nombre de usuario para la cuenta:

situ

<Retroceder> <Continuar>

Figura 20: Ingreso del nombre de usuario para el acceso al servidor

3.1.4. Selección de Paquetes para el servidor.

Los paquetes que se utilizó para la creación básica del servidor fueron: Lamp server, Open ssh, Dns server y Samba Server.

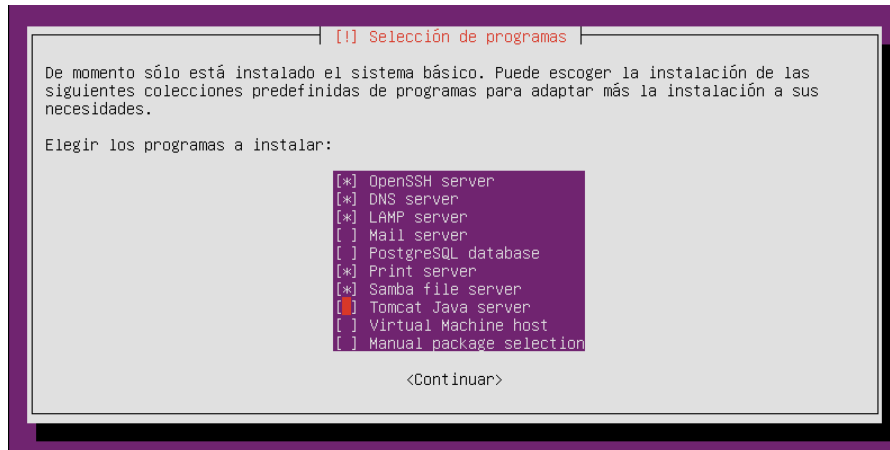


Figura 21: Selección de paquetes para la creación del servidor.

3.1.5. Instalación De Entorno Gráfico del servidor.

La instalación de un entorno grafico facilita la administración de los servidores, por esta razón se instaló el entorno grafico Ubuntu-Unity con el siguiente comando.

TABLA XIX: COMANDO DE INSTALACIÓN DE INTERFAZ

```
situ@simertnet:~$ sudo apt-get install ubuntu-desktop
```

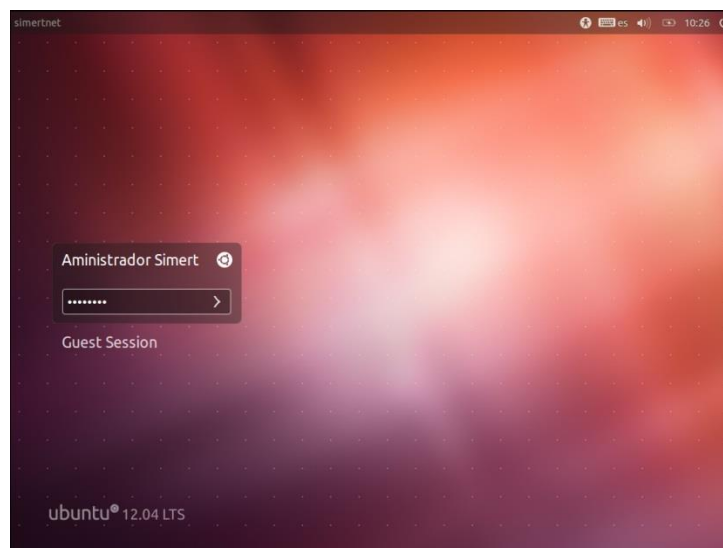


Figura 22: Pantalla principal para el servidor

3.1.6. Configuraciones para la creación del servidor.

En primer lugar, se procedió a configurar los parámetros de red en ambos servidores, siendo la IP pública la siguiente: 190.12.61.30 Servidor Web Posterior a esto se procedió a realizar la respectiva configuración de cada servidor.

3.1.6.1. Configuración de tarjetas de Red.

Necesitaremos que nuestra tarjeta de red tenga una IP estática ya que cuando compartamos un servicio Web abrir y re direccionar el puerto 8080, 3088 para protocolo TCP con tráfico de paquetes entrantes y salientes en la interfaz LAN y WAN.

La tarjeta de red de nuestro servidor se le ha asignado la dirección 192.168.1.110 con los siguientes comandos:

TABLA XLII: COMANDO DE EDICIÓN DE INTERFACES DE RED

```
situ@simertnet:~$ sudo nano /etc/network/interfaces
```

La configuración por defecto en Linux Ubuntu Server hace que nuestra tarjeta de red (eth0) recoja una IP dinámica vía DHCP que nuestro Router le asigna.

Para asignarle la IP 192.168.1.110 modificamos y agregamos las siguientes líneas, referente a nuestra tarjeta de red (eth0) y agregamos la siguiente configuración quedándonos así:

TABLA XX: CONFIGURACIÓN DE LA TARJETA DE RED

```
auto eth0

iface eth0 inet static

address 192.168.1.110

netmask 255.255.255.192

gateway 192.168.1.1
```

Aplicamos los cambios y reiniciamos la red con el siguiente comando:

TABLA XXI: CONFIGURACIÓN DE PROXY

```
situ@simertnet:~$ sudo /etc/init.d/networking restart
```

3.1.6.2. Configuración del Servidor como servidor de Aplicaciones

La configuración del servidor, implicó la instalación del lenguaje de programación JAVA para Linux (jdk 1.8.0) y NetBeans IDE 8.0.

3.1.6.3. Configuración del Servidor como servidor de Dominios (DNS)

Como herramienta principal para esta configuración se utilizó el WEBMIN; la misma que sirvió para realizar la configuración respectiva del servidor DNS, creando el siguiente dominio: `www. simertnet.sytes.net`

3.1.6.4. Configuración del Servidor como servidor de base de datos.

Para el mismo, se requirió configurar el MYSQL para habilitar el servicio.

Phpmyadmin es una herramienta escrita en PHP, usada para la administración de base de datos MySQL a través de páginas web utilizando Internet. Se utilizó la versión 5.1.72 de phpmyadmin con el siguiente comando.

TABLA XXII: COMANDO DE INSTALACIÓN PHPMYADMIN

```
situ@simertnet:~$ sudo aptitude install phpmyadmin
```

Durante la instalación introducimos la nueva contraseña para nuestro usuario root. . Posteriormente seleccionamos la opción “apache2” para que la instalación se adapte al servidor web ya instalado.

3.1.6.4.1. Configuración de roles en phpmyadmin.

Para conservar la integridad de los datos y de las estructuras será conveniente que sólo algunos usuarios puedan realizar determinadas tareas, y que otras, que requieren mayor conocimiento sobre las estructuras de bases de datos y tablas, sólo puedan realizarse por un número limitado. Po esta razón se creó los usuarios: administrador con todos los privilegios y coordinador como usuario restringido.

Para realizar esta configuración seleccionamos la opción “Privilegios” como se muestra en la (figura 23).



Figura 23: Pantalla Principal de phpmyadmin.

Creamos el usuario administrador seleccionando la opción **“Agregar un nuevo usuario”** como se muestra en la (figura 24)

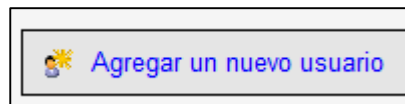


Figura 24: Crear un nuevo usuario para el acceso al MYSQL

Y procedemos a llenar los campos de nuestro nuevo usuario y sus respectivos privilegios

Información de la cuenta

Nombre de usuario: Use el campo de texto

Servidor: Cualquier servidor ¹

Contraseña: Use el campo de texto

Debe volver a escribir:

Generar la contraseña:

Figura 25: Ingreso de datos del usuario Mysql

Seleccionamos los correspondientes privilegios que se otorga al nuevo usuario como se muestra en la (figura 26).

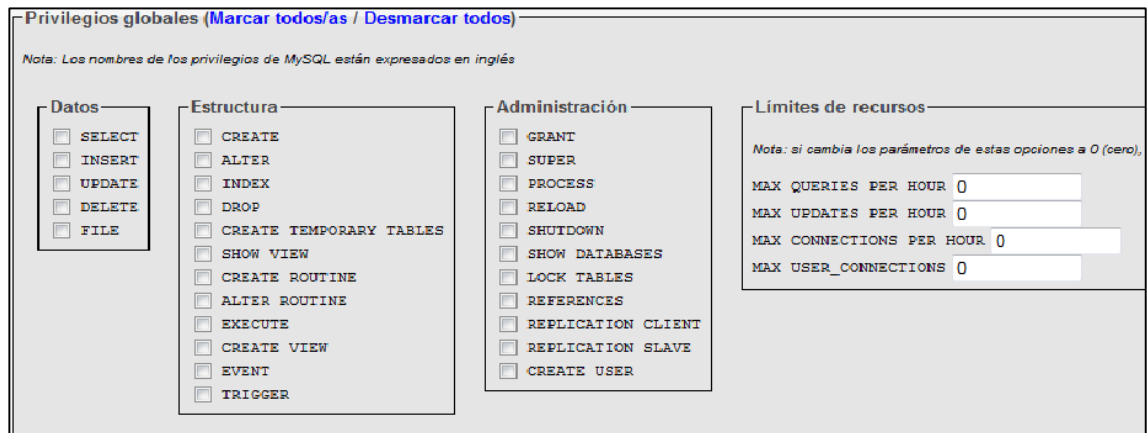


Figura 26: Selección de Privilegios

3.1.6.5. Configuración del Servidor SIMERT como servidor web Apache y su VirtualHost con Nalp

Para que las aplicaciones Web funcionen a cabalidad, se ha utilizado una ip publica que actualmente tiene el SITU. Cabe señalar que para la comunicación con el servidor SIMERTweb se realiza un NAT dentro del proxy donde se direcciona el puerto 8080 y 3088 a nuestro servidor web.

Para ello se creó un virtualhost en el directorio /etc/apache2/sites-available con la siguientes especificaciones (ver figura 27).

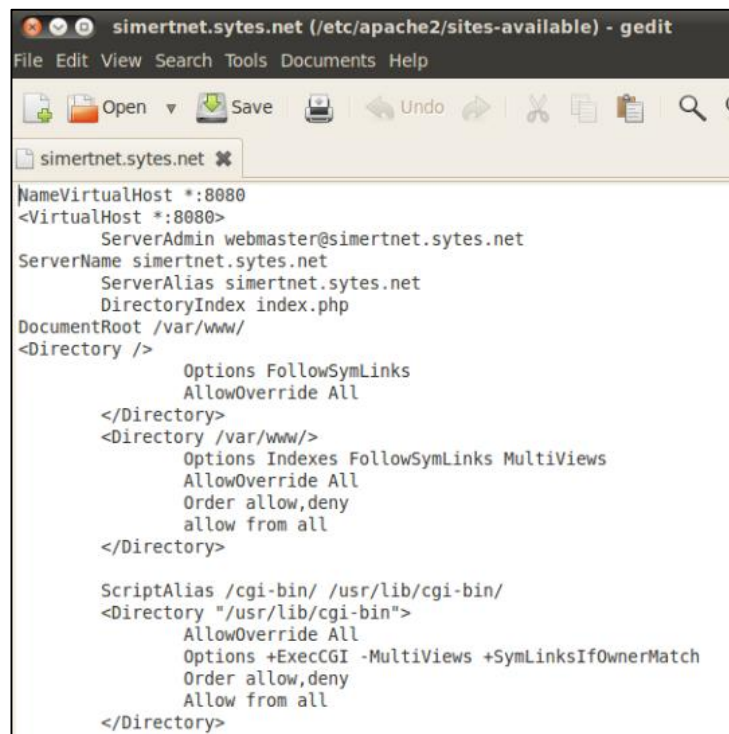


Figura 27: Creación de VirtualHost

Posteriormente aplicamos los cambios con el siguiente comando:

TABLA XXIII: COMANDO DE REINICIO DE VIRTUALHOST

```
situ@simertnet:~$ sudo a2ensite simertnet.sytes.net  
  
situ@simertnet:~$ sudo /etc/init.d/apache2 restart
```

Los diferentes proyectos de las aplicaciones web se los ha ubicado dentro del directorio web por defecto de Linux, en `/var/www/`. Posteriormente se ha establecido los permisos correspondientes como lectura, escritura y ejecución.

3.1.6.6. Configuración del Servidor SIMERT como Servidor de Correo

Para habilitar este servidor, se realizó las respectivas configuraciones, tanto del **Postfix** como el Gmail.

3.1.6.6.1. Instalación de Postfix

Se instaló Postfix en su versión 2.7 con el siguiente comando:

TABLA XXIV: COMANDO DE INSTALACIÓN DE POSTFIX Y DEPENDENCIAS

```
situ@simertnet:~$ sudo apt-get install postfix mailutils libsasl2-2  
libsasl2-modules
```

Posteriormente nos solicitará ciertas configuraciones como:

Tipo de configuración: "Sitio de Internet".

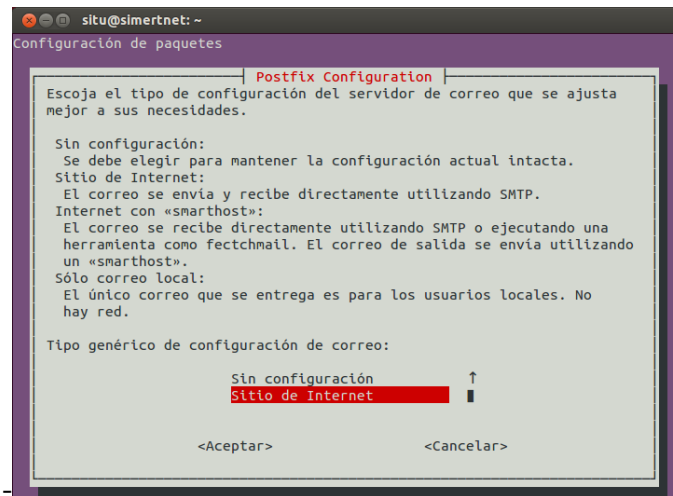


Figura 28: Configuración de correo

Nombre del sistema de correo: **“simertnet”**.

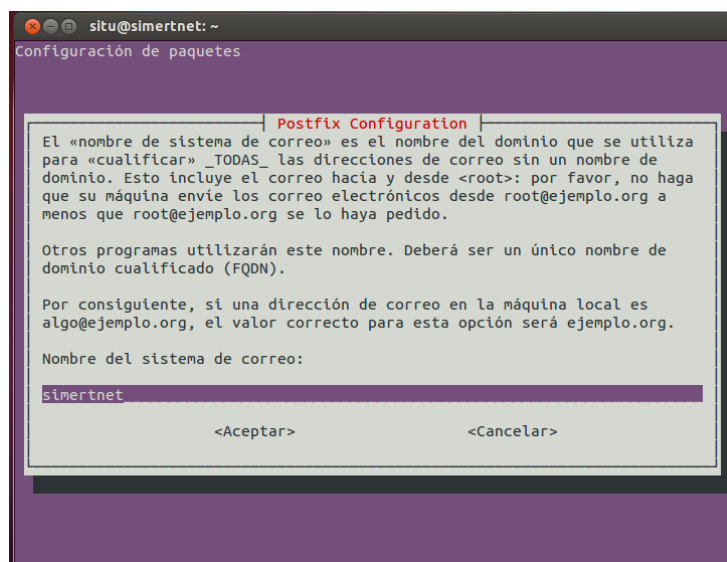


Figura 29: Ingreso del nombre para el sistema de Correos "simertnet"

3.1.6.6.2. Configuración de main.cf

Main.cf es el archivo de configuración de Postfix, la mayoría de las opciones de configuración se especifican en este archivo.

Modificamos el archivo main.cf con el siguiente comando:

TABLA XXV: COMANDO DE EDICIÓN DE ARCHIVO MAIN.CF

```
situ@simertnet:~$ sudo nano /etc/postfix/main.cf
```

Añadimos las siguientes líneas:

TABLA XXVI: LÍNEAS DE CONFIGURACIÓN DE MAIN.CF

```
host = [smtp.gmail.com]:587
_sasl_auth_enable = yes
_sasl_password_maps = hash:/etc/postfix/sasl/passwd
_sasl_security_options = noanonymous
use_tls = yes
tls_CAfile = /etc/postfix/cacert.pem
```

En resumen, le estamos diciendo a postfix que utilice relayhost para conectarse al servidor de Gmail, que utilice smtp_sasl_password_maps para extraer los datos de conexión SASL y que utilice smtp_tls_CAfile como certificado para la conexión segura.

3.1.6.6.3. Autenticación de correo

Para la autenticación de correo se utilizó el SMTP de gmail que nos permitirá el envío de correos externos.

Se creó el archivo **/etc/postfix/sasl/passwd** con el siguiente contenido:

TABLA XXVII: AUTENTICACIÓN DE CORREO

```
situ@simertnet:~$ echo "[smtp.gmail.com]:587
simertnet@gmail.com:*****" > /etc/postfix/sasl/passwd
```

Y lo protegemos adecuadamente con el siguiente comando:

TABLA XXVIII: COMANDO PARA CAMBIO DE PERMISOS

```
situ@simertnet:~$ chmod 600 /etc/postfix/sasl/passwd
```

Seguidamente, se transformó el archivo passwd a un fichero indexado de tipo hash mediante la instrucción:

TABLA XXIX: COMANDO PARA LA CREACIÓN DE ARCHIVOS HASH

```
situ@simertnet:~$ postmap /etc/postfix/sasl/passwd
```

3.1.6.6.4. Utilización del certificado adecuado.

Los Certificados SSL se utilizan para validar la cadena de confianza de todas las Autoridades Certificadoras pertenecientes a la jerarquía nacional de certificación. Para certificar nuestro sistema de correo se instaló los certificados correspondientes. Con el siguiente comando:

TABLA XXX: COMANDO PARA INSTALAR CERTIFICADOS

```
situ@simertnet:~$ aptitude install ca-certificates
```

Para añadir la autoridad certificadora Equifax (la que certifica correos de Gmail) al fichero de certificados que utilizará postfix, ejecutamos el siguiente comando en una consola de root:

TABLA XXXI: COMANDO PARA AÑADIR CERTIFICADOS EQUIFAX

```
situ@simertnet:~$ cat /etc/ssl/certs/Equifax_Secure_CA.pem >
/etc/postfix/cacert.pem
```

3.1.6.6.5. Añadir un remitente nuevo en nuestra cuenta de gmail

El remitente nuevo que se está usando, se lo denominó “situ situ@simertnet.sytes.net”, Para su configuración nos ubicamos en opciones Configuración > Cuentas e importación > Enviar mensaje como y añadimos como nuevo remitente la dirección de correo de nuestro servidor de correo (ver figura 26)

General	Etiquetas	Recibidos	Cuentas	Filtros	Reenvío y correo POP/IMAP	Chat	Clips de la Web	Labs	Sin cone
Cambiar la configuración de la cuenta:		Cambiar contraseña Cambiar opciones de recuperación de contraseña Otra configuración de la cuenta de Google							
Enviar mensaje como: <small>(Utiliza Gmail para enviar mensajes desde tus otras direcciones de correo electrónico)</small> Más información		Simert net <simertnet@gmail.com> Dirección de respuesta: situ@simertnet.sytes.net		establecer como predeterm					
		situ <situ@simertnet.sytes.net> Añadir otra dirección de correo tuya		predeterminada					
		Cuando responda a un mensaje: <input type="radio"/> Responder desde la misma dirección a la que se ha enviado el mensaje <input checked="" type="radio"/> Responder siempre desde mi dirección predeterminada (actualmente es situ@simertnet.sytes)							
		<small>(Nota: podrás modificar tu dirección cuando respondas al mensaje. Más información)</small>							
Comprobar el correo de otras cuentas (mediante POP3): Más información		Añadir una cuenta de correo POP3 tuya							
¿Utilizas Gmail para trabajar?		Las empresas pueden utilizar la tecnología de Gmail para el correo electrónico del trabajo. Más in							
Conceder acceso a tu cuenta: <small>(Permite a otros usuarios leer y enviar mensajes en tu nombre)</small> Más información		Añadir otra cuenta							
		<input checked="" type="radio"/> Marcar la conversación como leída cuando la abren otros usuarios <input type="radio"/> Dejar las conversaciones como no leídas cuando las abren otros usuarios							
Añadir espacio de almacenamiento adicional:		Estás utilizando 0 GB (0%) de tus 15 GB. ¿Necesitas más espacio? Comprar más espacio de almacenamiento							

Figura 30: Configuración del Nuevo Remitente

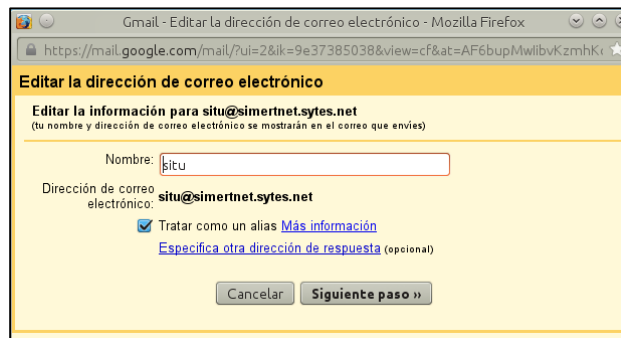


Figura 31: Configuración de Nombre del remitente

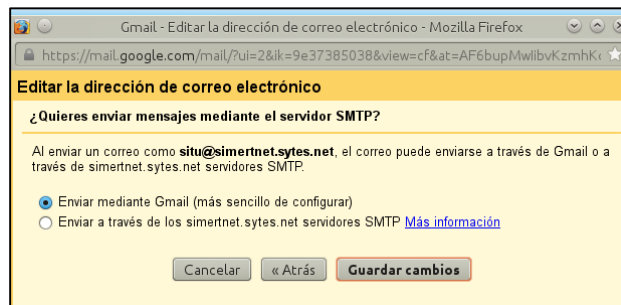


Figura 32: Elección de SMTP

Para verificar que somos usuarios legítimos de la misma gmail nos enviará un mensaje con un código de verificación, que nos llegará a nuestra cuenta local. Una vez verificada la cuenta, gmail permitirá que se envíen mensajes con ese remitente y no lo modificará.

3.1.6.6.6. Puesta en funcionamiento.

Para comprobar el funcionamiento correcto del servidor de correo se aplicó una serie de comandos. En un primer terminal se ejecutó el siguiente comando para monitorear el comportamiento del correo.

TABLA XXXII: COMANDO DE VISUALIZACIÓN DE ENVÍO DE CORREO

```
situ@simertnet:~$ tail -f /var/log/mail.log
```

Y en el segundo terminal ejecutamos:

TABLA XXXIII: COMANDO DE ENVÍO DE CORREO

```
situ@simertnet:~$ echo "Éste es un correo de prueba" | mail
correo[arroba].com
```

Como resultado final tendremos la siguiente imagen:

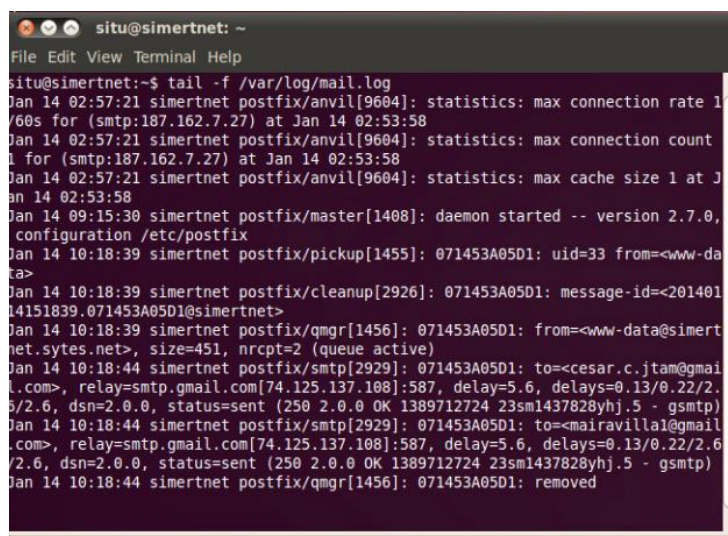


Figura 33: Comprobación de envío de correo

3.1.7. Configuración física del Router

TABLA XXXIV: CONFIGURACIÓN FÍSICA DEL ROUTER

EQUIPO	INTERFACES	DIRECCIÓN IP	MASCARA
Router	Serial	192.168.1.1	255.255.255.224
	Fast Ethernet	192.168.1.8	255.255.255.0

3.1.8. Configuración del Proxy

En el firewall se realizó la apertura de los puertos 8080 y 3088 para el protocolo TCP con tráfico de paquetes entrantes y salientes en la interfaz LAN y WAN mediante las siguientes líneas de comando:

TABLA XXXV: CONFIGURACIÓN DE PROXY

```
iptables --table nat --append PREROUTING --in-interface eth0 --protocol tcp
--dport 8080 --jump DNAT --to 192.168.1.110
iptables --table nat --append PREROUTING --in-interface eth1 --protocol tcp
--dport 8080 --jump DNAT --to 192.168.1.110

iptables --table nat --append PREROUTING --in-interface eth0 --protocol tcp
--dport 3088 --jump DNAT --to 192.168.1.110
iptables --table nat --append PREROUTING --in-interface eth1 --protocol tcp
--dport 3088 --jump DNAT --to 192.168.1.110
```

```
iptables --table nat --append PREROUTING --in-interface eth0 --protocol tcp
--dport 8080 --jump DNAT --to 192.168.1.110
iptables --table nat --append PREROUTING --in-interface eth0 --protocol tcp
--dport 3088 --jump DNAT --to 192.168.1.110

iptables --table filter --append FORWARD --out-interface eth1 --protocol tcp
--dport 8080 --jump ACCEPT
iptables --table filter --append FORWARD --out-interface eth1 --protocol tcp
--dport 3088 --jump ACCEPT

iptables --table filter --append OUTPUT --protocol tcp --dport 8080 --jump
ACCEPT
iptables --table filter --append OUTPUT --protocol tcp --dport 3088 --jump
ACCEPT
```

3.2. Codificación o generación de código Java para la aplicación sockets denominada “SERVER-SIMER” para el servidor.

3.2.1. Requerimientos de hardware y software para la instalación y utilización de la aplicación “SERVER-SIMERT”.

Para el desarrollo de la aplicación sockets denominada “SERVER-SIMERT” se ha utilizado los siguientes requerimientos:

3.2.1.1. Requerimientos de hardware

Entre los requerimientos de hardware tenemos:

- Procesador: Intel ® Xeon ® E5606 (4 núcleos, 4.80 GT/s, 2.13 GHz)
- Socket de procesador Socket 1366
- Memoria: Internal memory 4 GB, memoria interna DDR3 máximo 192 GB

3.2.1.2. Requerimientos de software

Entre los requerimientos del software tenemos:

- Sistema Operativo Ubuntu Server Edition 10.04 TLS Lucid Lynx
- Base de Datos Mysql v. 5.5.2
- Phpmyadmin v. 5.1.72 para la administración de la base de datos
- Sistema de infraestructura LAMP
- Postfix v. 2.7 para el almacenamiento de correos

- Lenguaje de programación JAVA, JDK 1.8.0, plataforma virtual NETBEANS IDE 8.0
- Clase ServerSockets para la comunicación con los clientes.
- Mysql conecctor -java 5.1.6 -bin.jar
- Librerías Commons-pool-1.6 y Commons-dbcp-1.4 para el pool de conexiones.
- Librería Jasypt-1.9.1. para la encriptación y des encriptación de datos.
- Enterprise Architect v 8.0 Herramienta case para la modelación del sistema.

