

UNIVERSIDAD POLITÉCNICA SALESIANA
SEDE CUENCA

CARRERA DE INGENIERÍA ELECTRÓNICA

*Trabajo de titulación previo a la obtención
del título de Ingeniero Electrónico*

**PROYECTO TÉCNICO CON ENFOQUE SOCIAL:
“PLATAFORMA PARA CONTROL Y GESTIÓN DE
INTERRUPCIONES DEL SERVICIO DE TELEFONÍA
MÓVIL AVANZADA PARA LA ARCOTEL”**

AUTORES:

CHRISTIAN RAFAEL MARCA GUARACA
ALEXIS BOLIVAR CRESPO CARREÑO

TUTOR:

ING. EDGAR OCHOA FIGEROA MG.T

CUENCA – ECUADOR

2019

CESIÓN DE DERECHOS DE AUTOR

Nosotros, Christian Rafael Marca Guaraca con documento de identificación N° 0105137566 y Alexis Bolívar Crespo Carreño con documento de identificación N° 1400998363, manifestamos nuestra voluntad y cedemos a la Universidad Politécnica Salesiana la titularidad sobre los derechos patrimoniales en virtud de que somos autores del trabajo de titulación: **“PLATAFORMA PARA CONTROL Y GESTIÓN DE INTERRUPCIONES DEL SERVICIO DE TELEFONÍA MÓVIL AVANZADA PARA LA ARCOTEL”**, mismo que ha sido desarrollado para optar por el título de: *Ingeniero Electrónico*, en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En aplicación a lo determinado en la Ley de Propiedad Intelectual, en mi condición de autor me reservo los derechos morales de la obra antes citada. En concordancia, suscribo este documento en el momento que hago entrega del trabajo final en formato impreso y digital a la Biblioteca de la Universidad Politécnica Salesiana.

Cuenca, 20 de febrero del 2019

Christian Rafael Marca Guaraca

CI: 0105137566

AUTOR

Alexis Bolívar Crespo Carreño

CI: 1400998363

AUTOR

CERTIFICACIÓN

Yo declaro que bajo mi tutoría fue desarrollado el trabajo de titulación: **“PLATAFORMA PARA CONTROL Y GESTIÓN DE INTERRUPCIONES DEL SERVICIO DE TELEFONÍA MÓVIL AVANZADA PARA LA ARCOTEL”**, realizado por Christian Rafael Marca Guaraca y Alexis Bolívar Crespo Carreño, obteniendo el *Proyecto Técnico con enfoque social* que cumple con todos los requisitos estipulados por la Universidad Politécnica Salesiana.

Cuenca, 20 de Febrero del 2019



Ing. Edgar Ochoa Figueroa

CI: 0102134574

TUTOR DEL TRABAJO DE TITULACIÓN

DECLARATORIA DE RESPONSABILIDAD

Nosotros, Christian Rafael Marca Guaraca con número de cédula 0105137566 y Alexis Bolívar Crespo Carreño con número de cédula 1400998363, autores del trabajo de titulación: **“PLATAFORMA PARA CONTROL Y GESTIÓN DE INTERRUPCIONES DEL SERVICIO DE TELEFONÍA MÓVIL AVANZADA PARA LA ARCOTEL”**, certificamos que el total contenido del *Proyecto Técnico con enfoque social*, es de nuestra exclusiva responsabilidad y autoría

Cuenca, 20 de febrero del 2019

Christian Rafael Marca Guaraca

CI: 0105137566

AUTOR

Alexis Bolívar Crespo Carreño

CI:1400998363

AUTOR

AGRADECIMIENTOS

Quiero expresar mi más profundo agradecimiento a la AGENCIA DE CONTROL Y REGULACIÓN DE LAS TELECOMUNICACIONES y a su cuerpo humano por confiar en mí, permitirme realizar todo el proceso técnico y humano que dio paso al poder realizar el presente proyecto dentro de su establecimiento.

De manera especial quiero expresar mi sincero agradecimiento a los ingenieros Esteban Andrade y Edgar Ochoa, por ser los principales impulsores y colaboradores en todo el proceso del proyecto, gracias a su dirección, enseñanza y colaboración, hicieron posible el desarrollo del proyecto

Finalmente, quisiera agradecer a mis padres quienes han sido la piedra angular de este largo camino para cumplir con este objetivo tan alto, gracias a sus consejos, valores y principios inculcados han hecho que este sueño sea realidad.

Christian Rafael Marca Guaraca

Gracias a mi familia por todo el apoyo y la confianza y a todas las personas que hicieron posible este proyecto.

Alexis Bolívar Crespo Carreño

DEDICATORIAS

A mis padres

Todo el esfuerzo y la dedicación que han logrado que alcance tal objetivo, son pruebas fervientes del fruto que hoy cosechan por su arduo camino para formarme como persona, hoy con la cara en alto y muy orgullosamente me permito expresarme a ustedes María Mercedes Guaraca Guaraca y Ángel Rodrigo Marca Plasencia, que son mi mayor orgullo y los responsables de mis mayores logros y gracias ustedes he concluido con esta meta, a falta de palabras me permito decirle infinitas gracias.

Christian Rafael Marca Guaraca

A mi familia.

Alexis Bolívar Crespo Carreño

ÍNDICE GENERAL

AGRADECIMIENTOS.....	I
DEDICATORIAS.....	II
ÍNDICE GENERAL.....	III
ÍNDICE DE FIGURAS.....	VI
ÍNDICE DE TABLAS.....	VIII
RESUMEN.....	IX
INTRODUCCIÓN.....	X
ANTECEDENTES DEL PROBLEMA DE ESTUDIO.....	XII
JUSTIFICACIÓN (IMPORTANCIA Y ALCANCES).....	XIII
OBJETIVOS.....	XIV
OBJETIVO GENERAL.....	XIV
OBJETIVOS ESPECÍFICO.....	XIV
Capítulo 1: Estado del Arte Y Fundamentación Teórica.....	1
1.1 Trabajos previos.....	1
1.1.1 Aplicativo para notificación de interrupciones del servicio de telefonía fija.....	2
1.1.2 Señal Móvil Ecuador.....	4
1.1.3 Infoantenas.....	6
1.2 Análisis de datos.....	8
1.2.1 Variables.....	8
1.2.2 Codificación de variables.....	9
1.2.3 Estadísticas descriptivas.....	10
1.2.4 Prueba Chi-cuadrado χ^2 de independencia.....	10
1.3 Herramientas de Desarrollo.....	13
1.3.1 Plataforma Web.....	13
1.3.2 Historia de los Servicios Web.....	13
1.3.3 Aplicaciones en Internet y Programación en la Red.....	14
1.3.4 Modelos de Plataforma Web.....	16
1.3.5 Tecnologías y Librerías Usadas en el Desarrollo de la Plataforma.....	19
CAPÍTULO 2: MARCO METODOLÓGICO.....	23
2.1 Regulación del Servicio Móvil Avanzado en el Ecuador.....	23
2.1.1 Competencias de la ARCOTEL.....	24
2.1.2 Disposición de la ARCOTEL.....	25