3.2.2. Estándares para el código de la aplicación “SERVER-SIMERT”.

Para todo el proceso de implementación del código de la aplicación sockets denominada “SERVER-SIMERT”, se utilizó los estándares o convenciones de programación empleados en el desarrollo de software sobre la plataforma Java, las mismas que facilitan el mantenimiento, la legibilidad y comprensión rápida de una aplicación, entre los estándares que se siguieron son:

- Se aconseja, como buena práctica de programación, incluir en la entrega de la aplicación la documentación de los ficheros fuente de todas las clases. Dicha documentación será generada por la herramienta "javadoc". Como norma general utilizaremos las siguientes etiquetas:
 - ✓ @author Nombre: Añade información sobre el autor o autores del código.
 - ✓ @version InformacionVersion: Permite incluir información sobre la versión y fecha del código.
 - ✓ @since Version: Se utiliza para especificar cuándo se ha añadido a la API la clase, interfaz, método o campo. Debería incluirse el número de versión u otro tipo de información.
 - ✓ @return Descripción: Inserta la descripción indicada en la sección "Returns:" de la documentación del método. Este tag debe aparecer en los comentarios de documentación de todos los métodos, salvo en los constructores y en aquellos que no devuelvan ningún valor (void).
 - ✓ @throws NombreClase Descripción: Añade el bloque de comentario "Throws:" incluyendo el nombre y la descripción de la excepción especificada. Todo comentario de documentación de un método debe contener un tag "@throws" por cada una de las excepciones que pueda elevar. La descripción de la excepción puede ser tan corta o larga como sea necesario y debe explicar el motivo o motivos que la originan.

- Los paquetes se escribirán siempre en letras minúsculas para evitar que entren en conflicto con los nombres de clases o interfaces. Por ejemplo:
`java.io.FileReader;`
`java.awt.event.MouseListener;`
`javax.swing.JFrame;`
- Para la importación de paquetes obligatorios seguirá el siguiente orden:
 - ✓ Paquetes del JDK de java.
 - ✓ Paquetes de utilidades no pertenecientes al JDK de Java, de framework de desarrollo o de proyectos opensource tales como apache, hibernate, springframework, etc.
 - ✓ Paquetes desarrollados para el Ayuntamiento de Málaga.
 - ✓ Paquetes de la aplicación
- Los nombres de clases deben ser sustantivos y deben tener la primera letra en mayúsculas. Si el nombre es compuesto, cada palabra componente deberá comenzar con mayúsculas. Los nombres serán simples y descriptivos. Debe evitarse el uso de acrónimos o abreviaturas. Por ejemplo:
`public class IngresoDatosUsuario()`
- Los métodos deben ser verbos escritos en minúsculas. Cuando el método esté compuesto por varias palabras cada una de ellas tendrá la primera letra en mayúsculas. Por ejemplo:
`Public String realizaLecturaEscrituraFichero ()`
- Como norma general se establecen 4 caracteres como unidad de sangría.
- La longitud de línea no debe superar los 80 caracteres por motivos de visualización e impresión. Cuando una expresión ocupe más de una línea, esta se podrá romper o dividir en función de los siguientes criterios:
 - ✓ Tras una coma.
 - ✓ Antes de un operador.
 - ✓ Alinear la nueva línea con el inicio de la expresión al mismo nivel que la línea anterior.
 - ✓ Si las reglas anteriores generan código poco comprensible, entonces estableceremos tabulaciones de 8 espacios.
- Distinguiremos tres tipos de comentarios de implementación:
 - ✓ Comentarios de bloque: Permiten la descripción de ficheros, clases, bloques, estructuras de datos y algoritmos, por ejemplo:

```
/**
 * comprueba si el fichero de autenticación existe
 * @return: si o no el fichero existe
 */
```

```
Public String realizaLecturaEscrituraFichero ()
```

- ✓ Comentarios de línea: Son comentarios cortos localizados en una sola línea y tabulados al mismo nivel que el código que describen. Si ocupa más de una línea se utilizará un comentario de bloque. Deben estar precedidos por una línea en blanco, por ejemplo:

```
/**Variables para almacenar los datos de usuario y contraseña**/
String usuario1="", contrasena1="";
```

- ✓ La sentencia "try/catch", en el bloque "catch" siempre se imprimirá una traza de error indicando el tipo de excepción generada siempre debe tener el formato siguiente,

```
try {
    sentencias;
} Catch (ClaseException e) {
    sentencias;
}
```

- Las variables se escribirán siempre en minúsculas. Las variables compuestas tendrán la primera letra de cada palabra componente en mayúsculas. Nunca podrán comenzar con el carácter "_" o "\$". Los nombres de variables deben ser cortos y sus significados tienen que expresar con suficiente claridad la función que desempeñan en el código. Debe evitarse el uso de nombres de variables con un sólo carácter, excepto para variables temporales. Por ejemplo:

```
String datos_usuariopass="";
Int alerta=0;
```

- Se utilizarán espacios en blanco en los siguientes casos:
 - ✓ Entre una palabra clave y un paréntesis. Esto permite que se distingan las llamadas a métodos de las palabras clave. Por ejemplo:

```
while (true) {
...
}
```

- ✓ Tras cada coma en un listado de argumentos. Por ejemplo:

```
Public String validarUsuarioContrasen(String login, String login1)
```


- ✓ Para separar un operador binario de sus operandos, excepto en el caso del operador ("."). Nunca se utilizarán espacios entre los operadores unarios ("++" o "--") y sus operandos. Por ejemplo:
Cont++;
- ✓ Para separar las expresiones incluidas en la sentencia "for". Por ejemplo:
for (expresion1; expresion2; expresion3)
for (int i = 0; i < tamañomensaje; i++)

3.2.3. Arquitectura de la aplicación Sockets “SERVER-SIMERT”.

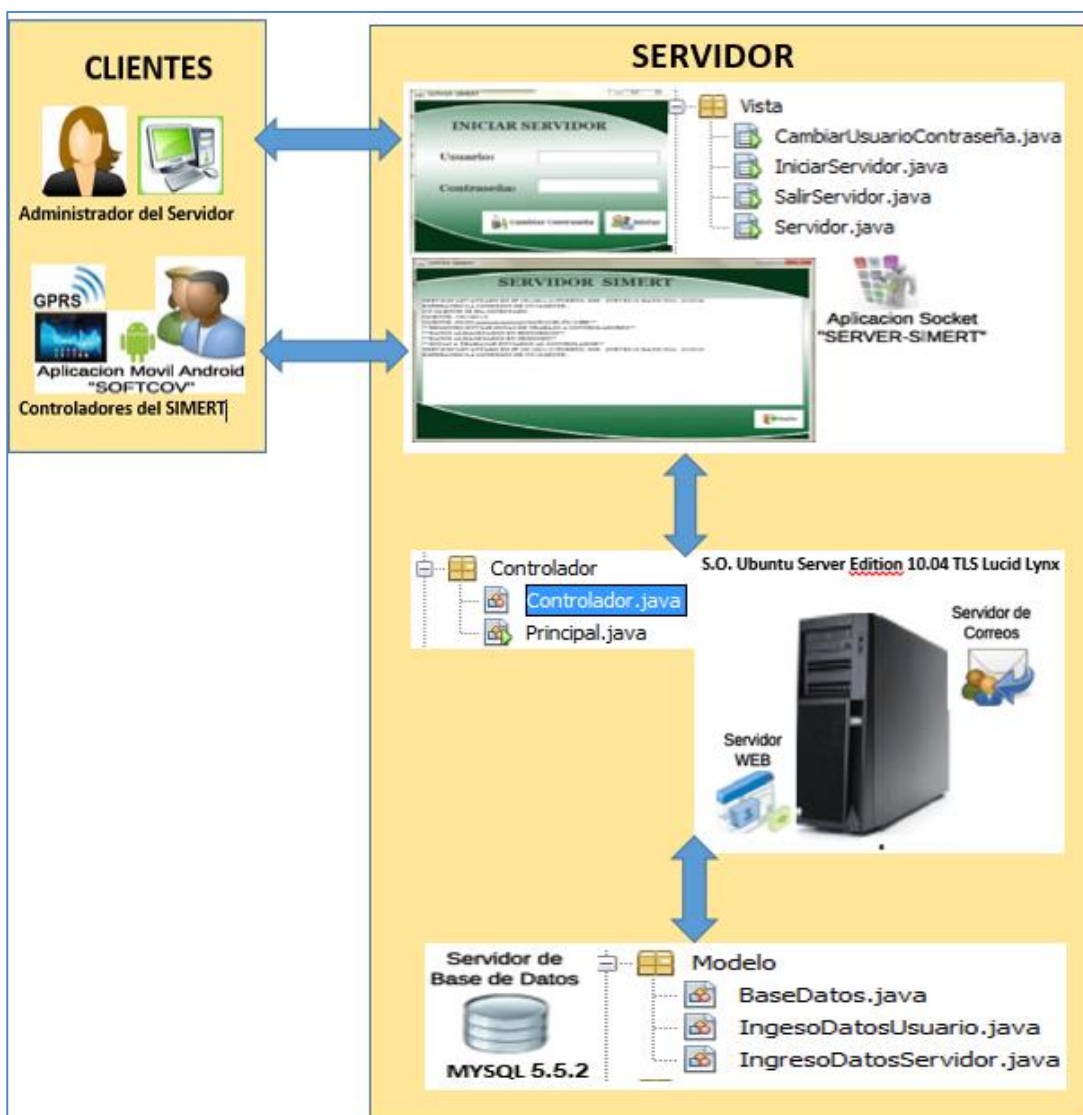


Figura 34: Arquitectura Cliente-Servidor con MVC para la aplicación sockets "SERVER-SIMERT"

Como se muestra en la (figura 34) la aplicación sockets denominada "SERVER-SIMERT" consta de una arquitectura Cliente-Servidor que utiliza el modelo-vista-controlador (MVC) a continuación se detallan cada uno de las capas:

La capa vista: formada por los recursos textuales y la interfaz gráfica que serán interactuadas por el usuario administrador del servidor y a su vez para visualizar las conexiones de los clientes que acceden al servidor en este caso los controladores del SIMERT.

La capa controlador: contiene los métodos para interactuar con la capa vista y modelo.

La capa modelo: contiene las sentencias SQL para el acceso y conexión con la base de datos, así como también la autenticación de acceso al sistema del usuario administrador del servidor y la validación de los datos para la conexión del servidor con los clientes.

3.2.3.1. CAPA VISTA

La capa vista corresponde a cada una de las actividades que constituyen la aplicación en cada una de ellas se presenta los componentes que permitirán al usuario el ingreso y visualización de información.

En esta capa el usuario administrador del servidor interactúa con la interfaz de usuario en este caso como se muestra en la (figura 35) para iniciar el servidor realiza el ingreso de datos como nombre de usuario y contraseña donde escoge la opción Iniciar, estas acciones son enviadas a la capa controlador para el acceso al sistema y ejecución de la aplicación.



Figura 35: Interfaz gráfica "Iniciar Servidor"

La capa vista solo contiene la interfaz gráfica de las ventanas, donde los objetos de cada una de las interfaces envían la notificación de la acción solicitada por el usuario a la capa controlador para generar la salida solicitada para el usuario.

En la (figura 36) se muestra la capa vista con cada una de las clases pertenecientes a las ventanas que se utiliza dentro de la aplicación.

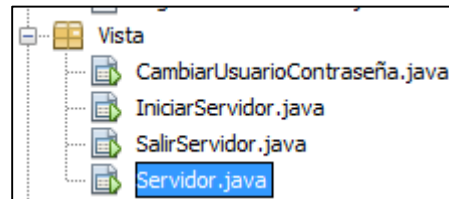


Figura 36: Capa Vista

3.2.3.2. CAPA CONTROLADOR

La capa controlador responde a eventos (acciones del usuario administrador del servidor) de la capa vista e invoca peticiones a la capa modelo cuando se hace alguna solicitud sobre la información (por ejemplo, insertar un registro en una base de datos).

La capa controlador hace de intermediario entre la 'vista' y el 'modelo'.

Esta capa contiene dos tipos de clases como se muestra en la (figura 37), la clase Principal como su nombre lo dice es la clase que llama a ventana principal de la capa vista “iniciar servidor” para ejecutar la aplicación y la clase controlador donde se encuentran todos los métodos para responder a los eventos de la capa vista e invocar las peticiones a la capa modelo.

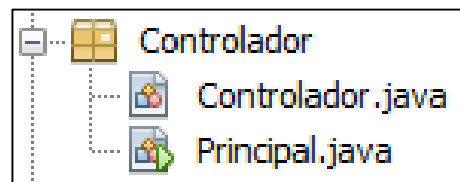


Figura 37: Capa Controlador

3.2.3.3. CAPA MODELO

Contiene información con la cual el sistema opera, por lo tanto gestiona todos los accesos a dicha información, tanto consultas como actualizaciones, implementando también los privilegios de acceso que se hayan descrito en las especificaciones de la

aplicación (capa controlador). Se encarga de enviar a la capa vista aquella parte de la información que en cada momento se le solicita para que sea mostrada (al usuario administrador del servidor). Las peticiones de acceso o manipulación de información llegan a la capa modelo a través de la capa controlador

La capa modelo consta de 3 clases como se muestra en la (figura 38), la clase Base de Datos donde se encuentra los métodos de conexión y acceso a la base de datos. La clase IngresoDatosUsuario donde se encuentra los datos para la autenticación de acceso al sistema del usuario administrador del servidor y la clase IngresoDatosServidor donde se encuentran los datos para la conexión del servidor y los clientes.

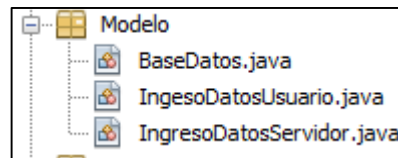


Figura 38. Capa Modelo

A continuación se muestra en la (Tabla XXXVI) la generación del código para establecer la conexión con la base de datos que se encuentra en la clase Base Datos.

TABLA XXXVI: CÓDIGO PARA LA CONEXIÓN CON LA BASE DE DATOS.

CÓDIGO FUENTE PARA LA CONEXIÓN CON LA BASE DE DATOS
<pre> /*la variable dataSource es la que debería ver el resto de nuestro programa*/ private DataSource; public void inicializaDataSource() { /*Hacer el select, insert, update, delete*/ BasicDataSource = new BasicDataSource(); basicDataSource.setDriverClassName("com.mysql.jdbc.Driver"); basicDataSource.setUsername("root"); basicDataSource.setPassword("situ123"); </pre>

```
basicDataSource.setUrl("jdbc:mysql://127.0.0.1/simert");  
dataSource = basicDataSource;  
}
```

Como se muestra en la (Tabla XXXVI), este código permite establecer una conexión con la base de datos. Primeramente se ha instalado y configurado las librerías apache commons-dbcp, la librería commons-pool y el driver de nuestra base de datos mysql-connector-java-5.1.6.jar.

Una vez configurado todo, para usar *BasicDataSource* no tenemos más que hacer un *new* de esa clase y pasarle los parámetros adecuados de nuestra conexión con los métodos *set ()* disponibles para ello.

Con *setDriverClassName ()* el nombre de la clase que es el driver de nuestra conexión a la base de datos, MySQL en nuestro ejemplo

Con *setUsername()* un nombre de usuario válido para la base de datos (root).

Con *setPassword ()* la password de base de datos para dicho usuario (situ123).

Con *setUrl ()* la cadena de conexión con la base de datos, en nuestro caso en el mismo ordenador en que va a correr nuestro ejecutable (localhost o 127.0.0.1) y el nombre de la base de datos (SIMERT).

3.3. Simulación en el Packet Tracer de la arquitectura cliente – servidor para la conectividad de la red de datos del SIMERT.

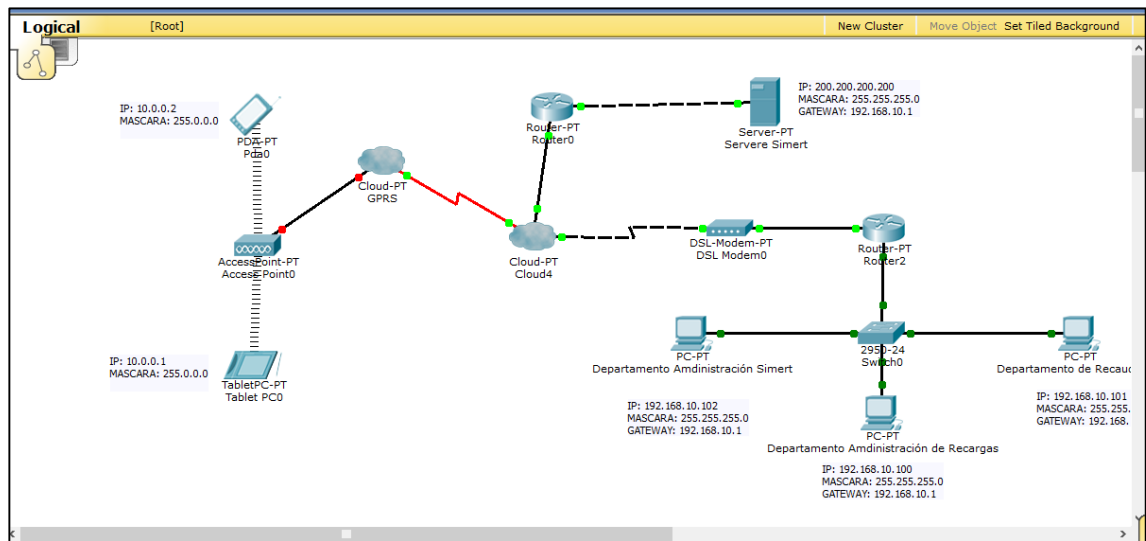


Figura 39: Arquitectura Cliente-Servidor para la conectividad de la red de datos del SIMERT en el Packet Tracer.

3.3.1. Realizar el subneteo.

El subneteo nos permite una mejor administración, control del tráfico y seguridad al segmentar la red por función también reduce el tráfico de broadcast de nuestra red. Para ello vamos a segmentar nuestra red.

TABLA XXXVII: REALIZAR EL SUBNETEO

IP CLASE C	192.168.1.0
MASCARA:	255.255.255.0
1 RED	32 Host

TABLA XXXVIII: CANTIDAD DE BITS

Bits	512	256	128	64	32	16	8	4	2	1
N	9	8	7	6	5	4	3	2	1	0

Procedimiento.

$$\text{Formula: } 2^n - 2$$

$$2^6 - 2 = \text{red}$$

$$64 - 2 = 62 \text{ hosts}$$

Tabla XXXIX: DESCRIPCIÓN DE SUBNETEO

Red(R,R,R,H)	192	168	1	0
Mascara	255	255	255	0
Octetos	-----	-----	-----	-----
Octetos	11111111	11111111	11111111	11000000
Mascara	255	255	255	192
Sub red de 32 Hosts	192.168.1.0 (Red)	192.168.1.63 (Broadcast)		
Rango de IP	192.168.1.1	192.168.1.62		

3.3.2. Asignar los Ip's para las computadoras del SIMERT.

TABLA XL: ASIGNACIÓN DE IPS

DESCRIPCIÓN	PC	DIRECCIÓN	MASCARA	GATEWAY
Administración Recargas	Pc-3	192.168.10.10 0	255.255.255. 0	192.168.10.1
Recaudación	Pc-2	192.168.1.101	255.255.255. 0	192.168.10.1
Administración SIMERT	Pc-1	192.168.10.10 2	255.255.255. 0	192.168.10.1

4. FASE 4: MEDICIÓN DE RESULTADOS.

En esta última fase es necesario realizar una evaluación del trabajo realizado en el SIMERT, se evaluará la utilización de los servicios y por defecto se estará evaluando la conformidad, la aceptación por parte de los usuarios hacia la nueva implementación, realizando las respectivas pruebas de funcionalidad y verificar que la infraestructura HW/SW permita un correcto flujo de datos entre todas las áreas del SIMERT. Se ha realizado las pruebas de carga al servidor (Ver sección Resultados, Pág. 108, apartado 4.1.1. Pruebas de carga al servidor). Se ha realizado las respectivas pruebas a la aplicación sockets “SERVER-SIMERT” como: Pruebas Unitarias (Ver sección Resultados, Pág. 111, apartado 4.2.1. Pruebas Unitarias), Pruebas de Validación (Ver sección Resultados, Pág. 116, apartado 4.2.2. Pruebas de Validación) y Pruebas de accesibilidad, funcionalidad y usabilidad (Ver sección Resultados, Pág. 120, apartado 4.2.3. Prueba de Accesibilidad, funcionalidad y usabilidad).

4.1. PRUEBAS DEL SERVIDOR

Se realizaron algunas pruebas al servidor que se especifican a continuación, para verificar el respectivo funcionamiento.

4.1.1. Pruebas de carga al servidor

Las pruebas de carga o de estrés nos han ayudado a determinar el comportamiento del sistema bajo un nivel de exigencia mayor al que es capaz de manejar, también determinaremos la capacidad, la velocidad de respuesta y conoceremos los límites bajo los cuales nuestra aplicación se comportará de manera apropiada.

Siege es una alternativa libre que proporciona ciertas funcionalidades. Una prueba de carga se define como el proceso que se le impone a un sistema basado en una cantidad predefinida de peticiones o procedimientos.

TABLA XLI: COMANDO DE INSTALACION DE SIEGE

```
situ@simertnet:~$ sudo apt-get install siege
```

Ejecución de Carga con Siege.

Para la ejecución de carga al servidor usaremos el siguiente comando:

TABLA XLII: COMANDO DE EJECUCION DE SIEGE

```
situ@simertnet:~$ sudo siege -b 5 -c 50 -r 500  
http://localhost:8080/simertweb/
```

*Figura 40: Comando de ejecución de Carga con Siege.*

Descripción del comando:

- siege: nos permite hacer uso de la herramienta.
- -t especificamos el tiempo que tomará la prueba.
- -c especificamos la cantidad de conexiones concurrentes.
- -b obliga a no tener ningún retraso entre cada usuario simulado (modo benchmark).
- -q elimina la salida resultante de cada petición que va mostrando el proceso durante la prueba.
- -r especifica el número de repeticiones de la prueba
- Finalmente colocamos la dirección que deseamos probar.

El comando hará 50 conexiones con unas 500 repeticiones continuas una tras otra con un retraso de 5 segundos por petición. El resultado será un archivo estático test.html.

```

root@simertnet: /home/situ
File Edit View Terminal Help
HTTP/1.1 200 0.31 secs: 1487 bytes ==> /simertweb/
HTTP/1.1 200 0.30 secs: 1487 bytes ==> /simertweb/
HTTP/1.1 200 0.23 secs: 1487 bytes ==> /simertweb/
HTTP/1.1 200 0.28 secs: 1487 bytes ==> /simertweb/
HTTP/1.1 200 0.26 secs: 1487 bytes ==> /simertweb/
HTTP/1.1 200 0.29 secs: 1487 bytes ==> /simertweb/
HTTP/1.1 200 0.12 secs: 1487 bytes ==> /simertweb/
HTTP/1.1 200 0.21 secs: 1487 bytes ==> /simertweb/
HTTP/1.1 200 0.15 secs: 1487 bytes ==> /simertweb/
HTTP/1.1 200 0.11 secs: 1487 bytes ==> /simertweb/
HTTP/1.1 200 0.10 secs: 1487 bytes ==> /simertweb/
HTTP/1.1 200 0.11 secs: 1487 bytes ==> /simertweb/
HTTP/1.1 200 0.13 secs: 1487 bytes ==> /simertweb/
HTTP/1.1 200 0.08 secs: 1487 bytes ==> /simertweb/
HTTP/1.1 200 0.08 secs: 1487 bytes ==> /simertweb/
HTTP/1.1 200 0.08 secs: 1487 bytes ==> /simertweb/
HTTP/1.1 200 0.06 secs: 1487 bytes ==> /simertweb/
HTTP/1.1 200 0.03 secs: 1487 bytes ==> /simertweb/
HTTP/1.1 200 0.04 secs: 1487 bytes ==> /simertweb/
HTTP/1.1 200 0.03 secs: 1487 bytes ==> /simertweb/
done.
Transactions:          25000 hits
Availability:          100.00 %
Elapsed time:          432.31 secs
Data transferred:     35.45 MB
Response time:         0.86 secs
Transaction rate:     57.83 trans/sec
Throughput:            0.08 MB/sec
Concurrency:           49.85
Successful transactions: 25000
Failed transactions:   0
Longest transaction:  1.50
Shortest transaction: 0.03

FILE: /var/siege.log
You can disable this annoying message by editing
the .siegerc file in your home directory; change
the directive 'show-logfile' to false.
root@simertnet:/home/situ#
    
```

Figura 41: Resultados de la prueba de carga con Siege.

Como resultado final nos brindara un reporte de las principales características de la prueba de carga, a continuación detallamos los resultados en la Tabla XLIII.

TABLA XLIII: DESCRIPCIÓN DE RESULTADOS DE CARGA CON SIEGE

Transaccions	25000	Transaccions
Availability	100.00%	Disponibilidad
Elapsed time	432.31 secs	El tiempo transcurrido
Response Time	0.86 secs	Tiempo de Respuesta
Data transfered	35.45 MB	Los datos transferidos
Transaction rate	57.83 trans/sec	tasa de transacción
Throughput	0.08 Mb/sec	Rendimiento

Concurrency	49.85	Concurrencia
Successful transactions	25000	Éxito de transacciones
Failed transactions	0	transacciones fallidas
Longest Transaction	1.50	Transacción Mayor
Shortest Transaction	0.03	Transacción corta

4.2. PRUEBAS DE LA APLICACIÓN SOCKETS “SERVER-SIMERT”

Se realizaron las pruebas de cada una de las tareas de la aplicación sockets de nominada “SERVER-SIMERT”, para verificar el respectivo funcionamiento de acuerdo a los requerimientos establecidos e identificar fallas que al momento de la codificación no han podido ser encontradas.

4.2.1. Pruebas Unitarias

Las pruebas unitarias han sido realizadas mediante la creación de un proyecto de pruebas denominado SERVER-SIMERT (Test) el cual es de tipo JAVA Test Project y la utilización de JUnit 4.x [10], dentro de él se ha creado una clase por cada componente a ser evaluado.

Los componentes evaluados son:

4.2.1.1. Validación del acceso a la aplicación “SERVER-SIMERT” a través del nombre de usuario y contraseña.

El código fuente del test realizado para evaluar el componente Validación del acceso a la aplicación “SERVER-SIMERT” a través del nombre de usuario y contraseña se muestra en la (Tabla XLIV).

TABLA XLIV: CÓDIGO FUENTE DEL MÉTODO
“TESTVALIDARUSUARIOCONTRASEN”

<p>CÓDIGO FUENTE PARA LA CLASE INGRESODATOSUSUARIOTEST DEL MÉTODO TESTCONFIRMATRAMA1</p>
<pre>public class IngresoDatosUsuarioTest { /**</pre>

```
* Test of validarUsuarioContrasen method,  
* of class IngresoDatosUsuario.  
*/  
@Test  
public void testValidarUsuarioContrasen() {  
    System.out.println("validarUsuarioContrasen");  
    String login = "juvamogu1989";  
    String login1 = "0704389675";  
    IngresoDatosUsuario instance = new IngresoDatosUsuario();  
    String expectedResult = "0";  
    String result = instance.validarUsuarioContrasen(login,login1);  
    assertEquals(expectedResult, result);  
}  
}
```

Al ejecutar el Test si el nombre del usuario almacenado en la variable **login** y la contraseña almacenada en la variable **login1** es correcta, el valor dado en la variable **expResult** será igual al obtenido en el método **validarUsuarioContrasen**, presentando como respuesta de aprobación una barra de color verde tal como se visualiza en la (figura 42).

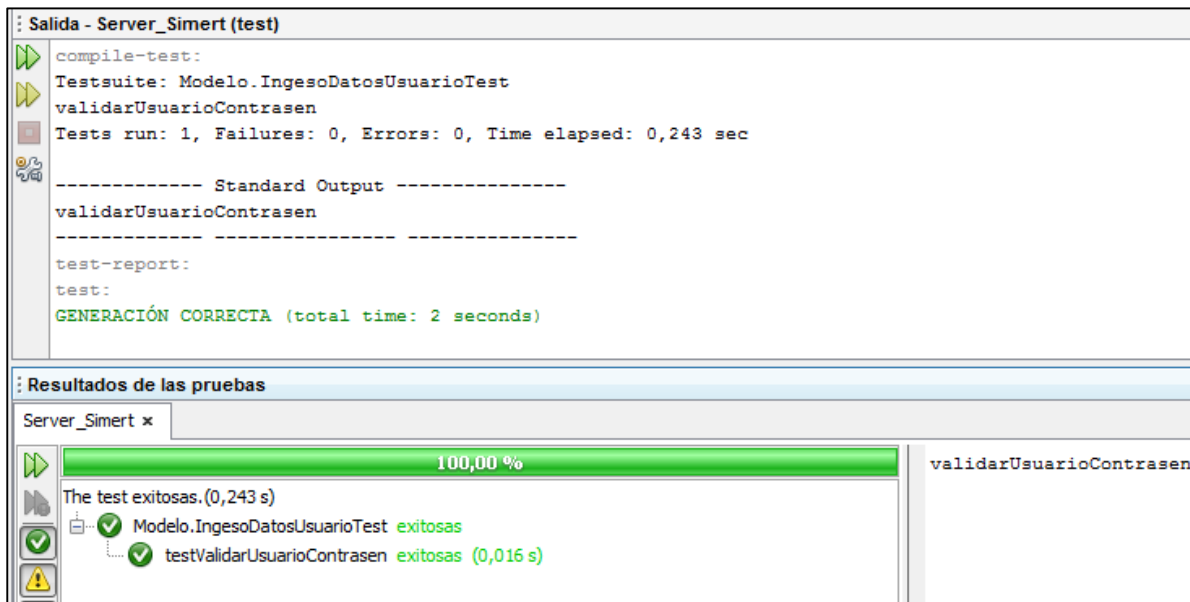


Figura 42: Respuesta de Aprobación para el acceso a la aplicación "SERVER-SIMERT".

Al ejecutar el Test si el nombre del usuario almacenado en la variable **login** y la contraseña almacenada en la variable **login1** es incorrecta, el valor dado en la variable **expResult** no será igual al obtenido en el método **validarUsuarioContrasen**, presentando como respuesta de aprobación una barra de color rojo tal como se visualiza en la (figura 43).

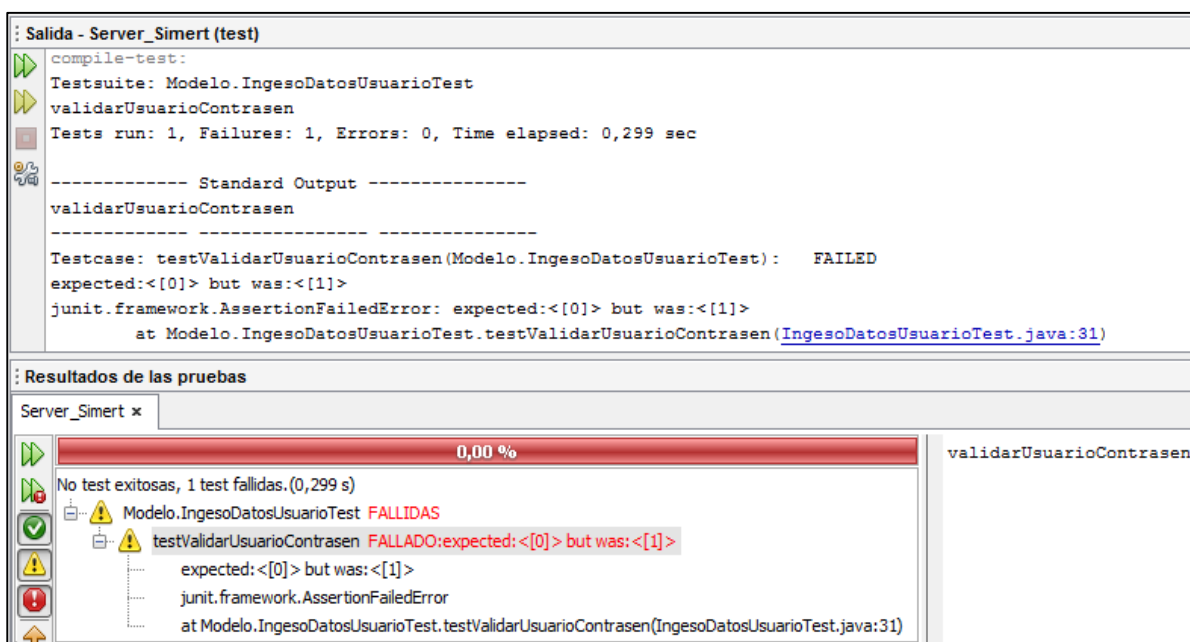


Figura 43: Respuesta de error para el acceso a la aplicación "SERVER-SIMERT".

4.2.1.2. Validación de la trama “Registro enviar zonas”

El código fuente del test realizado para evaluar el componente Validación de la trama del Registro enviar zonas a los controladores se muestra en la Tabla XLV.

TABLA XLV: CÓDIGO FUENTE PARA REALIZAR EL TESTCONFIRMATRAMA1.

CÓDIGO FUENTE PARA LA CLASE CONTROLADORTEST DEL MÉTODO TESTCONFIRMATRAMA1
<pre> public class ControladorTest { /** * Test of confirmaTrama1 method, of class Controlador. */ @Test public void testConfirmaTrama1() { System.out.println("confirmaTrama1"); String msj2 = "##;001;mariuxi;12345;98:52:B1:F8:70:BB;**"; Controlador instance = null; String expectedResult = "si";//resultado esperado String result = instance.confirmaTrama1(msj2); assertEquals(expectedResult, result); } } </pre>

Al ejecutar el Test si la trama almacenada en la variable msj2 es correcta, el valor dado en la variable **expResult** será igual al obtenido en el método **confirmaTrama1**, presentando como respuesta de aprobación una barra de color verde tal como se visualiza en la (figura 44).

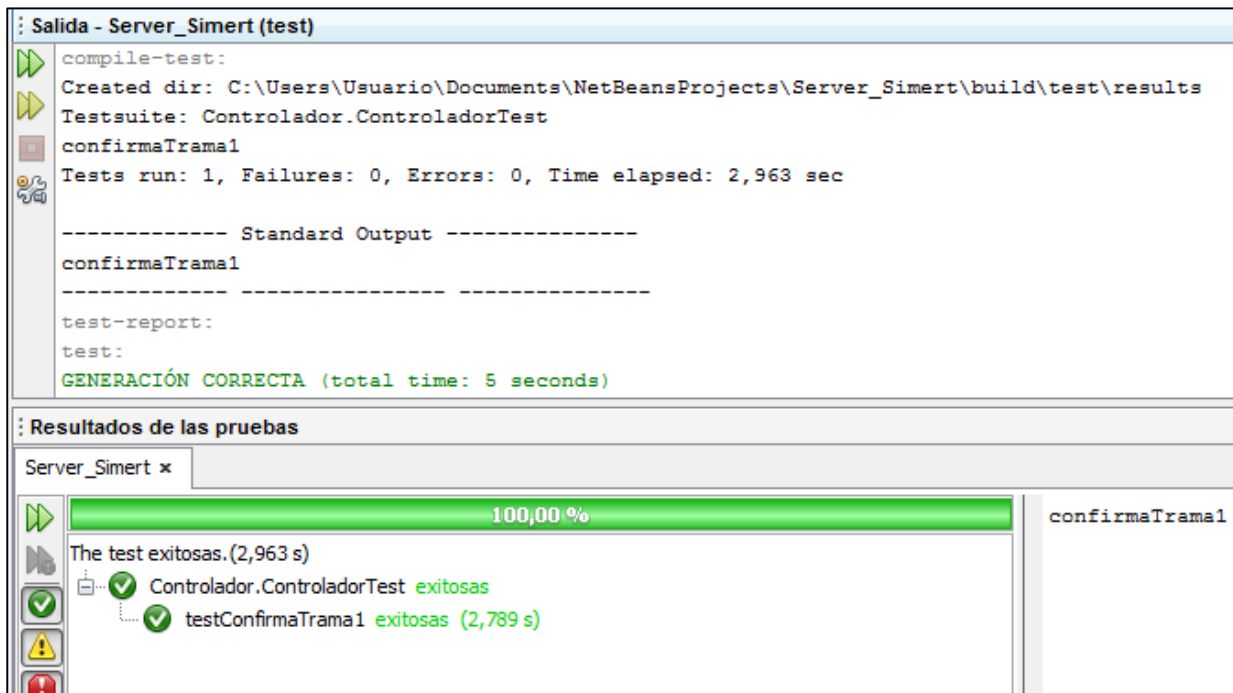


Figura 44: Respuesta de Aprobación de la trama "Registro enviar zonas de estacionamiento".

Al ejecutar el Test si la trama almacenada en la variable `msj2` es incorrecta, el valor dado en la variable `expResult` no será igual al obtenido en el método `confirmaTrama1`, presentando como respuesta de aprobación una barra de color rojo tal como se visualiza en la (figura 45).

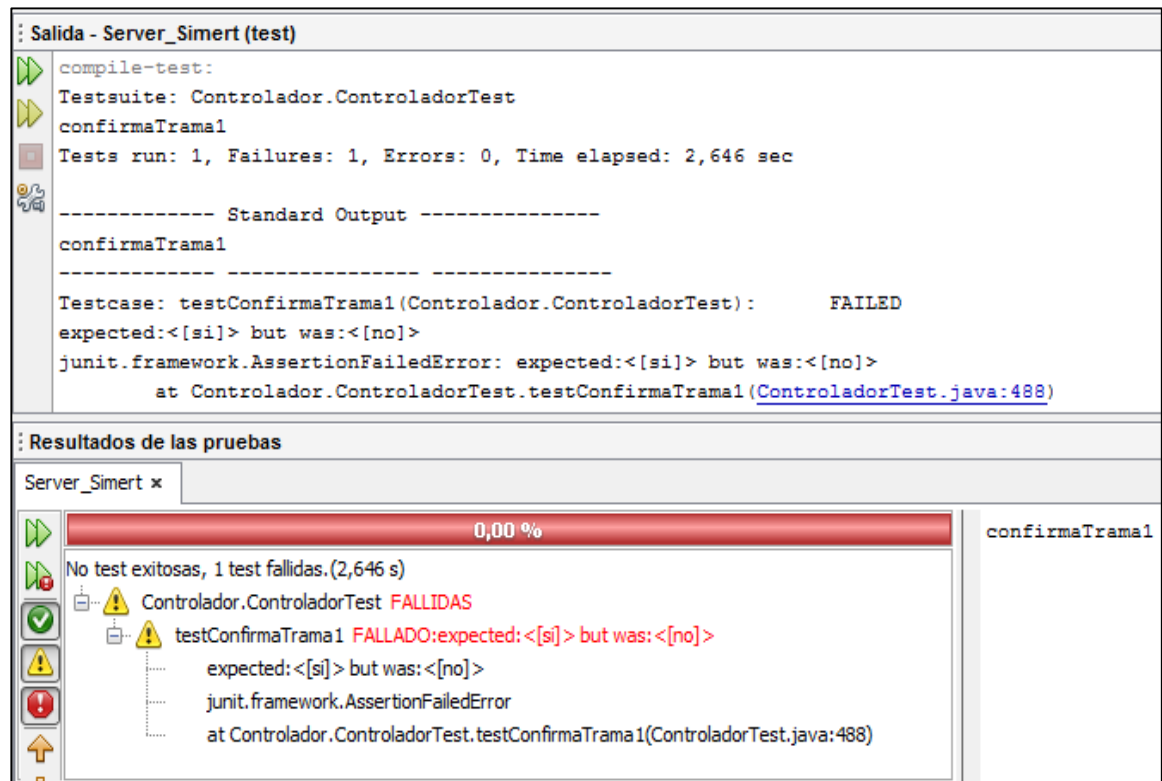


Figura 45: Respuesta de error de la trama "Registro enviar zonas de estacionamiento".

4.2.2. Pruebas de Validación.

Estas pruebas han sido dirigidas a los usuarios finales de la aplicación es decir al administrador del servidor del SIMERT con el fin de verificar que cumpla con los requisitos previstos y comprobar que sea accesible así como de fácil uso.

Para ello se realizó previamente un plan de pruebas que se muestran a continuación en las siguientes tablas donde se visualiza las tareas que se planificaron y evaluar el resultado obtenido.

4.2.2.1. Iniciar Servidor.

En la (Tabla XLVI) se ha descrito el procedimiento para comprobar la tarea Iniciar Servidor a partir del ingreso del nombre de usuario y contraseña del administrador del servidor.

TABLA XLVI: CASO DE PRUEBA - INICIAR SERVIDOR

Caso de Prueba: Iniciar Servidor
Descripción: El administrador del servidor ingresa el nombre de usuario y contraseña para acceder a la aplicación “SERVER-SIMERT”
Condiciones de la ejecución: Previamente los datos del administrador deben estar registrados en el fichero denominado “autenticación”.
Descripción del procedimiento de prueba: <ul style="list-style-type: none"> • El administrador del servidor ejecuta la aplicación • En la interfaz “Iniciar Servidor” ingresa el nombre de usuario y contraseña • Presiona el botón “Iniciar”. • Si los datos son incorrectos presenta un cuadro de diálogo como advertencia. • Si los datos son correctos accede a la aplicación, visualizando la interfaz “Servidor SIMERT”
Resultado esperado: Acceder a la aplicación
Evaluación de la prueba: Se cumple
Incidentes de prueba: no
Resumen de la prueba: El administrador del servidor pudo acceder a la aplicación al ingresar los datos correctamente.

4.2.2.2. Cambiar Nombre de Usuario y Contraseña.

En la (Tabla XLVII) se ha descrito el procedimiento a seguir por el administrador del servidor para efectuar la tarea Cambiar Nombre de Usuario y Contraseña y el resultado obtenido a partir de este caso de prueba.

TABLA XLVII: CASO DE PRUEBA - CAMBIAR NOMBRE DE USUARIO Y CONTRASEÑA

Caso de Prueba: Cambiar Nombre de Usuario y Contraseña
Descripción: El administrador del servidor ingresa el nombre de usuario y contraseña para acceder a la aplicación “SERVER-SIMERT”
Condiciones de la ejecución: Previamente los datos del administrador deben estar registrados en el fichero denominado “autenticación”.
Descripción del procedimiento de prueba: <ul style="list-style-type: none"> • El administrador del servidor ejecuta la aplicación

<ul style="list-style-type: none"> • En la interfaz “Iniciar Servidor” ingresa el nombre de usuario y contraseña • Presiona el botón “Cambiar Contraseña”. • Si los datos son incorrectos presenta un cuadro de diálogo como advertencia. • Si los datos son correctos accede a la aplicación, visualizando la interfaz “Cambiar Usuario y Contraseña” • Ingresa el nombre de usuario y contraseña que desea cambiar, los cuales deben ser mayores a 8 caracteres. • Presiona el botón “Cambiar” • Si los datos son menores a 8 caracteres presenta un cuadro de diálogo como advertencia. • Si los datos ingresados son correctos visualiza la interfaz “Iniciar Servidor” para que acceda a la aplicación con los nuevos datos.
Resultado esperado: Cambiar nombre de usuario y contraseña del administrador del servidor.
Evaluación de la prueba: Se cumple
Incidentes de prueba: no
Resumen de la prueba: El administrador del servidor pudo cambiar su nombre de usuario y contraseña para acceder nuevamente a la aplicación

4.2.2.3. Salir del Servidor

En la (Tabla XLVIII) se ha descrito el procedimiento a seguir por el administrador del servidor para efectuar la tarea Salir del Servidor y el resultado obtenido a partir de este caso de prueba.

TABLA XLVIII: CASO DE PRUEBA - SALIR DEL SERVIDOR

Caso de Prueba: Salir del Servidor
Descripción: El administrador del servidor accede a la aplicación “SERVER-SIMERT” visualizando la interfaz “Servidor SIMERT”
Condiciones de la ejecución: Previamente el administrador del servidor debe haber accedido a la aplicación.
Descripción del procedimiento de prueba: <ul style="list-style-type: none"> • El administrador accede a la aplicación del servidor y visualiza la interfaz “Servidor SIMERT”.

<ul style="list-style-type: none"> • Presiona el botón “Salir”. • Visualiza la interfaz “Salir del Servidor”. • Ingresa el nombre de usuario y contraseña. • Presiona el botón “Salir”. • Si los datos son incorrectos presenta un cuadro de diálogo como advertencia. • Si los datos son correctos cierra la aplicación “SERVER-SIMERT” y la comunicación con los clientes.
Resultado esperado: Cierra la aplicación sockets denominada “SERVER-SIMERT”
Evaluación de la prueba: Se cumple
Incidentes de prueba: no
Resumen de la prueba: El administrador del servidor cierra la aplicación sockets ingresando correctamente el nombre de usuario y contraseña.

4.2.2.4. Visualización de las tramas recibidas acerca de los registros de estacionamiento vehicular del SIMERT.

En la (Tabla XLIX) se ha descrito el procedimiento para la visualización de las tramas de los diferentes registros de estacionamiento vehicular que son enviados por los clientes que son los controladores del SIMERT, obteniendo el resultado a partir de este caso de prueba.

TABLA XLIX. CASO DE PRUEBA - VISUALIZACIÓN DE LAS TRAMAS RECIBIDAS ACERCA DE LOS REGISTROS DE ESTACIONAMIENTO VEHICULAR DEL SIMERT.

Caso de Prueba: Visualización de las tramas recibidas acerca de los registros de estacionamiento vehicular del SIMERT.
Descripción: El administrador del servidor accede a la aplicación “SERVER-SIMERT” visualizando la interfaz “Servidor SIMERT”
Condiciones de la ejecución: Previamente el administrador del servidor debe haber accedido a la aplicación.
Descripción del procedimiento de prueba: <ul style="list-style-type: none"> • El administrador accede a la aplicación del servidor y visualiza la interfaz “Servidor SIMERT”. • Visualiza a través del textArea las conexiones de los clientes que son los controladores del SIMERT junto con la hora, fecha, dirección IP y mac del

dispositivo Tablet durante el acceso al servidor, así como también las tramas de cada uno de los registros de estacionamiento vehicular y los mensajes de los procesos que se realizan internamente con la información.
Resultado esperado: Visualizar cada una de las tramas de los registros de estacionamiento vehicular que son enviadas por los controladores del SIMERT.
Evaluación de la prueba: Se cumple
Incidentes de prueba: no
Resumen de la prueba: El administrador del servidor visualiza toda la información de los registros de estacionamiento vehicular enviadas por los controladores del SIMERT a través de tramas.

4.2.3. Prueba de Accesibilidad, funcionalidad y usabilidad.

Esta prueba fue aplicada a la persona encargada de manipular el sistema y administrar el servidor, además el coordinador del Sistema Municipal de Estacionamiento emitió un certificado luego de haber cumplido con las pruebas de funcionalidad (Ver Anexo 14).

4.2.3.1. Análisis de Prueba: Administrador del Servidor.

Una vez realizado el análisis en base a la información recolectada en la encuesta aplicada a la Ing. Tania Verónica Vásquez Romero encargada de la técnica de la Unidad de Tránsito (Ver Anexo 16), se ha obtenido los siguientes resultados:

1. ACCESIBILIDAD.

1.1. ¿Tuvo algún inconveniente al ejecutar la aplicación sockets “SERVER-SIMERT”?