2.1.3	Derechos y Obligaciones de Usuarios y Prestadores de Servicios del Sistema de Telefonía Móvil Avanzada	27
2.1.4	Régimen Sancionatorio	29
2.1.5	Normativa de Interrupciones del Servicio de Telefonía Móvil Avanzada (SMA)	31
2.1.6	Servicio Móvil Avanzado en el Ecuador	33
2.2	Implementación de Plataforma Web	33
2.2.1	Estructura de la Plataforma	34
2.2.2	Diseño de Base de Datos.....	44
2.2.3	Gestión de Sesión del Aplicativo	45
2.2.4	Estado de Estaciones Base (BSs).....	48
2.2.5	Consumo de Servicios.....	50
2.2.6	Configuración y Ejecución de la Plataforma	54
2.2.7	Proceso de Uso de la Plataforma.....	64
2.3	Determinación de métodos para análisis estadístico.....	65
2.3.1	Procedimiento de análisis.....	66
CAPÍTULO 3: IMPLEMENTACIÓN Y ANÁLISIS DE RESULTADOS		68
3.	tmp	68
3.1	Descripción de la Plataforma y componentes del Sistema de Manejo de Interrupciones del SMA	68
3.2	Componentes Generales de la Plataforma.....	69
3.2.1	Registro e Inicio de Sesión de Usuario	69
3.2.2	Ingreso de Interrupción	72
3.2.3	Visualización de Interrupciones.....	75
CAPÍTULO 4: ANÁLISIS ESTADÍSTICO DE INTERRUPCIONES DEL SERVICIO MÓVIL AVANZADO.....		84
4.1	Descripción de la Plataforma y componentes del Sistema de Manejo de Interrupciones del SMA	84
4.2	Descripción de la información	84
4.3	Codificación de variables.....	85
4.4	Estadísticas descriptivas.....	86
4.5	Método chi-cuadrado de independencia.....	93
4.4.1	Hipótesis.....	93
4.4.2	Valor de α	93
4.4.3	Valor crítico	93
4.4.4	Tabulación cruzada	94
4.4.5	Prueba chi-cuadrado de independencia y nivel de asociación	96
4.4.6	Prueba post hoc	97
4.4.7	Reporte de resultados	97

CAPÍTULO 5: CONCLUSIONES Y RECOMENDACIONES	98
REFERENCIAS BIBLIOGRÁFICAS	100
APÉNDICES.....	105
APÉNDICE A: ESTRUCTURA DE BASE DE DATOS	105
APÉNDICE B: DIAGRAMA DE ÁRBOL DE COMPONENTES SMA_APP.....	105
APÉNDICE C: DIAGRAMA DE PROCESOS PARA MANEJO DE APLICATIVO.....	105
APÉNDICE D: ESTRUCTURA DE ARCHIVOS SMA_API.....	105
APÉNDICE E: ÁRBOL DE ESTADOS GLOBAL	105
Anexos	106
Anexo 1: Valores críticos de χ^2	106

ÍNDICE DE FIGURAS

Figura 1-1. Plataforma Señal Móvil Ecuador.	5
Figura 1-2. Plataforma Señal Móvil Ecuador en iPad.....	5
Figura 1-3. Plataforma Señal Móvil Ecuador Smartphone.	6
Figura 1-4. Plataforma Infoantenas.....	7
Figura 1-5. Radiobases mostradas en plataforma Infoantenas.....	8
Figura 1-6. Distribución chi-cuadrado.	11
Figura 2-1 Agendas Zonales dispuestas por la SENPLADES.	26
Figura 2-2 Estructura de aplicación para manejo del estado global.....	35
Figura 2-3 Flujo es estado en aplicación React.....	37
Figura 2-4 Diagrama de árbol de componentes del aplicativo SMA_APP.....	39
Figura 2-5 Flujo de estado de aplicación, cuando se usa la plataforma.	40
Figura 2-6 Diagrama de árbol de componentes del aplicativo BSS.....	41
Figura 2-7 Estructura principal propuesta para el aplicativo SMA_API.	42
Figura 2-8 Estructura propuesta de la carpeta Routes.....	42
Figura 2-9 Estructura propuesta de la carpeta Services.	43
Figura 2-10 Estructura propuesta de la carpeta Core.	43
Figura 2-11 Estructura propuesta de la carpeta DataBase.	43
Figura 2-12 Diseño de la base de datos propuesto.	44
Figura 2-13 Diseño de base de datos para restricción a la información según rol de usuario.....	46
Figura 2-14 Base de datos al lado del cliente mediante IndexedDB.....	49
Figura 2-15 Flujo de servicios de los aplicativos.....	50
Figura 2-16 Flujo de datos de los aplicativos.....	51
Figura 2-17 Agrupación de Servicios de SMA_API propuesta en contenedor de software Docker.	57
Figura 2-18 Ejecución del contenedor de software Docker para iniciar los servicios de SMA_API.....	62
Figura 2-19 Compilación correcta de contenedor de software Docker.....	63
Figura 2-20 Ingreso a la línea de comandos de la imagen de PostgreSQL.....	63
Figura 2-21 Acondicionamiento correcto de la base de datos.	64
Figura 2-22 Esquema simplificado de funcionalidad de aplicativo.	64
Figura 2-23. Diagrama de flujo para desarrollo de análisis.	67