Análisis Cuantitativo

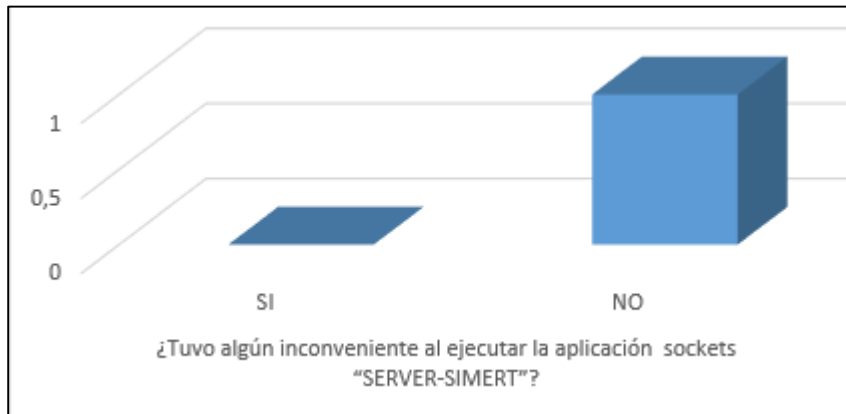


Figura 46: Análisis de la ejecución de la aplicación sockets “SERVER-SIMERT”.

Interpretación de Resultados:

Como se observa en la (figura 46), la Ingeniera encargada de la Técnica de la Unidad de Tránsito manifestó que no tuvo problema alguno al ejecutar la aplicación sockets “SERVER-SIMERT” porque le resulta muy sencilla su ejecución.

1.2. ¿Tuvo algún inconveniente al ingresar y cerrar la aplicación sockets “SERVER-SIMERT” con el nombre de usuario y contraseña?

Análisis Cuantitativo

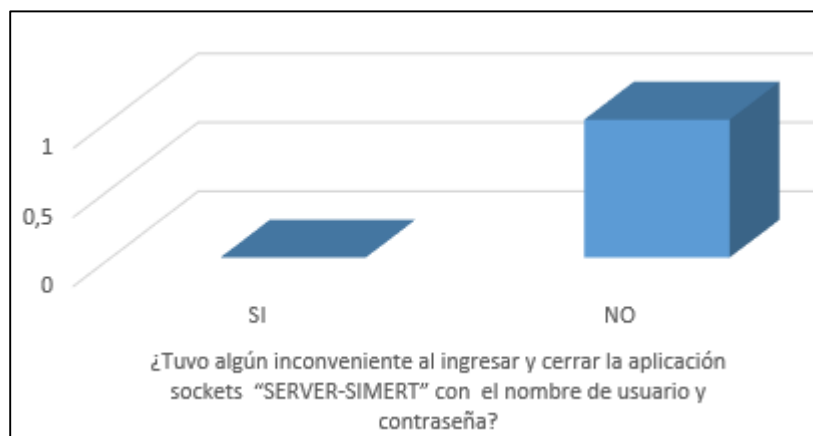


Figura 47: Análisis del acceso y finalización de la aplicación sockets “SERVER-SIMERT”

Interpretación de Resultados:

Como se muestra en la (figura 47), la Ingeniera encargada de la Técnica de la Unidad de Tránsito manifestó que no tuvo ningún problema al ingresar y cerrar la aplicación sockets “SERVER-SIMERT” porque los componentes como los cuadros de diálogos que se muestran cuando existe alguna irregularidad le resulta comprensible.

2. FUNCIONALIDAD.

2.1. ¿Presentó algún error la aplicación sockets “SERVER-SIMERT” durante el envío y recepción de los siguientes registros de estacionamiento vehicular?

- Registro enviar zonas a los controladores.

Análisis Cuantitativo.

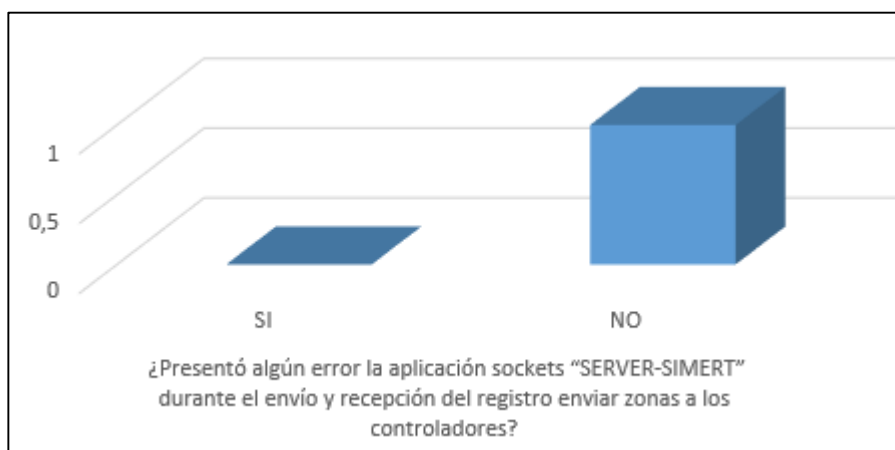


Figura 48: Análisis del envío y recepción del registro "Enviar zonas a los controladores".

Interpretación de Resultados:

Como se muestra en la (figura 48), la Ingeniera encargada de la Técnica de la Unidad de Tránsito manifestó que no hubo error alguno durante el envío y recepción de la trama del registro enviar zonas a los controladores porque se visualizó correctamente los datos enviados por el controlador.

- Registro ingresar cliente exonerado o recarga.

Análisis Cuantitativo.

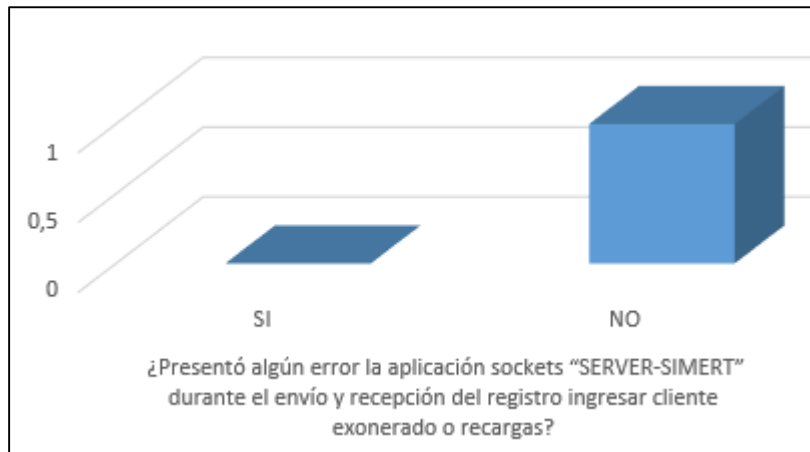


Figura 49: Análisis del envío y recepción del registro "Ingresar cliente exonerado o recarga".

Interpretación de Resultados:

La Ingeniera encargada de la Técnica de la Unidad de Tránsito Como se observa en la (figura 49), manifestó que no hubo error alguno durante el envío y recepción de la trama del registro ingresar cliente exonerado o recarga porque se visualizó correctamente los datos enviados por el controlador.

- Registro ingresar multa por recarga.

Análisis Cuantitativo.

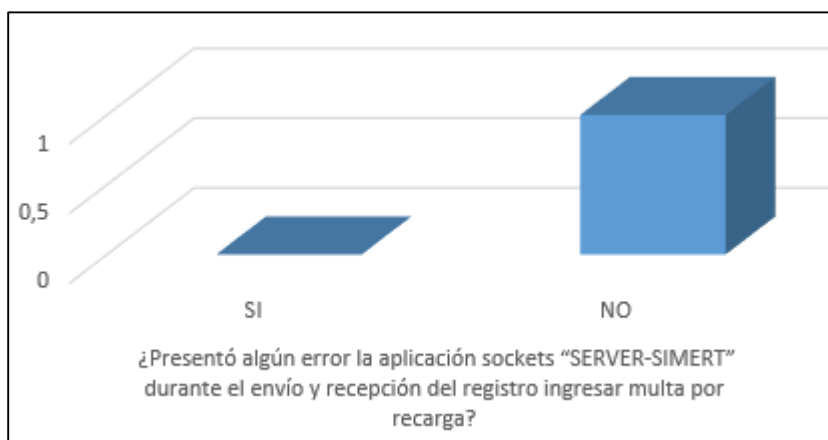


Figura 50: Análisis del envío y recepción del registro "Ingresar multa por recarga".

Interpretación de Resultados:

Como se observa en la (figura 50), la Ingeniera encargada de la Técnica de la Unidad de Tránsito manifestó que no hubo error alguno durante el envío y recepción de la trama del registro ingresar multa por recarga porque se visualizó los datos enviados por el controlador.

- Registro validar código tarjeta.

Análisis Cuantitativo.

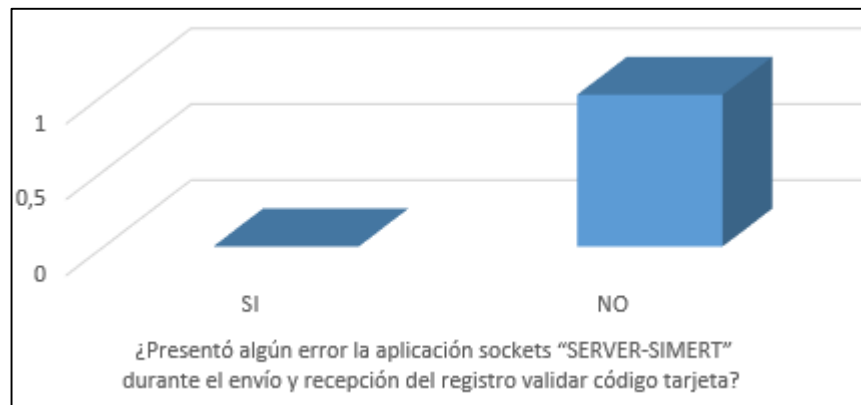


Figura 51: Análisis del envío y recepción del registro "Validar código de tarjeta".

Interpretación de Resultados:

Como se observa en la (figura 51), la Ingeniera encargada de la Técnica de la Unidad de Tránsito manifestó que no hubo error alguno durante el envío y recepción de la trama del registro validar código tarjeta porque se visualizó correctamente los datos enviados por el controlador.

- Registro modificar espacios de tarjeta.

Análisis Cuantitativo.

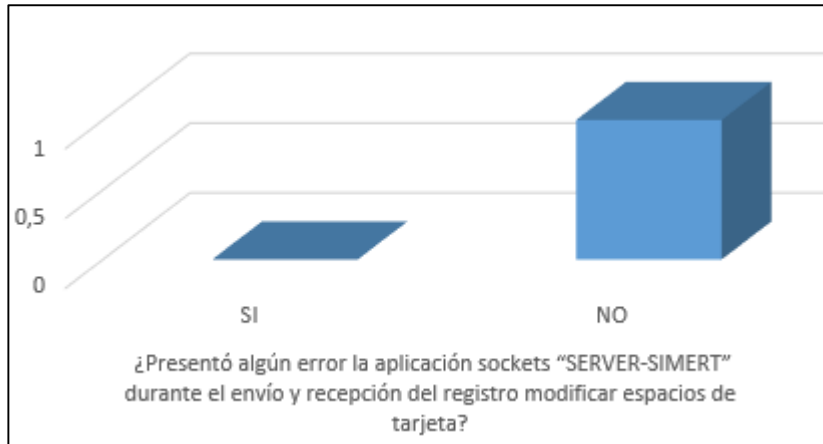


Figura 52: Análisis del envío y recepción del registro "Modificar espacios de tarjeta".

Interpretación de Resultados:

Como se observa en la (figura 52), la Ingeniera encargada de la Técnica de la Unidad de Tránsito manifestó que no hubo error alguno durante el envío y recepción de la trama del registro modificar espacios de tarjeta porque se visualizó correctamente los datos enviados por el controlador.

- Registro ingresar personal del municipio.

Análisis Cuantitativo

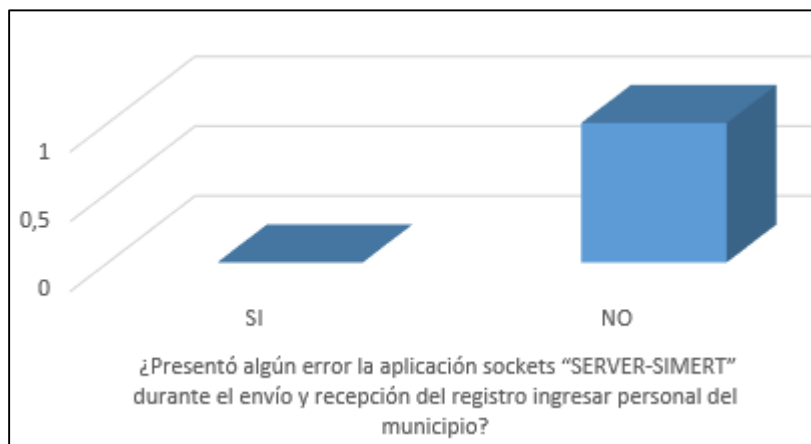


Figura 53: Análisis del envío y recepción del registro "Ingresar personal del municipio".

Interpretación de Resultados:

Como se observa en la (figura 53), la Ingeniera encargada de la Técnica de la Unidad de Tránsito manifestó que no hubo error alguno durante el envío y recepción de la trama del registro ingresar personal del municipio porque se visualizó correctamente los datos enviados por el controlador.

- **Registro ingresar multa por tarjeta**

Análisis Cuantitativo.

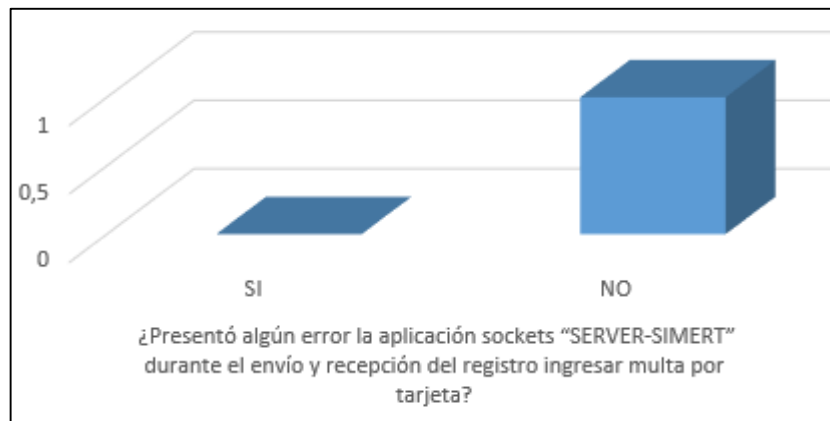


Figura 54: Análisis del envío y recepción del registro "Ingresar multa por tarjeta".

Interpretación de Resultados:

Como se observa en la (figura 54), la Ingeniera encargada de la Técnica de la Unidad de Tránsito manifestó que no hubo error alguno durante el envío y recepción de la trama del registro ingresar multa por tarjeta porque se visualizó correctamente los datos enviados por el controlador.

- Registro ingresar código de tarjeta.

Análisis Cuantitativo

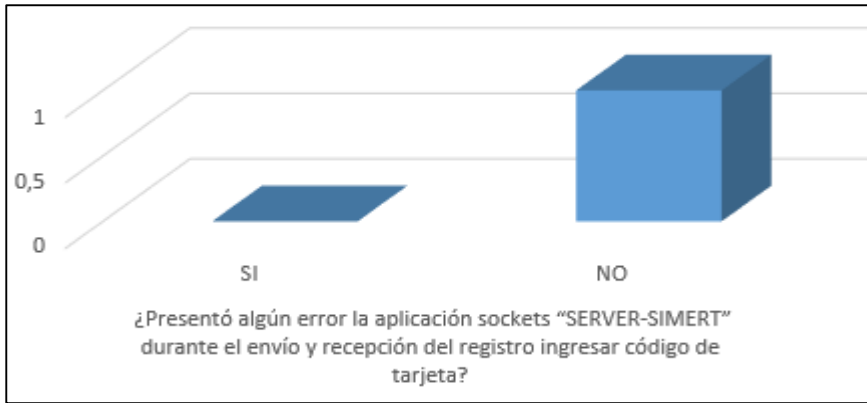


Figura 55: Análisis del envío y recepción del registro "Ingresar código de tarjeta".

Interpretación de Resultados:

Como se observa en la (figura 55), La Ingeniera encargada de la Técnica de la Unidad de Tránsito manifestó que no hubo error alguno durante el envío y recepción de la trama del registro ingresar código de tarjeta porque se visualizó correctamente los datos enviados por el controlador.

- Registro ingresar nueva tarjeta.

Análisis Cuantitativo

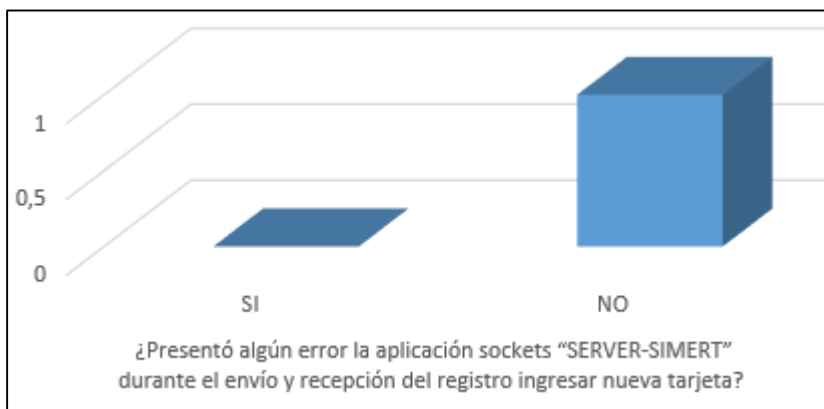


Figura 56: Análisis del envío y recepción del registro "Ingresar nueva tarjeta".

Interpretación de Resultados:

Como se observa en la (figura 56), la Ingeniera encargada de la Técnica de la Unidad de Tránsito manifestó que no hubo error alguno durante el envío y recepción de la trama del registro ingresar nueva tarjeta porque se visualizó correctamente los datos enviados por el controlador.

- **Registro ingresar cliente recarga estacionado.**

Análisis Cuantitativo

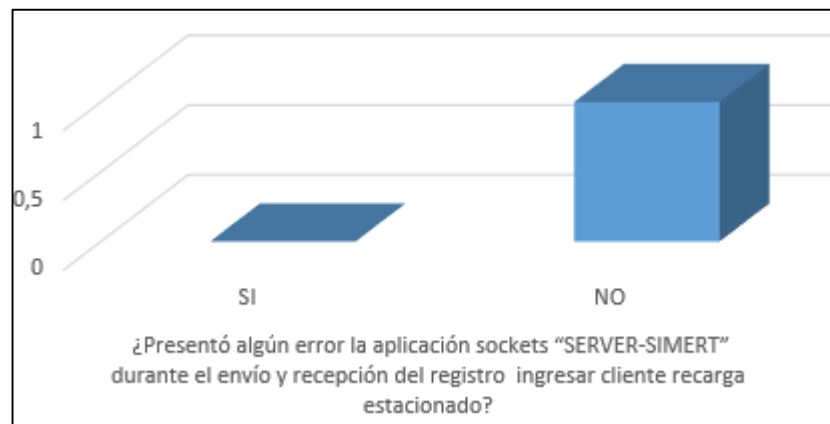


Figura 57: Análisis del envío y recepción del registro "Ingresar cliente recarga estacionado".

Interpretación de Resultados:

Como se observa en la (figura 57), la Ingeniera encargada de la Técnica de la Unidad de Tránsito manifestó que no hubo error alguno durante el envío y recepción de la trama del registro ingresar cliente recarga estacionado porque se visualizó correctamente los datos enviados por el controlador.

- **Registro ingresar datos posicionales.**

Análisis Cuantitativo

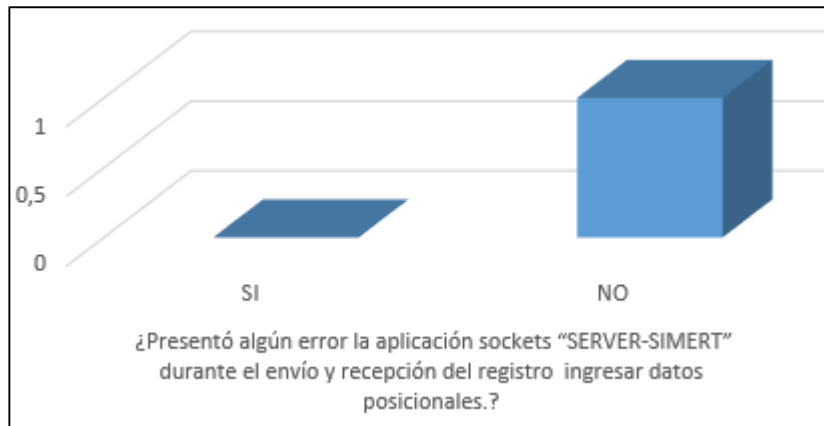


Figura 58: Análisis del envío y recepción del registro "Ingresar datos posicionales".

Interpretación de Resultados:

Como se observa en la (figura 58), la Ingeniera encargada de la Técnica de la Unidad de Tránsito manifestó que no hubo error alguno durante el envío y recepción de la trama del registro ingresar datos posicionales porque se visualizó correctamente los datos enviados por el controlador.

- **Registro eliminar controlador de sesiones.**

Análisis Cuantitativo

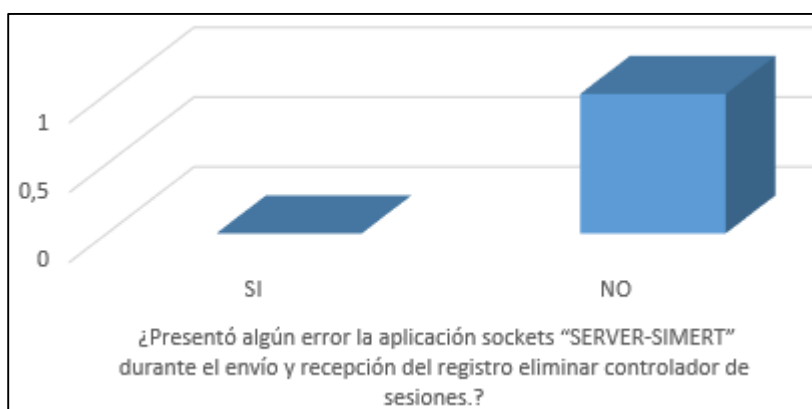


Figura 59: Análisis del envío y recepción del registro "Eliminar controlador de sesiones".

Interpretación de Resultados:

Como se observa en la (figura 59), la Ingeniera encargada de la Técnica de la Unidad de Tránsito manifestó que no hubo error alguno durante el envío y recepción de la trama

del registro eliminar controlador de sesiones porque se visualizó correctamente los datos enviados por el controlador.

- **Registro ingresar sincronización de multas.**

Análisis Cuantitativo

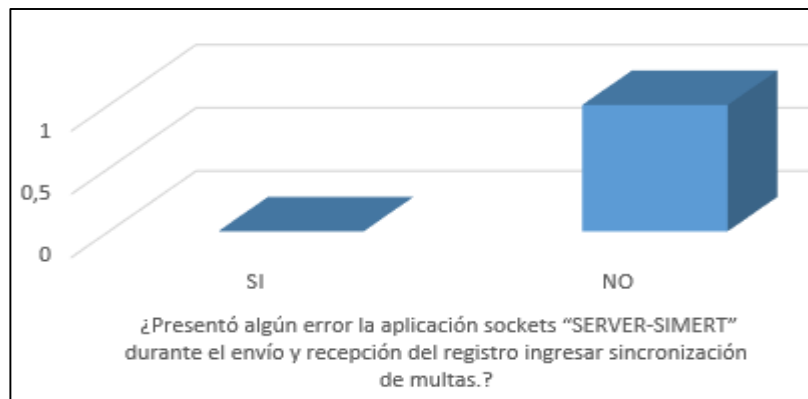


Figura 60: Análisis del envío y recepción del registro "Ingresar sincronización de multas".

Interpretación de Resultados:

Como se observa en la (figura 60), la Ingeniera encargada de la Técnica de la Unidad de Tránsito manifestó que no hubo error alguno durante el envío y recepción de la trama del registro ingresar sincronización de multas porque se visualizó correctamente los datos enviados por el controlador.

2.2. ¿Considera usted que la aplicación sockets “SERVER_SIMERT” es rápido en el envío y recepción de datos durante la comunicación con los clientes?

Análisis cuantitativo

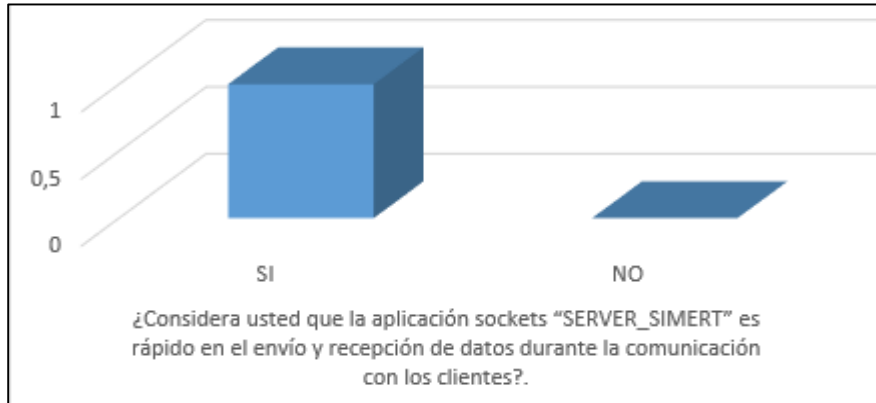


Figura 61: Análisis de la velocidad en la transmisión de datos

Interpretación de Resultados:

Como se observa en la (figura 61), la Ingeniera encargada de la Técnica de la Unidad de Tránsito manifestó que la aplicación es rápida en el envío y recepción de los datos durante la comunicación con los clientes que son los controladores del SIMERT porque se utiliza la clase ServerSockets y la tecnología GPRS bajo el protocolo TCP y direccionamiento IP como túnel de comunicación asegurando la fiabilidad y rapidez en la gestión de la información.

2.3. ¿La aplicación sockets “SERVER-SIMERT” cumple con todas las especificaciones requeridas manifestadas al inicio y durante la ejecución del proyecto?

Análisis Cuantitativo

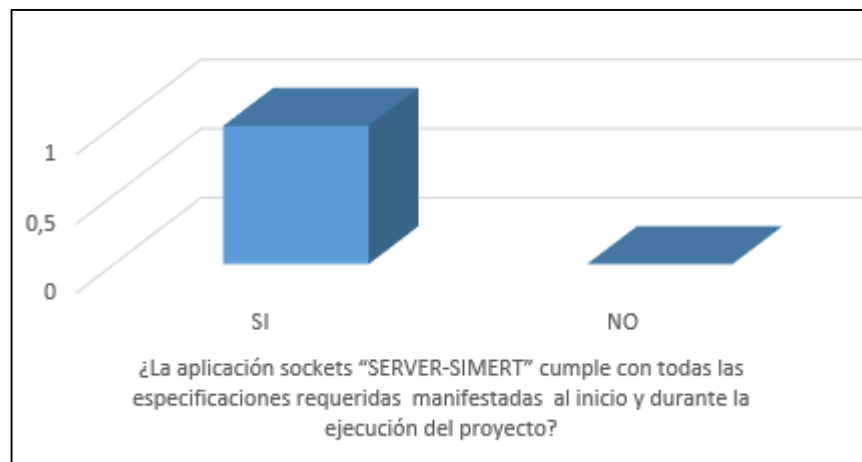


Figura 62: Análisis de los requerimientos de la aplicación sockets “SERVER-SIMERT”

Interpretación de Resultados:

Como se observa en la (figura 62), la Ingeniera encargada de la Técnica de la Unidad de Tránsito manifestó que la aplicación si cumple con las especificaciones requeridas manifestadas al inicio y durante la ejecución del proyecto porque se ha comprobado la conectividad entre los clientes que son los controladores del SIMERT y el servidor a través del envío y recepción de tramas de cada uno de los registros de estacionamiento vehicular, potenciando la seguridad y agilidad en la gestión de la información.

3. USABILIDAD

3.1. ¿Considera que la aplicación sockets “SERVER-SIMERT” tiene una interfaz amigable y de fácil uso?

Interpretación Cuantitativa

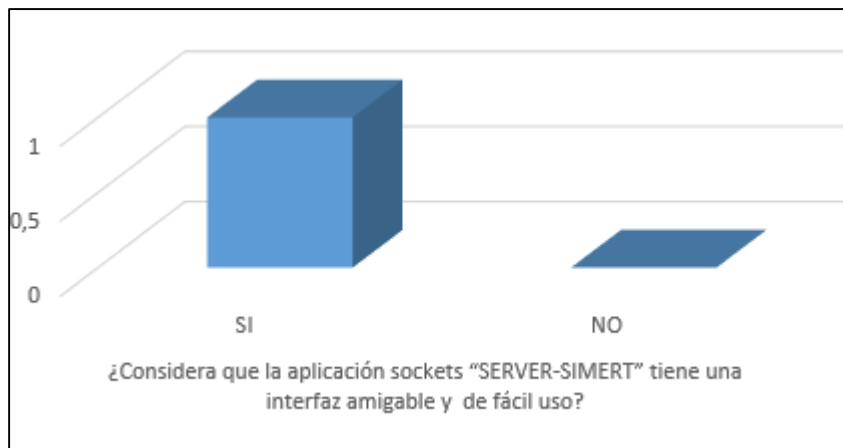


Figura 63: Análisis de la interfaz gráfica de la aplicación sockets “SERVER-SIMERT”.

Interpretación de Resultados:

Como se observa en la (figura 63), la Ingeniera encargada de la Técnica de la Unidad de Tránsito manifestó que la aplicación si posee una interfaz amigable y de fácil uso porque utiliza componentes sencillos y de fácil comprensión.

g. Discusión

1. Desarrollo de la propuesta alternativa

El presente trabajo de titulación denominado **“Tecnificación del Sistema Municipal De Estacionamiento Rotativo Tarifado (SIMERT) de la Ciudad de Loja, módulo conectividad de la red de datos del SIMERT utilizando la tecnología GPRS”**, proporcionó a la Institución la implementación de la arquitectura cliente-servidor para llevar a cabo la conectividad de la red de datos del SIMERT, a través de la implementación de un servidor y creación de una aplicación sockets para el mismo denominada **“SERVER–SIMERT”**; con la finalidad de brindar seguridad y agilizar los múltiples procesos que realiza la Institución, ofreciendo así un servicio seguro, transparente y oportuno.

El objetivo general así como sus objetivos específicos lograron ser abarcados en su totalidad, gracias a la aplicación ordenada de métodos y técnicas orientadas al desarrollo de la investigación y la metodología creada para realizar la conectividad de la red de datos del SIMERT

A continuación se detalla los objetivos específicos y como se logró con su total cumplimiento:

➤ **Objetivo específico 1: Realizar un análisis de requisitos para llevar a cabo la conectividad de la red de datos entre las diferentes subáreas del SIMERT.**

Para llevar a cabo este objetivo se ha hecho uso de la fase de análisis de la metodología (Ver sección Resultados, Pág. 44, apartado 1. FASE 1: Análisis de requerimientos), cumpliendo con las siguientes actividades

- Elaboración de un documento correspondiente a la Especificación de Requisitos según el formato que provee el estándar IEEE 830 (ver Anexo 7),
- Recopilación de información confiable de la estructura organizacional del Sistema Integrado Municipal de Estacionamiento Rotativo Tarifado (SIMERT).
- Recopilación de información sobre el perfil técnico y actividades que realiza el personal del SIMERT.
- Recopilar información sobre los recursos que actualmente posee el SIMERT.

- Recopilar información acerca de los proveedores que brinden el servicio de GPRS.
- Análisis técnico.
- Análisis de seguridad.
- Analizar las aplicaciones y Herramientas de Desarrollo.
- Definir la topología que se utilizará para la conectividad de la red de datos del SIMERT.
- Análisis de los requerimientos funcionales y no funcionales.

Para realizar cada una de estas actividades se ha utilizado las técnicas y métodos como entrevistas, encuestas y observación directa; determinando cada uno de los problemas existentes para generar una solución apropiada.

➤ **Objetivo 2: Realizar el diseño de la arquitectura cliente-servidor para establecer la comunicación.**

Para cumplir con este objetivo se ha utilizado la fase de diseño de la metodología (ver sección Resultados, Pág. 57, apartado 2. FASE 2: Diseño de la infraestructura para la conectividad de la red de datos.), elaborando las siguientes tareas:

- Realizar el prototipado inicial
 - Diseño del modelo de casos de uso
 - Descripción de los casos de uso
 - Diseño del modelo de procesos o robustez.
 - Diseño del modelo de secuencia
 - Diseño de la arquitectura cliente-servidor.

Una vez realizada las tareas se logró tener una perspectiva del funcionamiento del SIMERT donde el prototipado inicial se lo ha realizado en JAVA, con el JDK 1.8.0 y plataforma virtual NETBEANS IDE 8.0, para la modelación del sistema se ha empleado la herramienta case Enterprise Architect v 8.0 y finalmente el diseño de la arquitectura cliente-servidor se lo ha realizado con el programa Lucid Chart. Cumpliendo de esta manera el objetivo planteado.

➤ **Objetivo 3: Implementar la arquitectura cliente-servidor para la conectividad de la red de datos del SIMERT.**

Para cumplir con el objetivo nos hemos planteado las siguientes actividades realizadas en la fase de implementación de la metodología (Ver sección Resultados, Pág. 84, apartado 3. FASE 3: Implementación).

- Creación del servidor SIMERT.
- Configuraciones para la creación del servidor.
- Codificación o generación de código Java para la aplicación sockets denominada “SERVER-SIMER” para el servidor.
- Requerimientos de hardware y software para la instalación y utilización de la aplicación “SERVER-SIMERT”.
- Estándares para el código de la aplicación
- Arquitectura de la aplicación Sockets “SERVER-SIMERT”

Para el desarrollo de estas actividades se ha implementado el servidor que será una distribución Linux con el sistema operativo Ubuntu Server Lucid Lynx 10.04 bajo la plataforma libre LAMP, el cual será configurado correctamente para que funcione como servidor de correo, servidor web, servidor de base de datos y servidor de aplicaciones, utilizando la base de datos Mysql v. 5.5.2., el Phpmyadmin v. 5.1.72 para la administración de la base de datos, Postfix v. 2.7 para el servicio de correos y el apache 2.4.2.

Para la creación de la aplicación sockets denominada “SERVER-SIMERT”, se ha utilizado el lenguaje de programación Java en su versión JDK 1.8.0 bajo la plataforma virtual IDE NetBeans 8.0, el conector para la base de datos -java 5.1.6 -bin.jar, las librerías Commons-dbcp-1.4 y Commons-pool-1.6 para el pool de conexiones, la Librería Jasypt-1.9.1 para la encriptación y des encriptación de datos, el Enterprise Architect v 8.0 para la modelación del sistema

En cuanto a la comunicación entre el servidor y los clientes se ha hecho uso de la tecnología GPRS bajo el protocolo TCP/IP, direccionamiento IP y utilizando la clase ServerSockets dentro de la aplicación.

Finalmente la implementación del presente proyecto se ha realizado bajo la licencia Creative Commons (Ver Anexo 15).

➤ **Objetivo 4: Realizar las pruebas pilotos para verificar la conectividad en la red de datos del SIMERT.**

Para cumplir este objetivo se ha realizado las pruebas de carga al servidor (Ver sección Resultados, Pág. 108, apartado 4.1.1. Pruebas de carga al servidor). Se ha realizado las respectivas pruebas a la aplicación sockets “SERVER-SIMERT” como: Pruebas Unitarias (Ver sección Resultados, Pág. 111, apartado 4.2.1. Pruebas Unitarias), Pruebas de Validación (Ver sección Resultados, Pág. 116, apartado 4.2.2. Pruebas de Validación) y Pruebas de accesibilidad, funcionalidad y usabilidad (Ver sección Resultados, Pág. 120, apartado 4.2.3. Prueba de Accesibilidad, funcionalidad y usabilidad).

2. Valoración Técnica Económica Ambiental

2.1. Valoración Técnico Económica

Desde un punto de vista técnico el proyecto es factible porque constituye una necesidad dentro del Sistema Municipal de Estacionamiento Rotativo Tarifado de la ciudad de Loja, debido a que el desarrollo de la aplicación sockets denominada “SERVER-SIMERT” y la implementación del servidor cubrirán los procesos que realiza diariamente el personal operativo y administrativo del SIMERT con los registros de estacionamiento vehicular, lo que permite una adecuada organización y fiabilidad de la información, facilitando las tareas de búsqueda y almacenamiento, integrando los datos entre los diferentes departamentos del SIMERT y proporcionando niveles de seguridad tanto en la comunicación como para el acceso al servidor y su aplicación.

El proyecto es también factible económicamente debido a que las herramientas utilizadas para el desarrollo son de licencia libre, además la institución que constituye la parte del cliente en este caso el Gobierno Autónomo descentralizado de la ciudad de Loja, cuenta con los recursos necesarios para una posterior puesta en marcha de la aplicación.

Los recursos humanos, técnicos, materiales y de servicio, utilizados durante el desarrollo del proyecto se describen en las siguientes tablas.

TABLA L: VALORACIÓN ECONÓMICA DE RECURSOS HUMANOS

DESCRIPCIÓN	Cantidad	Valor/Unitario	Horas	Valor / Hora	Valor Total
Recursos Humanos					
Tesistas: César Danilo Calle Loja Julia Vanessa Morocho Guadalima	2	\$ 0.00	2800	\$ 5.00	\$ 14000.00
				Total	\$ 14000.00

TABLA LI: VALORACIÓN ECONÓMICA DE RECURSOS TÉCNICOS

DESCRIPCIÓN	Cantidad	Valor/Unitario	Horas	Valor / Hora	Valor Total
Recursos Técnicos					
Hardware					
Computador Portátil Dell studio xps 1340	1		2800	\$0.70	\$ 1960.00
Computador Portátil Toshiba Satellite LS05-SP6904R	1		2800	\$0.70	\$ 1960.00
Servidor Hp Proliant ML 350 G6 E5606. Procesador Intel® Xeon® 2.13 GHz, 4.80 GT/s	1	\$ 1434.31			\$ 1434.31
Impresora Canon IP 1900	1	\$ 50.00			\$ 50.00
Software					
Sistema Operativo Ubuntu Server Edition 10.04 TLS Lucid Lynx.		\$ 0.00			\$ 0.00
Entorno de Desarrollo IDE Netbeans v. 8.0		\$ 0.00			\$ 0.00
Base de Datos Mysql v. 5.5.2		\$ 0.00			\$ 0.00

Phpmyadmin v. 5.1.72		\$ 0.00			\$ 0.00
Lenguaje de Programación Java - JDK 1.8.0		\$ 0.00			\$ 0.00
Enterprise Architect v. 8.0		\$ 0.00			\$ 0.00
Postfix v. 2.7		\$ 0.00			\$ 0.00
Apache 2.4.2		\$ 0.00			\$ 0.00
Comunicación					
Internet (mensual)		\$ 0.00	350	\$ 0.70	\$ 245.00
Recargas Claro (mensual)	80	\$ 1.00			\$ 80.00
Chip GPRS	20	\$ 0.00			\$ 0.00
Servicio GPRS	1000 MB	\$ 20,00(mes/chip)			\$ 400.00
				Total	\$ 6129.31

TABLA LII: VALORACIÓN ECONÓMICA DE RECURSOS MATERIALES

DESCRIPCIÓN	Cantidad	Valor/Unitario	Horas	Valor / Hora	Valor Total
Recursos Materiales					
Resmas de hojas	8	\$ 4.00			\$ 32.00
Cartuchos	4	\$ 20.00			\$ 80.00
Cd's	2	\$ 0.60			\$ 1.20
Copias	600	\$ 0.02			\$ 12.00
Suministros de oficina(perforadora, carpeta, esferos, perfiles, grapadoras)		\$ 10.00			\$ 10.00
				Total	\$ 135.20

TABLA LIII: VALORACIÓN ECONÓMICA DE RECURSOS DE SERVICIOS

DESCRIPCIÓN	Cantidad	Valor/Unitario	Horas	Valor / Hora	Valor Total
Recursos de Servicios					
Transporte Urbano (mensual)	200	\$ 0.50			\$ 100.00
Transporte Ejecutivo (mensual)	150	\$ 2.00			\$ 300.00
Total					\$ 400.00

TABLA LIV: TOTAL DE RECURSOS

DESCRIPCIÓN	VALOR
Recursos Humanos	\$ 14000.00
Recursos Técnicos	\$ 6129.31
Recursos Materiales	\$ 135.20
Recursos de Servicios	\$ 400.00
SUBTOTAL	\$ 20664.51
10 % (Imprevistos)	\$ 18598.06
TOTAL	\$ 39262.57

2.2. Valoración Ambiental

El proyecto es factible desde el punto de vista ambiental debido a que se eliminará en parte el uso de papel es decir las fichas de control (Ver Anexo 8) en las que actualmente el personal operativo realiza el registro de los vehículos estacionados en las áreas del SIMERT.

h. Conclusiones

Con la finalización del presente proyecto se ha llegado a establecer las siguientes conclusiones:

- La implementación del servidor Hp Proliant, mejoró las condiciones de conectividad de los equipos informáticos que conforman la red del SIMERT, minimizando notablemente fallos o caídas en las conexiones y mejorando el rendimiento de esta.
- Los resultados de la investigación muestran que el servidor bajo la plataforma Ubuntu Server proporciona mayor estabilidad en la red optimizando el ancho de banda que se tiene.
- La aplicación Sockets denominada “SERVER-SIMERT” para el servidor, logró, organizar, controlar y gestionar ágilmente la información de los registros de estacionamiento vehicular generados por el personal operativo del SIMERT.
- Las herramientas libres tales como: Mysql v. 5.5.2, Phpmyadmin v. 5.1.72, Postfix v. 2.7, JAVA JDK 1.8.0, NETBEANS IDE 8.0, Enterprise Architect v 8.0, clase ServerSockets, Mysql conector -java 5.1.6 -bin.jar, Librerías Commons-pool-1.6, Commons-dbcp-1.4 y Jasypt-1.9.1., utilizadas en el desarrollo del presente proyecto, reducen significativamente el costo de una aplicación.
- El patrón Modelo Vista Controlador (MVC) aplicado en el desarrollo del sistema, permitió una mayor eficiencia y factibilidad durante los procesos del mismo, facilitando un mantenimiento posterior más efectivo.
- El uso de la tecnología GPRS, el protocolo TCP/IP y el direccionamiento IP dentro del proyecto, facilitó la creación del túnel de comunicación con los clientes y el servidor, proporcionando estabilidad e integridad segura en la transferencia de información de los registros de estacionamiento vehicular del SIMERT.
- Las configuraciones de los puertos tanto para la aplicación socket y aplicaciones web, en el servidor proxy; permitió un mayor control y filtrado de las peticiones de los usuarios que acceden al servidor.



- El uso de la librería de encriptación Jasypt aseguró la integridad de la información de los registros de estacionamiento vehicular del SIMERT durante el envío y recepción de datos.

i. Recomendaciones

Al finalizar el presente proyecto se recomienda lo siguiente:

- Utilizar el framework NIO Netty en el desarrollo de aplicaciones con sockets, para potenciar la seguridad de la comunicación entre los clientes y el servidor; ya que Netty posee protocolos de soporte como: HTTP, WebSocket, SSL-StarTLS y servicios de transporte como: Socket, Datagram, HTTP TUNEL y IN-VM Pipe.
- Para futuros proyectos que se desarrollen en la carrera de Ingeniería en Sistemas, relacionados con la creación de servidores se debe utilizar la distribución Centos 6.4; ya que se obtienen mayores beneficios como: seguridad, estabilidad, fácil mantenimiento y soporte en el desarrollo activo de módulos y aplicaciones.
- Crear un centro de Cómputo en las instalaciones del SITU de acuerdo al estándar TIA942 para que garantice la seguridad de la información.
- Crear un Clúster de servidores para la integración de los servicios como: correo electrónico, aplicaciones web, base de datos y aplicaciones sockets, permitiendo maximizar las funciones del sistema y garantizar el rendimiento del mismo.
- Utilizar herramientas libres como Zentyal para maximizar la seguridad de la red por lo que zentyal incluye un framework que integra funcionalidades como servidor DNS, DHCP, firewall, backup, servidor VoIP, samba, correo, proxy, etc automatizando la mayoría de las tareas ahorrando tiempo en la administración de sistemas, además Zentyal gestiona las amenazas de seguridad (UTM), la infraestructura de red, y actúa como puerta de enlace a Internet (Gateway).

j. Bibliografía

1. LIBROS:

- [1] I. Alegría Loinaz, R. Cortiñas y A. Ezeiza., “Correo electrónico”, Linux Administración del sistema y la red, M. M., Ed. 1ª, Madrid: España, 2005, pp. 251-253.
- [2] I. Alegría Loinaz, R. Cortiñas y A. Ezeiza., “Protocolo TCP/IP”, Linux Administración del sistema y la red, M. M., Ed. 1ª, Madrid: España, 2005, pp. 423.

2. PÁGINAS WEB:

- [3] GAD Municipal de Loja, SIMERT <http://www.loja.gob.ec/contenido/SIMERT> (2010) Accedido el 17 de Julio 2013.
- [4] Ejemplos java y C/linux, Pool de conexiones BasicDataSource, <http://www.chuidiang.com/java/mysql/BasicDataSource-Pool-Conexiones.php> (2007) Accedido el 17 de Febrero de 2014.
- [5] C. Raquel, O. Mario y G.Carrio (2013, Mayo 17). “Servidor de Correo Electrónico”, [PDF] (1ra. Ed.) [Online]. Disponible en: http://upload.wikimedia.org/wikipedia/commons/a/a1/Servidor_ubuntu.pdf.
- [6] R. Clericus, (2005, Marzo), “Análisis Y Estudio De Redes GPRS”, [PDF] (1ra Ed.), [Online] Disponible: <http://cybertesis.uach.cl/tesis/uach/2005/bmficis211a/doc/bmficis211a.pdf>.
- [7] L. Calvert, y M. Donahoo. (2008, Abril 08), “TCP/IP Sockets In Java”, [PDF] (1ra Ed.) [Online] Disponible en: <http://www.dlsi.ua.es/asignaturas/sid/JSockets.pdf>.
- [8] K.Vanitha, y A.Sudha, (2013, Diciembre), “Método eficaz para webservice de mayor seguridad basada

interacción utilizando jasypt”, [PDF], (3ra ed.) [Online] Disponible en:
<http://www.ijcnwc.org/papers/vol3no62013/8vol3no6.pdf>.

[9] Jasypt, (2014, Febrero 25), “Java simplificado Encryption”, [Online] Disponible en: <http://www.jasypt.org/>.

[10] J. Puiggros, (2014, Enero), “Primeros Pasos con JUnit”, [Online] Disponible en: http://programacion.net/articulo/primeros_pasos_con_junit_265#0_porque.



k. Anexos



Anexo 1: Encuesta realizada a la Coordinadora del SIMERT.

Esta encuesta nos permitió conocer el perfil técnico y las actividades que desempeña la Coordinadora en el Sistema Municipal de estacionamiento rotativo tarifado (SIMERT).



UNIVERSIDAD NACIONAL DE LOJA
ÁREA DE LA ENERGÍA, LAS INDUSTRIAS Y LOS RECURSOS NATURALES NO
RENOVABLES
CARRERA DE INGENIERÍA EN SISTEMAS

ENCUESTA DIRIGIDA AL PERSONAL DEL SIMERT

Distinguida Ingeniera, dígnese a contestar con absoluta sinceridad los aspectos que a continuación se detallan, sus criterios y comentarios son estrictamente confidenciales entre usted y los investigadores, los mismos que servirán para conocer el perfil técnico y actividades que desempeña en el Sistema Municipal de Estacionamiento Rotativo Tarifado (SIMERT).

Nombre: *Sandra Ferro*
Fecha: *15-01-2014*

1.- ¿Qué función usted cumple dentro de la organización?

Administrador (X)

Secretaria ()

Contador ()

Otros:

2.- ¿Qué actividades realiza dentro de la organización?

Regulación del Personal Operativo, cálculos de horas de rotación, ocupación, y estacionamiento en base a las fichas de control.

3.- ¿Utiliza alguna aplicación o software para realizar sus actividades?

Si () No (X)

¿Porqué?: *actualmente todo proceso lo realizamos manualmente*



4.- ¿Para realizar sus actividades utiliza información adicional generada por alguno de los departamentos del SIMERT?

Si () No (X)

¿Porqué?: *la información requerida para las actividades son utilizadas directamente las fichas de control y los controladores registran.*

5.- ¿El departamento donde usted labora cuenta con políticas de seguridad?

Si () No (X)

¿Porqué?: *no existe un lugar fijo donde se pueda ubicar los departamentos del Simert.*

6.- ¿Qué Sistema Operativo usted utiliza actualmente?

Windows (X)

Linux ()

Otros:.....

7.- ¿Cuál es la versión del o los Sistemas Operativos instalados en su computador?

Windows XP y Windows Seven.

8.- ¿Le parecería adecuado, la utilización de otro Sistema Operativo dentro de la organización?

Si (r) No ()

¿Porqué?: *es de gran ayuda la utilización de nuevas tecnologías*



9.- ¿Tiene algún conocimiento del Sistema Operativo Ubuntu?