Figura 3-1 Esquema de procesos de la plataforma.....	69
Figura 3-2 Página de administrador para registro de usuarios.	70
Figura 3-3 Correo electrónico entregado al usuario luego del registro.....	70
Figura 3-4 Landing page (Página de Inicio) de la plataforma.	71
Figura 3-5 Modal de actualización de datos de usuario.	72
Figura 3-6 Aplicación de registro de interrupción.	73
Figura 3-7 Campos del registro de interrupción completados.	74
Figura 3-8 Modal de resumen de interrupción e ingreso de destinatarios de correo electrónico.	75
Figura 3-9 Correo Electrónico de registro de interrupción recibido.	75
Figura 3-10 Aplicativo de tabulación dinámica para visualización de interrupciones.	76
Figura 3-11 Aplicativo de tablero dinámico para visualización de interrupciones....	78
Figura 3-12 Visualizador de interrupción del operador.	80
Figura 3-13 Visualizador y administrador de interrupción de funcionario.	80
Figura 3-14 Visualizador de radiobases.	81
Figura 3-15 Modal de información de radiobases.....	82
Figura 3-16 Cuadro de sugerencias para ubicación de estructuras.	83
Figura 3-17 Herramientas de filtrado para aplicativo de mapas.	83
Figura 4-1. Cantidad de interrupciones.	87
Figura 4-2. Interrupciones no programadas según la causa.	87
Figura 4-3. Interrupciones programadas según la causa.	88
Figura 4-4. Interrupciones no programadas por mes.	89
Figura 4-5. Interrupciones programadas por mes.	89
Figura 4-6. Interrupciones no programadas por región.....	90
Figura 4-7. Interrupciones programadas por región.....	91
Figura 4-8. Interrupciones no programadas en región Amazónica.	92
Figura 4-9. Interrupciones programadas en región Sierra.....	92

ÍNDICE DE TABLAS

Tabla 1-1. Codificación de variable causa.	10
Tabla 2-1 Distribución Zonal de la ARCOTEL.	27
Tabla 2-2 Sanciones para las y los prestadores de Servicio de Telecomunicaciones.	31
Tabla 2-3 Monto de referencia para sanciones.	31
Tabla 2-4. Tipos de variables de información de interrupciones.	65
Tabla 4-1. Cantidad de radiobases a nivel de coordinación zonal 6.	85
Tabla 4-2. Variables y tipos.	85
Tabla 4-3. Codificación de variable Tipo.	85
Tabla 4-4. Codificación de variable causa.	86
Tabla 4-5 Codificación de variable región.	86
Tabla 4-6. Duración de interrupciones.	91
Tabla 4-7. Variables modificadas.	94
Tabla 4-8. Valores críticos.	94
Tabla 4-9. Tabulación cruzada Región*Causa.	95
Tabla 4-10. Frecuencias.	96
Tabla 4-11. Valor chi-cuadrado y Cramer phi.	96
Tabla 4-12. Valores chi-cuadrado, crítico y calculado.	96
Tabla 4-13. Valores residuales.	97

RESUMEN

En el presente proyecto se planteó y desarrollo una plataforma para el manejo de las interrupciones del Servicio Móvil Avanzado, el proceso de control de interrupciones es llevado a cabo por la Agencia de Regulación y Control de las Telecomunicaciones (ARCOTEL) que es la entidad que regula las telecomunicaciones del país, y es de suma importancia que este proceso sea realizado de manera eficiente ya que la cantidad de usuarios del mencionado servicio representan una gran cantidad de la población.