Si () No (X)

¿Por qué? *No contamos con los instaladores de este sistema*

10.- ¿Posee algún tipo de conocimiento sobre la tecnología GPRS y direccionamiento IP?

Si () No (X)

¿Por qué? *no utilizamos dicha tecnología*



Anexo 2: Encuesta realizada a la Secretaria del SIMERT.

Esta encuesta nos permitió conocer el perfil técnico y las actividades que desempeña la Secretaria en el Sistema Municipal de estacionamiento rotativo tarifado (SIMERT).



UNIVERSIDAD NACIONAL DE LOJA
ÁREA DE LA ENERGÍA, LAS INDUSTRIAS Y LOS RECURSOS NATURALES NO
RENOVABLES
CARRERA DE INGENIERÍA EN SISTEMAS

ENCUESTA DIRIGIDA AL PERSONAL DEL SIMERT

Distinguida Ingeniera, dígnese a contestar con absoluta sinceridad los aspectos que a continuación se detallan, sus criterios y comentarios son estrictamente confidenciales entre usted y los investigadores, los mismos que servirán para conocer el perfil técnico y actividades que desempeña en el Sistema Municipal de Estacionamiento Rotativo Tarifado (SIMERT).

Nombre: *María Rojas*
Fecha: *02-02-2014*

1.- ¿Qué función usted cumple dentro de la organización?

Administrador ()

Secretaria (x)

Contador ()

Otros:

2.- ¿Qué actividades realiza dentro de la organización?

Atención de clientes, de reclamos. Encargada de sacar turnos de ocupación de rotación y de estacionamiento de acuerdo a las hojas de control a los controladores o personal operativo registran de los vehículos estacionados.

3.- ¿Utiliza alguna aplicación o software para realizar sus actividades?

Si () No (x)

¿Porqué?:
Solamente utilizo aplicaciones de ofimática como excel, hojas de cálculo.



4.- ¿Para realizar sus actividades utiliza información adicional generada por alguno de los departamentos del SIMERT?

Si () No (X)

¿Porqué?: Actualmente no contamos con una organización estructurada.

5.- ¿El departamento donde usted labora cuenta con políticas de seguridad?

Si () No (X)

¿Porqué?: No hay una debida organización.

6.- ¿Qué Sistema Operativo usted utiliza actualmente?

Windows (X)

Linux ()

Otros:.....

7.- ¿Cuál es la versión del o los Sistemas Operativos instalados en su computador?

Xp y windows seven.

8.- ¿Le parecería adecuado, la utilización de otro Sistema Operativo dentro de la organización?

Si () No (X)

¿Porqué?: No contamos con los conocimientos basicos de otros sistemas operativos.

9.- ¿Tiene algún conocimiento del Sistema Operativo Ubuntu?

Si () No (X)

¿Por qué? Desconocía sobre la existencia de Ubuntu.

10.- ¿Posee algún tipo de conocimiento sobre la tecnología GPRS y direccionamiento IP?

Si (X) No ()

¿Por qué? Actualmente utilizamos en los teléfonos inteligentes esta clase de tecnología.

Anexo 3: Encuesta realizada a la Ingeniera encargada de la Técnica del SITU.

Esta encuesta nos permitió conocer el perfil técnico y las actividades que desempeña la Ingeniera encargada de la Técnica del SITU.



UNIVERSIDAD NACIONAL DE LOJA
ÁREA DE LA ENERGÍA, LAS INDUSTRIAS Y LOS RECURSOS NATURALES NO
RENOVABLES
CARRERA DE INGENIERÍA EN SISTEMAS

ENCUESTA DIRIGIDA AL PERSONAL DEL SIMERT

Distinguida Ingeniera, dígnese a contestar con absoluta sinceridad los aspectos que a continuación se detallan, sus criterios y comentarios son estrictamente confidenciales entre usted y los investigadores, los mismos que servirán para conocer el perfil técnico y actividades que desempeña en el Sistema Municipal de Estacionamiento Rotativo Tarifado (SIMERT).

Nombre: Tania Verónica Vásquez Romero

Fecha: 28/01/2014

1.- ¿Qué función usted cumple dentro de la organización?

Administrador ()

Secretaria ()

Contador ()

Otros: Técnica de la Unidad de Tránsito

2.- ¿Qué actividades realiza dentro de la organización?

Soporte Técnico, Administración de la red de la UMTTSV,
Monitoreo de Unidades de Transporte Urbano y Taxi
Ejecutivo, Asesoría en temas tecnológicos, Emisión de
reportes e informes técnicos.

3.- ¿Utiliza alguna aplicación o software para realizar sus actividades?

Si (x) No ()

¿Por qué? Porque son herramientas necesarios para la administración
del transporte urbano y taxis ejecutivos.



4.- ¿Para realizar sus actividades utiliza información adicional generada por alguno de los departamentos del SIMERT?

Si () No (X)

¿Porqué?: Actualmente no me han requerido información de ese departamento.

5.- ¿El departamento donde usted labora cuenta con políticas de seguridad?

Si (X) No ()

¿Porqué?: Porque se requiere mantener la confidencialidad de la información de la UTTTSV.

6.- ¿Qué Sistema Operativo usted utiliza actualmente?

Windows (X)

Linux (X)

Otros:.....

7.- ¿Cuál es la versión del o los Sistemas Operativos instalados en su computador?

Windows 8.1 Pro

8.- ¿Le parecería adecuado, la utilización de otro Sistema Operativo dentro de la organización?

Si () No (X)

¿Porqué?: Porque el personal está familiarizado con el Sistema Operativo actual.

9.- ¿Tiene algún conocimiento del Sistema Operativo Ubuntu?

Si (X) No ()

¿Por qué? *Porque es fundamental para los labores que desempeño*

10.- ¿Posee algún tipo de conocimiento sobre la tecnología GPRS y direccionamiento IP?

Si (X) No ()

¿Por qué? *Porque requiero de ese conocimiento para los labores que desempeño*



Ing. Tania Verónica Vásquez Romero
Técnica de la Unidad de Tránsito

Anexo 4: Encuesta realizada a los controladores del SIMERT.

Esta encuesta nos permitió conocer el perfil técnico y las actividades que desempeñan los controladores en el Sistema Municipal de Estacionamiento Rotativo Tarifado (SIMERT).



UNIVERSIDAD NACIONAL DE LOJA
ÁREA DE LA ENERGÍA, LAS INDUSTRIAS Y LOS RECURSOS NATURALES NO
RENOVABLES
CARRERA DE INGENIERÍA EN SISTEMAS

ENCUESTA DIRIGIDA AL PERSONAL DEL SIMERT

Distinguida Ingeniera, dígnese a contestar con absoluta sinceridad los aspectos que a continuación se detallan, sus criterios y comentarios son estrictamente confidenciales entre usted y los investigadores, los mismos que servirán para conocer el perfil técnico y actividades que desempeña en el Sistema Municipal de Estacionamiento Rotativo Tarifado (SIMERT).

Nombre: *Daniel Alvarez*
Fecha: *31-01-2014*

1.- ¿Qué función usted cumple dentro de la organización?

Administrador ()

Secretaria ()

Contador ()

Otros: *Controlador*

2.- ¿Qué actividades realiza dentro de la organización?

Las funciones que realizan son las de controlador de los registros de estacionamiento y el control vehicular.

3.- ¿Utiliza alguna aplicación o software para realizar sus actividades?

Si () No (X)

¿Porqué?: *Usamos todos los procesos de control manualmente a través de hojas de control*



4.- ¿Para realizar sus actividades utiliza información adicional generada por alguno de los departamentos del SIMERT?

Si () No ()

¿Porqué?: La información del administrador de zonas de trabajo y los horarios

5.- ¿El departamento donde usted labora cuenta con políticas de seguridad?

Si () No ()

¿Porqué?: El control de los registros de estacionamiento se los realiza en las zonas de trabajo.

6.- ¿Qué Sistema Operativo usted utiliza actualmente?

Windows ()

Linux ()

Otros:

7.- ¿Cuál es la versión del o los Sistemas Operativos instalados en su computador?

Windows xp y server

8.- ¿Le parecería adecuado, la utilización de otro Sistema Operativo dentro de la organización?

Si () No ()

¿Porqué?: Actualmente tenemos todo los procesos manualmente.

9.- ¿Tiene algún conocimiento del Sistema Operativo Ubuntu?

Si () No (X)

¿Por qué? *desconocimiento de la existencia de este sistema.*

10.- ¿Posee algún tipo de conocimiento sobre la tecnología GPRS y direccionamiento IP?

Si () No (X)

¿Por qué? *No porque no uso mucho estas tecnologías.*

Anexo 5: Entrevista realizada a la Ingeniera encargada de la Técnica del SITU sobre los recursos tecnológicos del SITU.



UNIVERSIDAD NACIONAL DE LOJA
ÁREA DE LA ENERGÍA, LAS INDUSTRIAS Y LOS RECURSOS NATURALES NO
RENOVABLES
CARRERA DE INGENIERÍA EN SISTEMAS

ENCUESTA A LOS INGENIEROS DEL SITU

Distinguida Ingeniera, dígnese a contestar con absoluta sinceridad los aspectos que a continuación se detallan, sus criterios y comentarios son estrictamente confidenciales entre usted y los investigadores, los mismos que servirán para conocer los recursos tecnológicos que actualmente dispone el Sistema Integrado de Transporte Urbano (SITU).

Nombre: Tania Verónica Vásquez Romero
Cargo: Técnica de la Unidad de Tránsito
Fecha: 23/01/2014

1.- ¿Actualmente el SITU posee un Centro de Cómputo?

Si () No ()

¿Por qué? *Está en proceso de equipamiento.*

2.- ¿Cuántos servidores se utiliza en el Centro de Cómputo?

En la sala de monitoreo se cuenta con 4 servidores

3.- ¿Posee algún tipo de seguridad el Centro de Cómputo?

Si () No ()

Describala: *Firewall en el proxy.*

4.- ¿Enumere y describa los dispositivos que utiliza el Centro de Cómputo?

- Servidor web. - En el que se deja la página para el sistema de rastreo de buses
- Servidor proxy
- Servidor de base de datos para el rastreo de buses
- Back up de servidores rastreo de buses
- Equipos para monitoreo (PCs)



5.- ¿El centro de cómputo del SITU cuenta con políticas de seguridad?

Si (X) No ()

¿Por qué? Para mantener la confidencialidad de la información

6.- ¿Qué sistema operativo tiene actualmente instalado los servidores?

Windows y Linux
(Windows 8) (Centos)

7.- ¿Qué Base de Datos tienen instalado en los servidores?

Mysql

8.- ¿La Base de Datos que utiliza posee una licencia gratuita o pagada?

Gratis

9.- ¿Qué tipo de información almacenan los servidores?

Todo el reporte de recorrido de buses

10.- ¿Cuál es el encargado de dar mantenimiento a los servidores?

El sistema de monitoreo está aún en fase de pruebas por lo que aún realiza el mantenimiento la empresa que lo instaló. Tania Vasquez -> mantenimiento proxy.

10.- ¿Cuál es el proveedor que brinda el servicio de internet para el SITU?

Punto net

11.- ¿Cuál es el ancho de banda que brinda el proveedor de internet?

2 Mbps.

12.- ¿El servicio brindado por el proveedor de Internet es?

Excelente ()

Bueno (X)

Regular ()

¿Por qué? *Un poco demorado el soporte técnico*

13.- ¿El SITU utiliza la tecnología GPRS?

Si (x) No ()

¿Por qué? *Para el reporte de recorrido de buses es necesario saber su ubicación actualizada al minuto*

14.- ¿Qué proveedor ofrece el servicio de la tecnología GPRS?

Movistar

15.- ¿Cuáles son los dispositivos utilizados para la comunicación con la tecnología GPRS?

-GPS

16.- ¿El SITU utiliza cableado estructurado para la comunicación con los diferentes departamentos?

No, ya que actualmente se encuentra ubicado en instalaciones provisionales no se ha realizado un cableado estructurado

Ing. Tania Verónica Vásquez Romero
Técnica de la Unidad de Tránsito

Anexo 6: Estructura de la base de datos del servidor.



Per

Anexo 7: Informe de especificación de requisitos.



**“Tecnificación del Sistema Municipal De Estacionamiento Rotativo Tarifado (SIMERT) de la ciudad de Loja, módulo conectividad de la red de datos del SIMERT utilizando la tecnología GPRS”
Especificación de requisitos de software**



Especificación de requisitos de software.

Proyecto: “Tecnificación del Sistema Municipal De Estacionamiento Rotativo Tarifado (SIMERT) de la ciudad de Loja, módulo conectividad de la red de datos del SIMERT utilizando la tecnología GPRS”

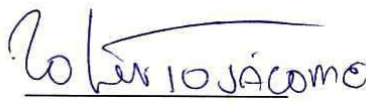



“Tecnificación del Sistema Municipal De Estacionamiento Rotativo Tarifado (SIMERT) de la ciudad de Loja, módulo conectividad de la red de datos del SIMERT utilizando la tecnología GPRS”
Especificación de requisitos de software

1 Ficha del documento

Fecha	Revisión	Autor	Verificado por el Departamento de Calidad.
12/03/2014	1.0	César Danilo Calle Loja Julia Vanessa Morocho Guadalima	 

Documento validado por las partes en fecha: 12/03/2014

Por el SIMERT	Por la Universidad Nacional de Loja
	
Fdo. Ing. Roberto Jácome	Fdo. Ing. Hernán Torres



**“Tecnificación del Sistema Municipal De
Estacionamiento Rotativo Tarifado (SIMERT) de la
ciudad de Loja, módulo conectividad de la red de
datos del SIMERT utilizando la tecnología GPRS”
Especificación de requisitos de software**

Contenido

Ficha del documento	171
Contenido	172
1. Introducción	174
1.1. Propósito	174
1.2. Alcance	174
1.3. Personal involucrado	176
1.4. Definiciones, acrónimos y abreviaturas.....	177
1.5. Referencias.....	178
1.6. Resumen	178
2. Descripción general	178
2.1. Perspectiva del producto.....	178
2.2. Funcionalidad del producto	182
2.3. Características de los usuarios	183
2.4. Restricciones	183
2.5. Suposiciones y dependencias.....	184
2.6. Evolución previsible del sistema	184
3. Requisitos específicos	184
3.1. Requisitos comunes de las interfaces	186
3.1.1. Interfaces de usuario	186
3.1.2. Interfaces de hardware	187
3.1.3. Interfaces de software	187
3.1.4. Interfaces de comunicación	187
3.2. Requisitos funcionales	188
3.2.1. Requisito funcional 1	188
3.2.2. Requisito funcional 2	188
3.2.3. Requisito funcional 3	188
3.2.4. Requisito funcional 4	188
3.2.5. Requisito funcional 5	188
3.2.6. Requisito funcional 6	189
3.2.7. Requisito funcional 7	189
3.3. Requisitos no funcionales	189
3.3.1. Requisitos de rendimiento	189



“Tecnificación del Sistema Municipal De Estacionamiento Rotativo Tarifado (SIMERT) de la ciudad de Loja, módulo conectividad de la red de datos del SIMERT utilizando la tecnología GPRS”

Especificación de requisitos de software

3.3.2.	Seguridad	189
3.3.3.	Fiabilidad	190
3.3.4.	Disponibilidad	190
3.3.5.	Mantenibilidad	190
3.3.6.	Portabilidad	190
4.	Apéndices.....	191
4.1.	Modelo de casos de uso.	191



“Tecnificación del Sistema Municipal De Estacionamiento Rotativo Tarifado (SIMERT) de la ciudad de Loja, módulo conectividad de la red de datos del SIMERT utilizando la tecnología GPRS”
Especificación de requisitos de software

1. Introducción

Una de las fases iniciales para llevar a cabo la conectividad de la red de datos del SIMERT es realizar un análisis de requisitos enfocándose en la obtención de los objetos que están directamente e indirectamente relacionados con la problemática. Es decir se debe realizar una recopilación de información confiable de la estructura organizacional del SIMERT y de los usuarios que utilizarán la red de datos, la idea de esta etapa es evaluar los requerimientos de los usuarios identificando posibles problemas desde el inicio.

Con el resultado de la fase de análisis se intenta identificar qué información ha de ser procesada, que función y rendimiento se desea, cuál será el comportamiento de la conectividad de la red de datos, que interfaces van a ser establecidas, que restricciones se pondrán, que tipo de seguridad se necesita y que criterios de validación se necesitan para definir una red de datos correcta. Como constancia de dicho análisis se debe desarrollar un documento denominado Especificación de Requisitos de Software (SRS).

El presente documento contiene las especificaciones de requisitos de software para el proyecto denominado “Tecnificación del Sistema Municipal De Estacionamiento Rotativo Tarifado (SIMERT) de la ciudad de Loja, módulo conectividad de la red de datos del SIMERT utilizando la tecnología GPRS”, contribuyendo en la organización de la información, facilitando las tareas de búsqueda y almacenamiento, integrando los datos entre las diferentes subáreas del SIMERT y proporcionando niveles de seguridad tanto en la comunicación como para el acceso al servidor y su aplicación.

Finalmente cabe recalcar que se ha desarrollado una metodología para llevar a cabo la conectividad de red de datos del SIMERT donde se ha cumplido con las actividades de cada una de las fases de manera organizada determinando principalmente los diferentes problemas existentes.

1.1. Propósito

El proyecto de fin de carrera tiene como propósito definir las especificaciones funcionales y no funcionales para la implementación del servidor y el desarrollo de la aplicación sockets denominado “SERVER-SIMERT”, potenciando el control y administración del servicio de estacionamiento vehicular del SIMERT.

1.2. Alcance

El presente proyecto denominado “Tecnificación del Sistema Municipal De Estacionamiento Rotativo Tarifado (SIMERT) de la ciudad de Loja, módulo conectividad de la red de datos del SIMERT utilizando la tecnología GPRS” es un gran aporte para potenciar el servicio a la ciudadanía, se proyecta como una base o un comienzo que contribuirá a la tecnificación de los procesos de control que actualmente se llevan a cabo de forma manual, afrontando los siguientes problemas como: falta de seguridad y agilidad en el almacenamiento y recopilación de información acerca de los registros de estacionamiento vehicular, puesto que la información crece cada día y los procesos de recuperación de la misma son más complejos así como también no cuenta con ningún tipo de conexión entre los diferentes departamentos del SIMERT.



“Tecnificación del Sistema Municipal De Estacionamiento Rotativo Tarifado (SIMERT) de la ciudad de Loja, módulo conectividad de la red de datos del SIMERT utilizando la tecnología GPRS”

Especificación de requisitos de software

El proyecto a desarrollarse tendrá como alcance la implementación de la arquitectura cliente-servidor, implementación de un servidor, desarrollo de una aplicación sockets denominada “SERVER - SIMERT” para el servidor y uso de la tecnología GPRS, cumpliendo los siguientes puntos:

Realizar un análisis de requisitos para llevar a cabo la conectividad de la red de datos entre las diferentes subáreas del SIMERT.

Recopilación de información confiable de la estructura organizacional del SIMERT (Sistema Integrado Municipal de Estacionamiento Rotativo Tarifado).

Realizar una encuesta al personal del SIMERT (Administradores y controladores) que utilizaran la arquitectura cliente-servidor.

Recopilar información sobre los recursos que actualmente posee el SIMERT.

Recopilar información acerca de los proveedores que brinden el servicio de GPRS.

Análisis de los requerimientos funcionales y no funcionales.

Análisis técnico.

Análisis de seguridad.

Especificar los casos de usos y escenarios.

Analizar las aplicaciones y Herramientas de Desarrollo

Realizar el diseño de la arquitectura cliente-servidor para establecer la comunicación.

Diseño del modelo de la base de datos para el almacenamiento de registros del SIMERT en el servidor.

Diseño del Modelo de casos de uso.

Prototipado Inicial.

PANTALLAS “SERVER-SIMERT”

Descripción de los casos de uso

Diseño del Modelo de procesos.

Diseño del Modelo de secuencia.

Realizar el diseño de la arquitectura cliente-servidor



“Tecnificación del Sistema Municipal De Estacionamiento Rotativo Tarifado (SIMERT) de la ciudad de Loja, módulo conectividad de la red de datos del SIMERT utilizando la tecnología GPRS”
Especificación de requisitos de software

Implementar la arquitectura cliente-servidor para la conectividad de la red de datos del SIMERT.

Programación de la aplicación del servidor utilizando aplicación sockets

Definir la topología que se utilizara para la conectividad de la red de datos del SIMERT.

Simulación de la arquitectura cliente-servidor en el Packet Tracer

Descripción de la arquitectura cliente – servidor.

Configuración del Proxy.

Configuración del servidor.

Realizar las pruebas para verificar la conectividad en la red de datos del SIMERT.

Integración de los módulos.

Prueba de unidades.

Prueba de funcionalidad.

1.3. Personal involucrado

Nombre	César Danilo Calle Loja
Rol	Analista, diseñador y programador.
Categoría profesional	Egresado de Ingeniería en Sistemas.
Responsabilidades	Análisis de información, diseño, programación de la aplicación sockets y configuración del servidor.
Información de contacto	cesar.c.jtam@gmail.com

Nombre	Julia Vanessa Morocho Guadalima
Rol	Analista, diseñadora y programadora.
Categoría profesional	Egresada de Ingeniería en Sistemas.
Responsabilidades	Análisis de información, diseño, programación de la aplicación sockets y configuración del servidor.
Información de contacto	juvamogu@gmail.com



“Tecnificación del Sistema Municipal De Estacionamiento Rotativo Tarifado (SIMERT) de la ciudad de Loja, módulo conectividad de la red de datos del SIMERT utilizando la tecnología GPRS”
Especificación de requisitos de software

1.4. Definiciones, acrónimos y abreviaturas

Nombre	Descripción
Administrador del Servidor.	Persona encarga del funcionamiento y control del servidor.
Personal Operativo	Controladores o trabajadores del SIMERT.
Cliente	Personal Operativo que utilizan el dispositivo Tablet para el envío y recepción de tramas, desde y hacia el servidor.
Usuario	Persona que utiliza el servicio de estacionamiento vehicular.
SIMERT	Sistema Municipal de Estacionamiento Rotativo Tarifado.
SITU	Sistema Integrado de Transporte Urbano.
SRS	Especificación de Requisitos Software.
SERVER-SIMERT	Nombre de la aplicación sockets que se encuentra en el servidor.
Tramas	Información de los registros de estacionamiento vehicular que son enviados en un formato establecido tanto para el cliente como para el servidor.
Servidor	Computadora en la que se ejecuta varias aplicaciones en beneficio de otras aplicaciones llamadas clientes.
RF	Requerimiento Funcional.
RNF	Requerimiento No Funcional.



“Tecnificación del Sistema Municipal De Estacionamiento Rotativo Tarifado (SIMERT) de la ciudad de Loja, módulo conectividad de la red de datos del SIMERT utilizando la tecnología GPRS”
Especificación de requisitos de software

1.5. Referencias

Titulo	Ruta	Autor
Especificación de Requisitos Software según el estándar de IEEE 830	https://www.dropbox.com/s/2cc2krxzy63vkvj/Guia%20IEEE830.pdf	Raúl Monferrer Agut

1.6. Resumen

Este documento está estructurado de tres partes, la primera parte consta de la introducción detallándose el propósito y alcance que tendrá el proyecto.

La segunda parte muestra la descripción general de las aplicaciones, en donde se determinan las principales funciones de las mismas, así mismo las restricciones, suposiciones y dependencias que intervienen en el desarrollo sin entrar en detalles.

En la tercera parte se especifica punto a punto los requisitos que deben cumplirse.

2. Descripción general

2.1. Perspectiva del producto

Como parte de la solución propuesta por este proyecto de tesis es realizar la conectividad de la red de datos del Sistema Municipal De Estacionamiento Rotativo Tarifado (SIMERT) del Cantón Loja de la provincia de Loja utilizando la tecnología GPRS.

Para su desarrollo se ha realizado el diseño de la arquitectura cliente-servidor como se muestra en la figura N°1 y su simulación a través del Packet Tracer, se ha creado un servidor con las configuraciones necesarias para que funcione como servidor web, servidor de correo, servidor de base de datos y servidor de aplicaciones. También se ha desarrollado una aplicación sockets para el servidor denominada “SERVER-SIMERT”, se ha hecho uso de la tecnología GPRS bajo el protocolo TCP/IP y direccionamiento IP para establecer la comunicación entre la arquitectura cliente-servidor.

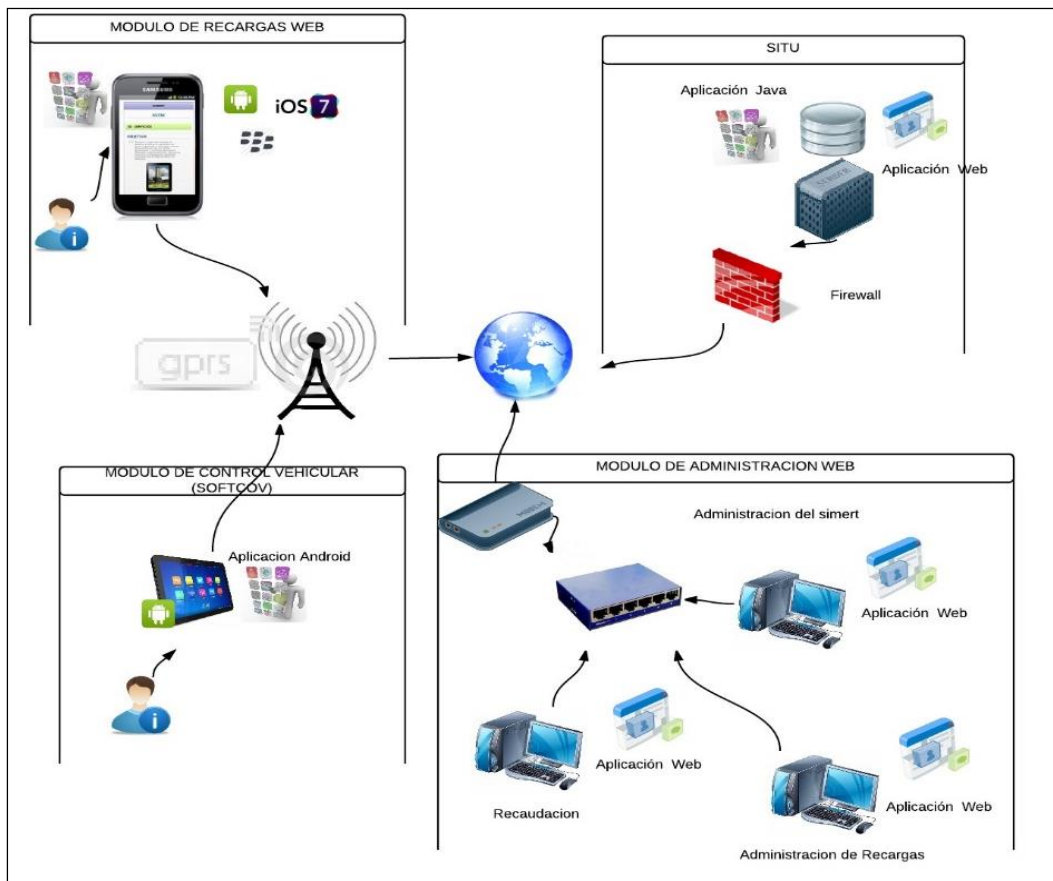


Figura N°1: Arquitectura Cliente-Servidor

A continuación se detalla la simbología de la arquitectura cliente – servidor:

IMAGEN	DESCRIPCIÓN
	<p>RED GPRS CLARO.</p> <p>La empresa de telecomunicaciones Claro brindara el servicio de GPRS, a través de los chips que cada Tablet utilizara. Se contratara un plan de megas para la trasmisión de datos.</p>
	<p>PROTECCIÓN FIREWALL.</p> <p>Un cortafuego (firewall en inglés) es una parte de un sistema o una red que está diseñada para bloquear el acceso no autorizado, permitiendo al mismo tiempo comunicaciones autorizadas.</p>
	<p>SERVIDOR.</p> <p>Un servidor es un nodo que, formando parte de una red, provee servicios como web, correo y almacenamiento de datos entre otros. Algunos servicios habituales son los servicios de archivos, que permiten a los usuarios almacenar y acceder a los archivos de una computadora y los servicios de aplicaciones, que realizan tareas en beneficio directo del usuario final.</p>
	<p>USUARIO FINAL</p> <p>Persona o personas que van a manipular de manera directa un producto de software.</p>
	<p>BASE DE DATOS</p> <p>Una base de datos o banco de datos es un conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso.</p>

<p>Aplicación Java</p> 	<p>APLICACIÓN JAVA</p> <p>Una aplicación es un tipo de programa informático diseñado como herramienta para permitir a un usuario realizar uno o diversos tipos de trabajos.</p>
	<p>TELÉFONO INTELIGENTE</p> <p>Dispositivo final que interactuara directamente con el usuario final y permitirá el uso y la ejecución de las aplicaciones clientes.</p>
	<p>APLICACIÓN WEB</p> <p>Se denomina aplicación web a aquellas herramientas que los usuarios pueden utilizar accediendo a un servidor web a través de Internet o de una intranet mediante un navegador.</p>
	<p>TABLET</p> <p>Dispositivo final que interactuara directamente con el usuario final y permitirá el uso y la ejecución de las aplicaciones clientes.</p>
	<p>Computadora</p> <p>Dispositivo final que interactuara directamente con el usuario final y permitirá el uso y la ejecución de las aplicaciones clientes.</p>

	<p>Router Adsl</p> <p>El encaminador ADSL es un dispositivo que permite la conexión a internet como también conectar al mismo tiempo uno o varios equipos o incluso una o varias redes de área local.</p>
	<p>Switch</p> <p>Un switch es un dispositivo de propósito especial diseñado para resolver problemas de rendimiento en la red, debido a anchos de banda pequeños y embotellamientos.</p>
	<p>Internet</p> <p>Internet es un conjunto descentralizado de redes de comunicación interconectadas que utilizan la familia de protocolos TCP/IP.</p>

2.2. Funcionalidad del producto

En la creación de la aplicación sockets “SERVER-SIMERT” para el servidor interviene el administrador del servidor quien visualizará las tramas de los registros de estacionamiento vehicular recibidas por el cliente.

El cliente es el personal operativo del SIMERT, quienes envían al servidor los registros de estacionamiento vehicular mediante tramas a través del dispositivo Tablet.

La aplicación sockets “SERVER-SIMERT se encarga de guardar, validar, modificar y eliminar los datos recibidos de cada una de las tramas.

Las tramas que se envían o se reciben tanto para el cliente como para el servidor son encriptadas y des encriptadas como método de seguridad en la información.

Los datos enviados y recibidos que son las tramas llegan al servidor a través de la red GPRS.

Para salir de la aplicación sockets “SERVER-SIMERT”, se valida los datos de usuario y contraseña que tendrá que ingresar el administrador del servidor.

Para la creación del servidor se ha realizado las configuraciones necesarias para que funcione como servidor de correos, servidor de aplicaciones, servidor de base de datos y servidor de aplicaciones. Porque tanto el módulo de administración “SIMERTWEB” como el módulo de



“Tecnificación del Sistema Municipal De Estacionamiento Rotativo Tarifado (SIMERT) de la ciudad de Loja, módulo conectividad de la red de datos del SIMERT utilizando la tecnología GPRS”
Especificación de requisitos de software

recargas web almacenan sus aplicaciones en el servidor, para su funcionamiento y junto con el módulo de control vehicular “SOFTCOV” utilizan la información almacenada de los registros de estacionamiento vehicular para sus diversas actividades, garantizando la seguridad y agilidad de la misma, como se muestra en la figura N°1.

2.3. Características de los usuarios

Tipo de usuario	Administrador del Servidor.
Formación	Funcionario público
Actividades	Control, manejo y visualización de las tramas que llegan al servidor enviadas por el personal operativo. Cerrar la aplicación sockets “SERVER-SIMERT”.

Tipo de usuario	Personal Operativo
Formación	Funcionario público
Actividades	Envío de tramas hacia el servidor, validando su acceso al mismo.

2.4. Restricciones

Se debe tener en cuenta lo siguiente para el desarrollo aplicación sockets “SERVER-SIMERT” y creación del servidor:

Funcionamiento de la aplicación con internet.

Uso de una dirección IP o host y un número de puerto puede ser cualquier entero entre 1 y 65535 para la comunicación entre servidor y cliente.

Implementar un diseño amigable para el usuario final.

Lenguajes y tecnologías usadas: JAVA con el JDK v. 1.8.0, plataforma virtual NETBEANS IDE 8.0, S.O. Ubuntu Server Edition 10.04 TLS Lucid Lynx, sistema de infraestructura LAMP, MYSQL v. 5.5.2., Phpmyadmin v. 5.1.72, Postfix v. 2.7, librerías Commons-pool-1.6, Commons-dbc-1.4 y Jasypt-1.9.1., Enterprise Architect v 8.0, Apache 2.4.2, y GPRS.

El servidor debe ser ubicado en una zona físicamente segura.

En el servidor se configurará solo los servicios y protocolos a usarse, ya que un servidor mal configurado es un potencial riesgo de caída del sistema.



**“Tecnificación del Sistema Municipal De
Estacionamiento Rotativo Tarifado (SIMERT) de la
ciudad de Loja, módulo conectividad de la red de
datos del SIMERT utilizando la tecnología GPRS”
Especificación de requisitos de software**

El servidor estará ubicado en la red detrás de un servidor Firewall garantizando la seguridad en la información.

2.5. Suposiciones y dependencias

Se asume que los requisitos descritos en el documento son estables.

El hardware y software a utilizar para la ejecución tanto de la aplicación sockets como para el servidor deben cumplir con los requisitos establecidos para garantizar el correcto funcionamiento de los mismos.

2.6. Evolución previsible del sistema

Crear un Centro de Cómputo en el SIMERT para potenciar el funcionamiento de los equipos.

Crear políticas de seguridad para garantizar la integridad de los servidores.

Tener conexión con la Base de Datos de la Policía Nacional para mayor control de los vehículos.

Implementar nuevos módulos en la aplicación para el control del envío y recepción de tramas y reportes sobre el acceso al servidor.

3. Requisitos específicos

Identificación del requerimiento:	RF01
Nombre del Requerimiento:	Autenticación del administrador del servidor
Descripción del requerimiento:	El sistema permitirá cerrar y abrir la aplicación mediante su nombre de usuario y contraseña. Cambiar usuario y contraseña
Prioridad del requerimiento	Alta

Identificación del requerimiento:	RF02
Nombre del Requerimiento:	Visualización del envío y recepción de tramas.
Descripción del requerimiento:	El sistema permitirá visualizar cada una de las tramas que son enviadas por el personal operativo.



“Tecnificación del Sistema Municipal De Estacionamiento Rotativo Tarifado (SIMERT) de la ciudad de Loja, módulo conectividad de la red de datos del SIMERT utilizando la tecnología GPRS”
Especificación de requisitos de software

Prioridad del requerimiento	Alta
------------------------------------	------

Identificación del requerimiento:	RF03
Nombre del Requerimiento:	Autenticación del Personal Operativo.
Descripción del requerimiento:	El sistema permitirá el ingreso del personal operativo al servidor, mediante su nombre de usuario y contraseña a través de la trama correspondiente.
Prioridad del requerimiento	Alta

Identificación del requerimiento:	RF04
Nombre del Requerimiento:	Validación de las tramas.
Descripción del requerimiento:	El sistema permitirá validar cada una de las tramas recibidas por el personal operativo.
Prioridad del requerimiento	Alta

Identificación del requerimiento:	RF05
Nombre del Requerimiento:	Almacenamiento de la información de las tramas.
Descripción del requerimiento:	El sistema permitirá almacenar cada uno de los datos de la trama en las tablas correspondientes de la base de datos.
Prioridad del requerimiento	Alta



“Tecnificación del Sistema Municipal De Estacionamiento Rotativo Tarifado (SIMERT) de la ciudad de Loja, módulo conectividad de la red de datos del SIMERT utilizando la tecnología GPRS”
Especificación de requisitos de software

Identificación del requerimiento:	RF06
Nombre del Requerimiento:	Modificación de la información de las tramas.
Descripción del requerimiento:	El sistema permitirá modificar cada uno de los datos de la trama en las tablas correspondientes de la base de datos.
Prioridad del requerimiento	Alta

Identificación del requerimiento:	RF07
Nombre del Requerimiento:	Eliminación de la información de las tramas.
Descripción del requerimiento:	El sistema permitirá eliminar cada uno de los datos de la trama en las tablas correspondientes de la base de datos.
Prioridad del requerimiento	Alta

3.1. Requisitos comunes de las interfaces

La aplicación sockets “SERVER-SIMERT” visualiza y realiza el envío y recepción de las tramas de los registros de control de estacionamiento vehicular recibidas por el personal operativo del SIMERT, así como también realiza la validación, modificación, eliminación y almacenamiento de las mismas.

El servidor permite el almacenamiento de la información de las tramas, correos electrónicos y aplicaciones web que son utilizados por el módulo de administración “SIMERTWEB”, módulo de recargas web y módulo de control vehicular “SOFTCOV” como se muestra en la figura N°1.

3.1.1. Interfaces de usuario

La interfaz de usuario estará constituida por botones, campos de textos, área de textos y otros componentes necesarios para que la visualización cumpla los objetivos del sistema propuesto para el cual fue programado

Para la visualización de la interfaz de usuario se la puede apreciar desde el servidor que se encuentra ubicado en el SITU.



“Tecnificación del Sistema Municipal De Estacionamiento Rotativo Tarifado (SIMERT) de la ciudad de Loja, módulo conectividad de la red de datos del SIMERT utilizando la tecnología GPRS”
Especificación de requisitos de software

3.1.2. Interfaces de hardware

Procesador de 2.20GHz o superior.

Memoria mínima de 3GB

Disco 300 GB

Kernel Linux 2.6.32

Mouse.

Teclado.

3.1.3. Interfaces de software

Navegador: Google Chrome.

S.O. Ubuntu Server Edition 10.04 TLS Lucid Lynx

JAVA con el JDK v. 1.8.0, plataforma virtual NETBEANS IDE 8.0.

Sistema de infraestructura LAMP.

MYSQL v. 5.5.2, Phpmyadmin v. 5.1.72

Postfix v. 2.7.

GPRS.

Librerías Commons-pool-1.6, Commons-dbcp-1.4 y Jasypt-1.9.1.

Enterprise Architect v 8.0.

Apache 2.4.2

3.1.4. Interfaces de comunicación

Para la comunicación entre los clientes que son los controladores o personal operativo del SIMERT con la aplicación sockets del Servidor se ha utilizado la tecnología GPRS bajo el protocolo TCP/IP, direccionamiento IP y uso de la clase ServerSockets dentro de la aplicación.

Para la comunicación de la base de datos del SIMERT con los diferentes módulos de las aplicaciones web como el módulo de administración “SIMERTWEB” y módulo de recargas web es directa a través de un direccionamiento IP, donde se utilizará la IP pública que actualmente tiene el SITU bajo protocolo TCP para transmitir y administrar la información.



**“Tecnificación del Sistema Municipal De
Estacionamiento Rotativo Tarifado (SIMERT) de la
ciudad de Loja, módulo conectividad de la red de
datos del SIMERT utilizando la tecnología GPRS”
Especificación de requisitos de software**

3.2. Requisitos funcionales

3.2.1. Requisito funcional 1

Autenticación del administrador del servidor:

El administrador del servidor quien se encargará del funcionamiento y control del mismo deberá autenticarse para ingresar y cerrar la aplicación sockets “SERVER-SIMERT”.

3.2.2. Requisito funcional 2

Visualización del envío y recepción de tramas:

El administrador del servidor quien se encargará del funcionamiento y control del mismo podrá visualizar a cada momento las tramas de los registros de control de estacionamiento vehicular que envía el personal operativo, donde por cada trama recibida se visualizará la dirección IP, la fecha y la hora en que el controlador o cliente ha enviado una trama. También se visualizará los mensajes que indican el nombre del registro al que pertenece la trama y los procesos que se ha realizado con la misma.

3.2.3. Requisito funcional 3

Autenticación del Personal Operativo:

El personal operativo quien se encargará del envío y recepción de tramas hacia y desde el servidor deberá autenticarse para acceder al mismo.

3.2.4. Requisito funcional 4

Validación de las tramas:

Para el envío y recepción de tramas tanto para el servidor como para el personal operativo deben ser encriptadas y des encriptadas a través de una clave como parte de la seguridad en la información.

Además las tramas se validan por números y caracteres definidos tanto para el inicio de trama, fin de trama como separador de datos, según al registro de estacionamiento vehicular correspondiente.

3.2.5. Requisito funcional 5

Almacenamiento de la información de las tramas:

Cada una de las tramas que son enviadas por el personal operativo y recibido en el servidor, corresponde a un tipo de registro de control vehicular.

Donde se procede a des encriptar y extraer los datos de la trama, almacenándolos en las tablas correspondientes al registro a través del uso de sentencias MYSQL.



“Tecnificación del Sistema Municipal De Estacionamiento Rotativo Tarifado (SIMERT) de la ciudad de Loja, módulo conectividad de la red de datos del SIMERT utilizando la tecnología GPRS”
Especificación de requisitos de software

3.2.6. Requisito funcional 6

Modificación de la información de las tramas:

Cada una de las tramas que son enviadas por el personal operativo y recibido en el servidor, corresponde a un tipo de registro de control vehicular.

Donde se procede a des encriptar y extraer los datos de la trama y a través del uso de consultas MYSQL se realiza la modificación de los datos en las tablas correspondientes al registro.

3.2.7. Requisito funcional 7

Eliminación de la información de las tramas:

Cada una de las tramas que son enviadas por el personal operativo y recibido en el servidor, corresponde a un tipo de registro de control vehicular.

Donde se procede a des encriptar y extraer los datos de la trama y a través del uso de consultas MYSQL se realiza la eliminación de los datos en las tablas correspondientes al registro.

3.3. Requisitos no funcionales

3.3.1. Requisitos de rendimiento

EL número de clientes que se conectarán al servidor son 27 controladores que a través de los dispositivos tablets enviarán las tramas correspondientes a los registros de estacionamiento vehicular. El número de transacciones o tramas enviadas por cada uno de los controladores será de 15 tramas en dos segundos como máximo.

Cada una de las transacciones de tramas enviadas será procesada máximo en dos segundos sin necesidad que los controladores o clientes tengan que esperar para completar la transacción.

Se utilizará la base de datos Mysql para el almacenamiento de la información en el servidor, utilizando un pool de conexiones a través de las librerías commons-pool y commons-dhcp de java, donde podemos tener varias conexiones abiertas, de forma que cuando un cliente necesite una, la "reserve" para su uso y cuando termine, la "libere" para que pueda ser usada por los demás clientes.

3.3.1. Seguridad

Las tramas que se envían o se reciben tanto para el cliente como para el servidor son encriptadas y des encriptadas utilizando la librería JASYPT como método de seguridad en la información.

El servidor estará ubicado en la red detrás de un servidor Firewall garantizando la seguridad en la información. Se ha utilizado ficheros logs de actividad para registrar el acceso de cada uno de los controladores al servidor, registrando la fecha, hora y datos del controlador quien ha realizado el acceso al servidor.



“Tecnificación del Sistema Municipal De Estacionamiento Rotativo Tarifado (SIMERT) de la ciudad de Loja, módulo conectividad de la red de datos del SIMERT utilizando la tecnología GPRS”
Especificación de requisitos de software

3.3.2. Fiabilidad

La utilización de la tecnología GPRS y direccionamiento IP garantiza la comunicación entre el cliente y el servidor. La utilización de un Firewall garantiza la integridad de la información.

3.3.3. Disponibilidad

La disponibilidad del servidor será del 100% las 24 horas del día, ya que el mismo se encontrará en el centro de cómputo del SITU garantizando la comunicación y el almacenamiento de la información para las diversas actividades que realizan las áreas del SIMERT.

3.3.4. Mantenibilidad

El administrador del servidor será la persona encargada del mantenimiento y el funcionamiento del mismo.

El administrador del servidor debe ser una persona con cargo de desarrollador de software y conocimiento de redes, además el sistema debe disponer de una documentación que permita realizar operaciones de mantenimiento.

3.3.5. Portabilidad

La aplicación sockets se podrá utilizar en cualquier sistema operativo, previo a la instalación del lenguaje de programación Java bajo la plataforma IDE Netbeans.

Para el servidor se debe realizar correctamente las configuraciones del Apache, Mysql y Postfix.

4. Apéndices

4.1. Modelo de casos de uso.

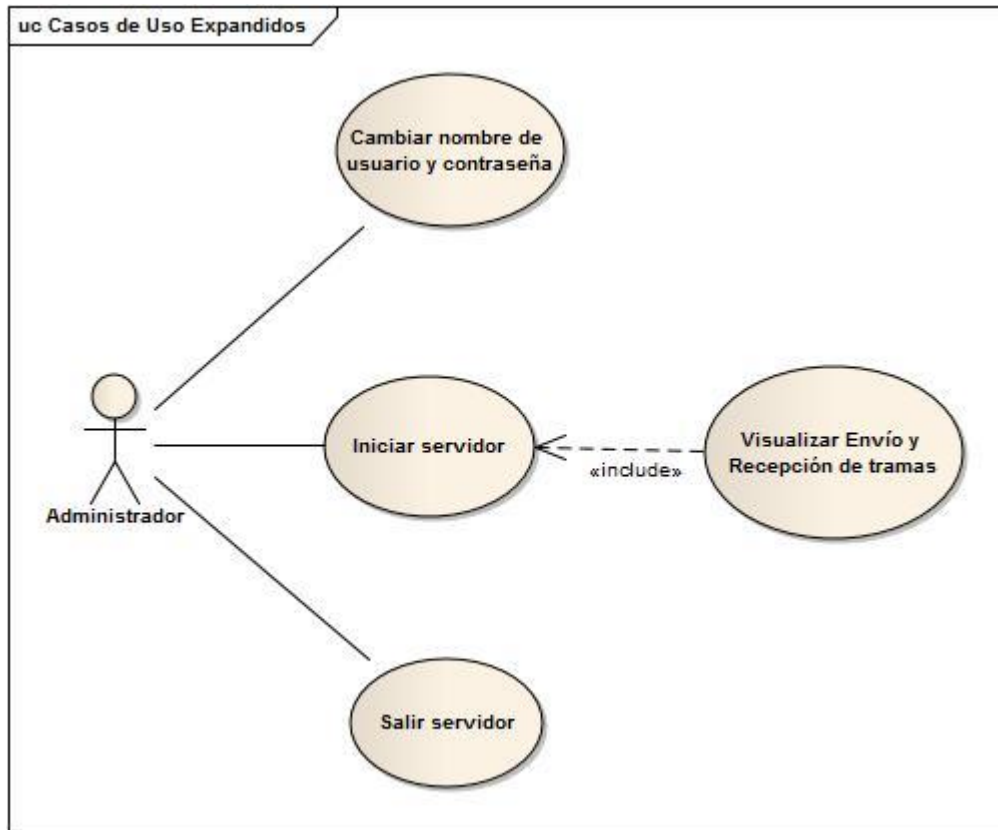


Figura N°2: Modelo de casos de uso.

Anexo 8: Hoja de la ficha de control de los registros de estacionamiento.

Anexo 9: TARJETA SIMERT

TARJETA PREPAGO

SIMERT

SISTEMA MUNICIPAL DE ESTACIONAMIENTO ROTATIVO TARIFARIO

Art. 11.- Tipo.- El tipo de pago de tarjeta tendrá un costo de USD \$ 1,50 con un período de duración de noventa días contados a partir de la fecha del primer uso, la misma que contendrá hasta seis horas de estacionamiento. La fracción de tiempo mínimo de uso para esta tarjeta será de treinta minutos.

Instrucciones:

1. Señale la hora de llegada del vehículo a la Zona del sistema y raye con una línea los casilleros correspondientes que va a utilizar.
2. Escribir en la tarjeta la fecha a ser utilizada especificando día, mes y año.
3. Ubicar la tarjeta en un lugar visible en el parabrisas.
4. Esta tarjeta tendrá una duración de 90 días a partir de su primer uso
5. Recuerde que el tiempo máximo de estacionamiento continuo en todas las zonas es de 3 horas.
6. Recuerde que si Ud. borra, corrige o sobre escribe la fecha la hora de llegada al estacionamiento del área SIMERT, se considera alteración

SANCIONES Y MULTAS:

TIEMPO	SANCIÓN
Por ausencia, uso de tarjetas falsificadas, caducadas o adulteradas	\$10
Tiempo sobrepasado de 16 a 30 min.	\$ 3
Tiempo sobrepasado de 31 a 60 min.	\$ 5
Tiempo sobrepasado de 61 a 120 min.	\$10
Tiempo sobrepasado de 121 min. a 180 min.	\$20
Tiempo sobrepasado de 181 min. en adelante y	\$40
En caso de ser remolcado pagará adicionalmente servicio de grúa	\$40

HORARIO

Zona A (Centro)	Zona C (Terminal)	Lunes a Viernes	08h00 - 18h00
		Sábados	09h00 - 13h00
Zona D (Mercado Mayorista)		Viernes, Sábado y Domingo	08h00 - 18h00
Zona B (1 al 30 Septiembre Jipiro)		Lunes a Domingo	10h00 - 24h00

Situ
SISTEMA INTEGRADO DE TRANSPORTE URBANO

GAD MUNICIPAL DE LOJA Recorre Loja de arriba abajo

USD \$ 1,50 /6HORAS/FRACCIONES DE 30 MINUTOS

SIMERT SISTEMA MUNICIPAL DE ESTACIONAMIENTO ROTATIVO TARIFADO

HORA	FECHA	TIEMPO fracciones 30min.
09:30	26-05-2014	—
	MUNICIPIO DE LOJA	
	MUNICIPIO DE LOJA	
	MUNICIPIO DE LOJA	
1:42	27-05-2014	
12:30	06-06-2014	R
	MUNICIPIO DE LOJA	
	MUNICIPIO DE LOJA	
	MUNICIPIO DE LOJA	
	MUNICIPIO DE LOJA	
	MUNICIPIO DE LOJA	
	MUNICIPIO DE LOJA	
	MUNICIPIO DE LOJA	
	MUNICIPIO DE LOJA	
	MUNICIPIO DE LOJA	


Nº 3046191

IMPRESO EN FEBRERO 20 - 2014
EMISIÓN DE: 2954001 - 3104000

Situ
SISTEMA INTEGRADO DE TRANSPORTE URBANO

Recorre Loja de arriba abajo

Cueva Sucreña
INGENIERÍA EN SISTEMAS
MUNICIPAL DE LOJA




7861000142663

Anexo 10: SERVIDOR HP PROLIANT ML 350 G6 E5606





Anexo 11: Cotización de GPRS del proveedor CLARO.