La normativa que rige el manejo de las interrupciones se encuentra establecido en varios artículos de la Ley Orgánica de Telecomunicaciones, según sea el tipo de interrupción se tiene un procedimiento, así también tiempos establecidos para cada estado de las interrupciones, esto se analizó con el fin de poder establecer los parámetros y características que la plataforma deberá cumplir para que los usuarios puedan llevar a cabo las operaciones de ingreso y control.

La plataforma permitirá el ingreso de interrupciones por parte de personal de las operadoras de telefonía celular, así como el manejo y control de las mismas, funciones que estarán a cargo de personal de la ARCOTEL, para ello se utilizó tecnologías de desarrollo WEB que permiten una plataforma versátil y de simple acceso mediante un navegador WEB que facilita la distribución de software a los usuarios.

También se desarrolló una plataforma de visualización de radiobases para conocer la localidad de cada una de ellas de una manera más intuitiva mediante el uso de mapas, además de que esta plataforma de visualización será también utilizada por la plataforma de interrupciones para visualizar las radiobases que tengan alguna interrupción en sus servicios, brindando una interacción más amigable para el usuario.

Finalmente se desarrolló un análisis estadístico de los datos de las interrupciones producidas en años pasados, para establecer un marco general de la distribución de las interrupciones para determinar alguna relación entre las variables que están presentes en estos eventos.

INTRODUCCIÓN

Solo en las cuatro provincias a cargo de la Coordinación Zonal 6 de la ARCOTEL se produjeron 127 interrupciones entre el año 2016 y 2018, que representan menos de una cuarta parte de las provincias del total nacional, cada interrupción debe ser reportada por la operadora involucrada, esto mediante la notificación vía correo electrónico a los encargados del manejo de las interrupciones del Servicio Móvil Avanzado en la ARCOTEL, posteriormente cuando la interrupción ha sido finalizada los funcionarios que realizan el seguimiento deben realizar un informe con los detalles de la interrupción. Este proceso debe ser realizado para cada una de las interrupciones que se producen por lo tanto se vuelve una tarea repetitiva, y dado que todo el proceso se lo realiza de manera manual no existe una base de datos a detalle de todas las interrupciones producidas no se puede realizar un seguimiento eficaz de la continuidad del Servicio de Telefonía Móvil Avanzado

En base a este problema, se planteó la plataforma de manejo de interrupciones, que busca facilitar todo el proceso que se lleva a cabo, para plantear la funcionalidad de la plataforma fue necesario realizar una revisión de la normativa que rige tanto los procedimientos como sanciones que se aplican dependiendo del tipo de interrupción, programadas y no programadas, las sanciones se establecen en función a los tiempos plazo dentro de los cuales se debe ingresar la interrupción y el plazo para el reporte final.

La plataforma permitirá la creación de cuentas para dos grupos diferentes, tanto para empleados de operadoras de telefonía móvil y funcionarios de la ARCOTEL, cada grupo de usuarios tendrán funciones limitadas según sea su grupo; Operadoras de telefonía móvil, podrán ingresar y observar un historial de las interrupciones pertenecientes solo a propia operadora, para el caso de funcionarios, podrán realizar el control de tiempos y visualizar un registro completo de todas las interrupciones de las operadoras que operan en el país, además de la generación del informe correspondiente por cada interrupción, en base a los artículos definidos en la Ley Orgánica de Telecomunicaciones.

También se mostrará el desarrollo de una plataforma para la visualización de radiobases de todo el país, de modo que se pueda acceder a esta información para conocer cuáles han sido afectadas por las interrupciones.