Claro																													
			<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 30%;">Fecha:</td> <td>10/05/2014</td> </tr> <tr> <td>Cotización No.:</td> <td>JE 2780</td> </tr> <tr> <td>Cliente:</td> <td>Vanessa Morocho</td> </tr> <tr> <td>Teléfono:</td> <td>968611327</td> </tr> <tr> <td>Email:</td> <td>jvamogu@gmail.com</td> </tr> </table>			Fecha:	10/05/2014	Cotización No.:	JE 2780	Cliente:	Vanessa Morocho	Teléfono:	968611327	Email:	jvamogu@gmail.com														
Fecha:	10/05/2014																												
Cotización No.:	JE 2780																												
Cliente:	Vanessa Morocho																												
Teléfono:	968611327																												
Email:	jvamogu@gmail.com																												
Servicio Cotizado: GPRS BULK CORPORATIVO																													
<p>La empresa de Telecomunicaciones Claro pone a su consideración paquetes de PLAN DE DATOS para <u>Tablets</u>.</p> <p>PLAN DE DATOS</p> <table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <thead> <tr> <th style="width: 15%;">Planes Controlados Banda Ancha <u>Movil</u></th> <th style="width: 15%;">Tarifa básica Final</th> <th style="width: 15%;">MB Incluidos</th> <th style="width: 15%;">Paquete <u>Flex</u></th> <th style="width: 15%;">Megas <u>Flex</u></th> <th style="width: 20%;">Tarifa Final MB Adicional</th> </tr> </thead> <tbody> <tr> <td>\$ 19 + <u>imp</u> (HSPA+)</td> <td>\$ 21,28</td> <td>1000</td> <td>\$2,49+<u>imp</u></td> <td>100</td> <td>\$0,112</td> </tr> <tr> <td>\$ 29 + <u>imp</u> (HSPA+)</td> <td>\$ 32,48</td> <td>2000</td> <td>\$3,49+<u>imp</u></td> <td>200</td> <td>\$0,112</td> </tr> <tr> <td>\$ 39 + <u>imp</u> (HSPA+)</td> <td>\$ 43,68</td> <td>3000</td> <td>\$3,99+<u>imp</u></td> <td>300</td> <td>\$0,112</td> </tr> </tbody> </table>						Planes Controlados Banda Ancha <u>Movil</u>	Tarifa básica Final	MB Incluidos	Paquete <u>Flex</u>	Megas <u>Flex</u>	Tarifa Final MB Adicional	\$ 19 + <u>imp</u> (HSPA+)	\$ 21,28	1000	\$2,49+ <u>imp</u>	100	\$0,112	\$ 29 + <u>imp</u> (HSPA+)	\$ 32,48	2000	\$3,49+ <u>imp</u>	200	\$0,112	\$ 39 + <u>imp</u> (HSPA+)	\$ 43,68	3000	\$3,99+ <u>imp</u>	300	\$0,112
Planes Controlados Banda Ancha <u>Movil</u>	Tarifa básica Final	MB Incluidos	Paquete <u>Flex</u>	Megas <u>Flex</u>	Tarifa Final MB Adicional																								
\$ 19 + <u>imp</u> (HSPA+)	\$ 21,28	1000	\$2,49+ <u>imp</u>	100	\$0,112																								
\$ 29 + <u>imp</u> (HSPA+)	\$ 32,48	2000	\$3,49+ <u>imp</u>	200	\$0,112																								
\$ 39 + <u>imp</u> (HSPA+)	\$ 43,68	3000	\$3,99+ <u>imp</u>	300	\$0,112																								
<p>Notas:</p> <ol style="list-style-type: none"> 1.- Cotización válida por 15 días laborables desde la fecha de emisión, después del cual los precios pueden cambiar. 2.- Tiempo de instalación dependiendo de la localidad (remitirse a propuesta del integrador) 3.- La cotización no incluye costos de obras civiles, ni antenas de mayor ganancia. 4.- Esta propuesta es solamente por el servicio GPRS. El alquiler de los equipos, soporte y monitoreo es parte de la propuesta del integrador (Porta no alquila equipos). 5.- En caso de envíos de mensajes a otras operadores el costo sera de \$0,06ctvs mas IVA. 																													
Cualquier duda o inquietud			Firma de aceptación																										
Elaborado por:	Ing. Juan Escandón Vélez		<div style="border: 1px solid black; height: 100px; width: 100%;"></div>																										
Fijo:	042693693 ext 5772																												
email:	jescandon@conecel.com																												
Celular:	91075526																												
			Fecha de aceptación:																										



Anexo 12: Cotización de GPRS del proveedor CNT.



PROFORMA

Loja, 28 de Mayo 2014

CLIENTE: SR. CESAR CALLE
CEDULA: 1104899404
TELÉFONO: 72547819- 0996955594

La Corporación Nacional de Telecomunicaciones pone en su consideración paquete de **PLAN DE DATOS** para Tablets.

PLAN DE DATOS PARA 12 MESES

<u>PAQUETE DE DATOS EMPRESA HSPA+</u>	<u>MEGAS INCLUIDAS</u>	<u>VALOR</u>
INTERNET HSPA+	512 MB	\$ 9,99
INTERNET HSPA+	750 MB	\$ 14,99
INTERNET HSPA+	1000 MB	\$ 19,99
INTERNET HSPA+	2000 MB	\$ 29,99

A estas tarifas se debe incluir el IVA.

Paquetes disponibles para 3.5G o Planes HSPA+ en tipo pospago controlado

REQUISITOS:

- Oficio con el Requerimiento
- CONTRATOS
- RUC de la Institución o Empresa
- Nombramiento del Representante Legal
- Copia de Cédula y Certificado de Votación Actualizado del Representante Legal

Atentamente,


Ing. Jorge Mogravejo
JEFE COMERCIAL CNT- LOJA
ANALISTA COMERCIAL CNT EP
LOJA 3731700 Ext 70045 -70041



Anexo 13: Cotización de GPRS del proveedor Movistar



PROFORMA

Loja, 17 de Mayo del 2014

Señores

Ciudad

El Centro de Atención y Ventas MOVISTAR, se complace en informarle sobre el **PLAN IM CTRL TABLET HPLUS EXT \$20** que disponemos para ustedes:

OFERTA COMERCIAL
 PLANES TABLETS EQUIPO PROPIO DE CLIENTE
 MESUALIDAD POR LA OFERTA DE 1000 MB, \$20 MAS IVA

Planes y Tarifas

Internet Móvil Pasajero Tablets



Tran to iPad ó Tablet y activa INTERNET MÓVIL

	Tarifa básica	MB	Minutos TD	SMS TD	MB Adicionales	Minut@ edición al	SMS* Adicionales
IM CTRL TABLET HPLUS EXT \$20	\$20	1000	40	40	\$0.10	\$0.15	\$0.06

- 1000 MB
- 40 Minutos
- 40 Mensajes de texto a toda la operadora.
- Promoción por el mes de las madres: Doble en megas, minutos y mensajes por un año
- 2000 MB
- 80 Minutos
- 80 Mensajes de texto a toda la operadora.

Atentamente;
LOURDES CELI
 EJECUTIVO DE VENTAS
 VIMOTECH - LOJA
 Local: 072585572
 Cel: 0993688892
 E-mail: loceli.vimotech@gmail.com

Anexo 14: Certificación del Coordinador del SIMERT.

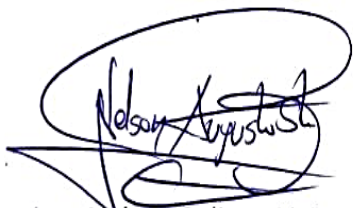
Ing. Nelson Salinas Livisaca
ADMINISTRADOR DEL SIMERT (E)

CERTIFICO:

Que los señores César Danilo Calle Loja y Julia Vanessa Morocho Guadalima realizaron las pruebas correspondientes a la aplicación web del SIMERT (Sistema de Estacionamiento Rotativo Tarifado) "SIMERTWEB", en las instalaciones del mismo, los días 15, 21 y 23 de enero del presente año, quienes aplicaron una encuesta a los usuarios del sistema para conocer el grado de aceptación.

Atentamente.

Loja, 23 de enero del 2014



Ing. Nelson Salinas Livisaca

ADMINISTRADOR DEL SIMERT (E)

Anexo 15: Licencias aplicadas al Proyecto

Licencia Creative Commons

Atribución-No Comercial-Compartir Igual CC BY-NC-SA

Esta licencia permite a otros distribuir, re mezclar, retocar, y crear a partir de tu obra de modo no comercial, siempre y cuando den crédito y licencien sus nuevas creaciones bajo las mismas condiciones.



Atribución

(BY)

El beneficiario de la licencia tiene el derecho de copiar, distribuir, exhibir y representar la obra y hacer obras derivadas siempre y cuando reconozca y cite la obra de la forma especificada por el autor o el licenciante.



No Comercial

(NC)

El beneficiario de la licencia tiene el derecho de copiar, distribuir, exhibir y representar la obra y hacer obras derivadas para fines no comerciales.



No Derivadas

(ND)

El beneficiario de la licencia solamente tiene el derecho de copiar, distribuir, exhibir y representar copias literales de la obra y no tiene el derecho de producir obras derivadas.



Compartir Igual

(SA)

El beneficiario de la licencia tiene el derecho de distribuir obras derivadas bajo una licencia idéntica a la licencia que regula la obra original.

EL presente proyecto se encuentra bajo la licencia Creative Commons

Desarrollo del proyecto

Desarrollo del proyecto: “Tecnificación Del Sistema Municipal De Estacionamiento Rotativo Tarifado (SIMERT) De La Ciudad De Loja, Módulo Conectividad De La Red De Datos Del SIMERT Utilizando La Tecnología GPRS”

Licencia



Tecnificación del SIMERT, módulo conectividad de la red de datos del SIMERT utilizando la tecnología GPRS” por César Calle, Vanessa Morocho se distribuye bajo una [Licencia Creative Commons Atribución-No Comercial-Compartir Igual 4.0 License](#).

Anexo 16: Encuesta realizada a la Ingeniera encargada de la Técnica del SITU sobre la accesibilidad, funcionalidad y usabilidad de la aplicación sockets.



UNIVERSIDAD NACIONAL DE LOJA
ÁREA DE LA ENERGÍA, LAS INDUSTRIAS Y LOS RECURSOS
NATURALES NO RENOVABLES
Carrera de Ingeniería en Sistemas

Proyecto: Tecnificación del Sistema Municipal De Estacionamiento Rotativo Tarifado (SIMERT) de la Ciudad de Loja, módulo conectividad de la red de datos del SIMERT utilizando la tecnología GPRS.

Dígnese a contestar con absoluta sinceridad los aspectos que a continuación se detallan, sus criterios y comentarios son estrictamente confidenciales entre usted y los investigadores, los mismos que servirán para comprobar y validar las funciones de la aplicación sockets "SERVER-SIMERT" del Sistema Municipal de Estacionamiento Rotativo Tarifado (SIMERT).

Nombre: Tania Verónica Vásquez Romero

Cargo: Técnica de la Unidad de Tránsito

Fecha: 23/05/2014

1. ACCESIBILIDAD.

1.1. ¿Tuvo algún inconveniente al ejecutar la aplicación sockets "SERVER-SIMERT"?

SI ()

NO (x)

Porque?

Porque recibí correctamente las instrucciones y es de fácil manejo

1.2. ¿Tuvo algún inconveniente al ingresar y cerrar la aplicación sockets "SERVER-SIMERT" con el nombre de usuario y contraseña?

SI ()

NO (x)

Porque?

Porque los componentes son sencillos y de fácil comprensión

2. FUNCIONALIDAD.

2.1. ¿Presentó algún error la aplicación sockets “SERVER-SIMERT” durante el envío y recepción de los siguientes registros de estacionamiento vehicular?

TAREA	SI	NO
Registro enviar zonas a los controladores.		x
Registro ingresar cliente exonerado o recarga.		x
Registro ingresar multa por recarga.		x
Registro validar código tarjeta.		x
Registro modificar espacios de tarjeta.		x
Registro ingresar personal del municipio.		x
Registro ingresar multa por tarjeta.		x
Registro ingresar código de tarjeta.		x
Registro ingresar nueva tarjeta.		x
Registro ingresar cliente recarga estacionado.		x
Registro ingresar datos posicionales.		x
Registro eliminar controlador de sesiones.		x
Registro ingresar sincronización de multas.		y

Porque? *Porque la aplicación sockets realizó correctamente los procesos de cada registro recibido*

2.2. ¿Considera usted que la aplicación sockets “SERVER_SIMERT” es rápido en el envío y recepción de datos durante la comunicación con los clientes?

SI (x) NO ()

Porque? *Porque se ha realizado las pruebas con los controladores para el envío de datos y no presentó error alguno y no se ha colgado la aplicación*

2.3. ¿La aplicación sockets “SERVER-SIMERT” cumple con todas las especificaciones requeridas, manifestadas al inicio y durante la ejecución del proyecto?

SI (x) NO ()

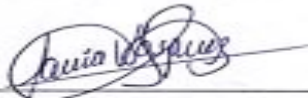
Porque? *Porque estuvimos en contacto con los asistentes*

3. USABILIDAD

3.1. ¿Considera que la aplicación "SERVER-SIMERT" tiene una interfaz amigable y de fácil uso para su ejecución?

SI (X) NO ()

Porque? *Porque he observado y manipulado la aplicación*
.....
.....



Ing. Tania Verónica Vázquez Romero
Técnica de la Unidad de Tránsito

Anexo 17: Certificado de traducción del Resumen del presente proyecto.

DRA. MG. SC. ROCIO ESPERANZA PEÑARANDA REQUELME, PROFESORA DE IDIOMA INGLES,

CERTIFICA:

Haber traducido del idioma Español al Idioma Inglés, el RESUMEN del trabajo de investigación: "TECNIFICACION DEL SISTEMA MUNICIPAL DE ESTACIONAMIENTO ROTATIVO TARIFADO (SIMERT) DE LA CIUDAD DE LOJA MODULO CONECTIVIDAD DE LA RED DE DATOS DEL SIMERT UTILIZANDO LA TECNOLOGÍA GPRS" cuya autoría es de JULIA VANESSA MOROCHO GUADALIMA Y DE CESAR DANILO CALLE LOJA.- Loja, 30 de Junio de 2014.



Dra. Mg. Sc. Rocio E. Peñaranda R.
PROFESORA DE IDIOMA INGLES

Anexo 18: Anteproyecto



**UNIVERSIDAD
NACIONAL
DE LOJA**



Área de la Energía, las Industrias y los Recursos Naturales No Renovables

CARRERA DE INGENIERÍA EN SISTEMAS

**“TECNIFICACIÓN DEL SISTEMA MUNICIPAL DE
ESTACIONAMIENTO ROTATIVO TARIFADO
(SIMERT) DE LA CIUDAD DE LOJA, MÓDULO
CONECTIVIDAD DE LA RED DE DATOS DEL
SIMERT UTILIZANDO LA TECNOLOGÍA GPRS”**

TESIS DE GRADO PREVIA A LA
OBTENCIÓN DEL TÍTULO DE
INGENIEROS EN SISTEMAS.

AUTORES:

*César Danilo Calle Loja
Julia Vanessa Morocho Guadalima*

DIRECTOR:

Ing. Hernán Leonardo Torres Carrión, Mg. Sc.

Loja-Ecuador

2014

A. TEMA

“TECNIFICACIÓN DEL SISTEMA MUNICIPAL DE ESTACIONAMIENTO ROTATIVO TARIFADO (SIMERT) DE LA CIUDAD DE LOJA, MÓDULO CONECTIVIDAD DE LA RED DE DATOS DEL SIMERT UTILIZANDO LA TECNOLOGÍA GPRS”



ÍNDICE

A. TEMA	216
B. PROBLEMÁTICA	218
1. SITUACIÓN PROBLEMÁTICA.....	218
2. PROBLEMA DE INVESTIGACIÓN.....	221
C. JUSTIFICACIÓN	220
D. OBJETIVOS.....	222
OBJETIVO GENERAL	221.
OBJETIVOS ESPECIFICOS	221
J. BIBLIOGRAFÍA.....	261

B. PROBLEMÁTICA

1. SITUACIÓN PROBLEMÁTICA

El Sistema Municipal De Estacionamiento Rotativo Tarifado (SIMERT), se creó mediante ordenanza el dieciocho de marzo del dos mil dos, con el objetivo de devolverles a todos los usuarios el derecho a utilizar la vía pública en forma organizada y ordenada, a través de la generación de una oferta permanente y continua de espacios libres para estacionamiento y reducir la contaminación ambiental provocada por el flujo continuo de vehículos.

Este sistema funciona en la ciudad de Loja desde el 3 de Mayo del 2002, es desde Enero del 2008 que viene funcionando como un área independientemente de la Unidad de Tránsito. Sus oficinas están situadas en el edificio de la Policía Municipal, en las calles Bernardo Valdivieso y Colón esq. Luego de la reestructuración, la dependencia cuenta con 50 personas entre controladores, inspectores y administrativos; parte del cambio fue la entrega de nuevos uniformes, adquisición de vehículos, motos y maquinaria para la permanente señalización. El personal que labora en esta área, a más de velar por el correcto uso de las tarjetas, colabora en la seguridad de los vehículos, ya que cuenta con un sistema de comunicación permanente que les permite solicitar ayuda en caso de ser necesario³.

El Sistema Municipal De Estacionamiento Rotativo Tarifado (SIMERT), no utiliza tecnología avanzada que potencie su funcionamiento y servicio a la ciudadanía lojana, entre los problemas que afronta son los siguientes:

- Inexistencia de algún tipo de conexión entre los diferentes departamentos del SIMERT que facilite y asegure el control y administración de la información.
- Existencia de recursos informáticos desactualizados debido a que actualmente se utiliza un mecanismo manual para la manipulación de la información.
- Falta de seguridad y agilidad en el almacenamiento y recopilación de información y en a acerca de los registros de estacionamiento vehicular y sanción de multas ya que los controladores del SIMERT lo realizan de forma manual en hojas impresas denominadas “Fichas de Control”.

³ Memorando N°715-SGM-2012g. Ordenanza que regula el sistema municipal de estacionamiento rotativo tarifado del cantón Loja, “SIMERT”.

- Pérdida económica para el SIMERT debido a que los controladores no pueden emitir la multa a todos los usuarios infractores por no llegar en el tiempo adecuado cuando el automóvil ha cumplido su periodo de estacionamiento.
- Falta de vigilancia y control del personal operativo del SIMERT para que cumplan con sus actividades en el horario y zona establecida por la administración.
- Falta de fiabilidad y agilidad para realizar los reportes de las tasas de rotación, estacionamiento y ocupación vehicular ya que la administración del SIMERT las realiza manualmente a través de la información almacenada en las Fichas de Control con el uso de las fórmulas establecidas por la institución dificultando su proceso.
- Aumento de tráfico vehicular debido a que los usuarios del SIMERT recorren constantemente las áreas de estacionamiento hasta encontrar algún espacio disponible.
- No existe mecanismos que apoyen a la toma de decisiones debido a la carencia de datos reales en que la administración pueda basarse.
- Ausencia de políticas de seguridad para el control de acceso a la información.

De esta manera analizando cada uno de los problemas por falta de seguridad y agilidad de la información que opera el SIMERT se ha creído conveniente desarrollar el proyecto denominado “TECNIFICACIÓN DEL SISTEMA MUNICIPAL DE ESTACIONAMIENTO ROTATIVO TARIFADO (SIMERT) DE LA CIUDAD DE LOJA, MÓDULO CONECTIVIDAD DE LA RED DE DATOS DEL SIMERT UTILIZANDO LA TECNOLOGÍA GPRS”.

2. PROBLEMA DE INVESTIGACIÓN

¿La conectividad de la red de datos del SIMERT mediante la aplicación de la tecnología GPRS permitirá agilizar y potenciar la seguridad en la información generada por cada uno de los procesos y con ello, brindar un servicio eficaz y eficiente?

C. JUSTIFICACIÓN

Este proyecto se justifica científicamente, porque se ha creado la necesidad de implantar nuevos canales de comunicación en el Sistema Municipal De Estacionamiento Rotativo Tarifado (SIMERT), que faciliten la realización de las diversas actividades y servicios de una manera rápida, segura y confiable para la comunidad lojana, utilizando recursos científicos como: Implementación de la arquitectura cliente-servidor para llevar a cabo la conectividad de la red de datos del SIMERT (Ver anexo N°2), a través de la implementación de un servidor, creación de una aplicación sockets para el mismo denominada “SERVER-SIMERT”, uso de la tecnología GPRS bajo el protocolo TCP/IP, direccionamiento IP y utilizando la clase ServerSockets dentro de la aplicación.

La Universidad Nacional de Loja ha formado profesionales de calidad con excelentes conocimientos para resolver problemas que afronta diariamente la sociedad.

Como estudiantes del Área de la Energía, las Industrias y los Recursos Naturales no Renovables de la Carrera de Ingeniería en Sistemas se ha creído conveniente efectuar el presente proyecto que consiste en “Tecnificación del Sistema Municipal De Estacionamiento Rotativo Tarifado (SIMERT) de la Ciudad de Loja, módulo conectividad de la red de datos del SIMERT utilizando la tecnología GPRS”.

Como entes participantes en el desarrollo de la investigación, se tiene la capacidad de poner en práctica los conocimientos tanto teóricos como prácticos adquiridos a lo largo de la carrera y valiéndose de metodologías apropiadas para la recolección de la información, se llevara a cabo dicho proyecto que durante su desarrollo se obtendrá nuevas experiencias para plasmarlos en la vida profesional.

Este proyecto será de mucha utilidad para el Municipio de la Ciudad de Loja, ya que facilitará el uso del Sistema Municipal De Estacionamiento Rotativo Tarifado (SIMERT), mejorando la administración y control del servicio de estacionamiento vehicular.

Desde el punto de vista Técnico-Tecnológico es factible realizar el proyecto porque se cuenta con el apoyo de los ingenieros del SIMERT para cualquier información y documentación.

Además se cuenta con las herramientas Técnico-Tecnológicas necesarias tanto para el desarrollo de la aplicación sockets denominada “SERVER-SIMERT” y para la implementación del servidor como: el lenguaje de programación Java bajo la plataforma virtual IDE NetBeans 8.0, la clase ServerSockets, el conector para la base de datos - java 5.1.6 -bin.jar y las librerías Jasypt 1.9.1, Commons-dbc-1.4 y Commons-pool-1.6, el sistema operativo Ubuntu Server Lucid Lynx 10.04 bajo la plataforma libre LAMP, la base de datos Mysql v. 5.5.2., Phpmyadmin v. 5.1.72, Postfix v. 2.7 y el apache 2.4.2. El presente proyecto se lo podrá realizar sin novedad alguna debido a que se cuenta con el apoyo económico de cada uno de los integrantes del equipo de trabajo y del Municipio de la ciudad de Loja (Ver Anexo N°1), además se cuenta con todos los recursos y tiempo necesario y especialmente interés para su desarrollo que potenciará el servicio de estacionamiento vehicular a la ciudadanía lojana. Este proyecto es factible y viable por los puntos justificados anteriormente tanto en el aspecto, científico, académico-social, técnico-tecnológico y económico.

D. OBJETIVOS

OBJETIVO GENERAL

- Realizar la Tecnificación del Sistema Municipal De Estacionamiento Rotativo Tarifado (SIMERT) de la Ciudad de Loja, módulo conectividad de la red de datos del SIMERT utilizando la tecnología GPRS.

OBJETIVOS ESPECÍFICOS

- Realizar un análisis de requisitos para llevar a cabo la conectividad de la red de datos entre las diferentes subáreas del SIMERT.
- Realizar el diseño de la arquitectura cliente-servidor para establecer la comunicación.
- Implementar la arquitectura cliente-servidor para la conectividad de la red de datos del SIMERT.
- Realizar las pruebas para verificar la conectividad en la red de datos del SIMERT.