ANTECEDENTES DEL PROBLEMA DE ESTUDIO

En el Ecuador el servicio de telefonía móvil avanzada hasta noviembre del 2018 se registraron 15'688.683 líneas activas según las estadísticas de la ARCOTEL [1], las cuales son servidas por las operadoras CLARO (CONECEL), MOVISTAR (OTECCEL) y CNT (CNT EP), para brindar el servicio, en el territorio ecuatoriano para octubre del 2018 se registraron 16678 Radio Bases (BSs), de las cuales, 2405 le corresponden a CNT EP, 9021 a CONECEL y 5252 a OTECEL [2], dispuestas a lo largo del país. La Agencia de Regulación y Control de las Telecomunicaciones (ARCOTEL) es la entidad encargada de la Regulación y Control de Servicios de Telecomunicaciones, tiene la obligación de regular la calidad y disponibilidad de servicio de Móvil Avanzado (SMA) en el Ecuador rigiéndose en lo dispuesto en la Ley Orgánica de Telecomunicaciones (LOT). Dentro de la LOT se establece que los prestadores de servicios de telecomunicaciones tienen la obligación de brindar un servicio continuo, eficiente y de calidad, por lo tanto tienen prohibido interrumpir el servicio, salvo casos en los que se requiera realizar labores de mantenimiento o mejoras tecnológicas, para lo cual deberán notificar con anticipación la interrupción esto en el caso de las interrupciones programadas, o en el caso de interrupciones no programadas dentro del plazo establecido dentro de la LOT, cuando una interrupción se produce debido a eventos fortuitos de índole no imputable al prestador del servicio, para lo cual se deberá presentar pruebas. En ambos tipos de interrupción, el contrato de concesión establece un protocolo a seguir para realizar el proceso de aprobación o verificación de la interrupción según corresponda el caso, este protocolo se establece en un lapso de 15 días laborables [3], todo el proceso de verificación y seguimiento de la interrupción actualmente es realizado mediante correos electrónicos, lo cual supone un que los funcionarios de la ARCOTEL a cargo del control y regulación de las interrupciones del servicio móvil avanzado, debe establecer un sistema manual de registro para realizar un seguimiento sin contingencias negativas, lo cual considerando que dicho funcionario no solo realizan este rol dentro de su área, el proceso genera ineficiencia en el proceso laboral del funcionario debido a causas no adjudicables al mismo.

JUSTIFICACIÓN (IMPORTANCIA Y ALCANCES)

El proceso de seguimiento y control de una interrupción del servicio móvil avanzado es un proceso sistemático regido por el contrato de concesión entre las operadoras y la ARCOTEL, el proceso actualmente es realizado a través de correos electrónicos, y registros manuales por parte de los funcionarios de la ARCOTEL, lo que genera cargas innecesarias de trabajo que pueden ser evitadas mediante la digitalización total o parcial del proceso.

La digitalización de un proceso hace referencia a reducir la interacción humana directa con el proceso, permitiendo realizar cualquier etapa de manera ya sea automática o semiautomática con el mínimo restricciones a como se realizaba en el forma manual, por ello se propone un prototipo de plataforma web para el manejo de la interrupciones del servicio de telefonía móvil avanzada, la plataforma tiene por visión, ser un sistema intuitivo y simplista para seguimiento del proceso de interrupciones del servicio móvil avanzado, a través de herramientas de visualización, filtrado y comunicación que facilitan el seguimiento y gestión de la interrupción, de forma que reduzca las cargas laborales innecesarias tanto de los funcionarios de la ARCOTEL como operadores de telefonía móvil.

La plataforma se limita a ser un sistema de manejo del proceso y gestión de interrupciones del servicio de telefonía móvil avanzada, es decir permite disminuir las tareas manuales que un funcionario de la agencia debe realizar para el control y gestión de una interrupción, mediante un conjunto de herramientas web desarrolladas para cumplir esta función, además de brindar un nuevos servicios que buscan facilitar el seguimiento e identificación de las interrupciones registradas mediante los aplicativos propuestos, más no implementa ningún tipo de sistema de monitoreo o detección de interrupciones.

OBJETIVOS

OBJETIVO GENERAL

- Desarrollar una Plataforma para el Control y Gestión de interrupciones del Servicio de Telefonía Móvil Avanzada (SMA) para la ARCOTEL.

OBJETIVOS ESPECÍFICO

- Revisar normativa del servicio de Telefonía Móvil Avanzada (SMA) y herramientas de desarrollo para la implementación del proyecto.
- Implementar una plataforma para visualización de estado de BSs de las operadoras móviles del país.
- Diseñar y construir plataforma para gestión y control del proceso administrativo de interrupciones del servicio de Telefonía Móvil Avanzada (SMA).
- Realizar cuadro estadístico de interrupciones de servicio de Telefonía Móvil Avanzada, mediante pruebas realizadas con la plataforma desarrollada, en conjunto con la ARCOTEL.

CAPÍTULO 1: ESTADO DEL ARTE Y FUNDAMENTACIÓN TEÓRICA

En esta sección se realizará un análisis de las plataformas existentes que tengan características similares a la que se requieren para la plataforma propuesta para este proyecto, así también la fundamentación teórica que se requiere para el entendimiento del presente proyecto.

|TRABAJOS PREVIOS

El proceso de manejo de interrupciones del Servicio Móvil Avanzado, es un proceso que inicia cuando un operador por algún motivo que debe especificar, deja de brindar el servicio a los usuarios, el operador debe notificar a la ARCOTEL de la suspensión del servicio, cuando se realiza la notificación se establece un tiempo en el cual se realiza el seguimiento de la interrupción, hasta que la agencia compruebe las evidencias entregadas por las operadoras de la interrupción para ser dada por terminada. actualmente este proceso es realizado a través de correos electrónicos, entre los funcionarios de la ARCOTEL y los funcionarios de las distintas operadoras móviles. Esta metodología es poco práctica, debido a los múltiples problemas que puede ocasionar, para evitar estos problemas se debe mantener un estricto proceso organizacional, mediante registros y seguimiento manual de todo el proceso. Es importante mencionar que el personal que realiza el seguimiento y control del proceso tiene un sinnúmero de obligaciones dentro del área en el que labora y este tipo de procesos generan cuellos de botella para el flujo de trabajo, por ello es muy complicado de mantener un óptimo proceso de seguimiento de interrupciones del servicio de telefonía móvil avanzada mediante una metodología manual.

Actualmente, podemos encontrar algunos proyectos similares al planteado o que tiene una finalidad similar, entre los más notorios encontramos al “APLICATIVO PARA NOTIFICACIÓN DE INTERRUPCIONES DEL SERVICIO DE TELEFONÍA FIJA” [4], SEÑAL MOVIL ECUADOR [5] he “Infoantenas o VCTEL” [6], los cuales se describen a continuación.

1.1.1 APLICATIVO PARA NOTIFICACIÓN DE INTERRUPCIONES DEL SERVICIO DE TELEFONÍA FIJA

Es un aplicativo desarrollado en Java EE en conjunto con el framework JavaServer Faces (JSF) para la SUPERTEL actual ARCOTEL, el aplicativo permite realizar el registro y seguimiento de interrupciones en el servicio de telefonía fija, el proceso de registro y manejo del aplicativo se describe a continuación.

EL proceso para el uso del aplicativo empieza ingresando a la página http://aplicaciones.supertel.gob.ec/PRJ_SUPERTEL_ONLINE , luego se debe realizar un registro del usuario en línea, una vez se complete el proceso de registro se debe iniciar sesión, una vez que se verifiquen las credenciales del usuario, este tiene acceso a la página principal, donde se puede encontrar un menú que contiene el enlace “Interrupción” que permite realizar las acciones al usuario, las acciones dependerán de si el usuario es un operador tiene las opciones de registrar una interrupción o de listar las interrupciones y si es un agente de la ARCOTEL puede listar las interrupciones.

1.1.1.1 Listar interrupciones

La opción de listar interrupciones devuelve un listado de interrupciones que han sido registradas, en donde podemos encontrar los siguientes campos.

- Documento, corresponde al nombre de la operadora.
- Fecha de Registro, corresponde a la fecha en la que se registra la interrupción.
- Tipo de interrupción, corresponde al tipo de interrupción declarada por la operadora.
- Fecha y hora de inicio, corresponde a las fecha y hora, que sucede la interrupción en el caso de no programada o la fecha y hora en la que se planea vaya a ocurrir la interrupción.

- Fecha y hora de fin, corresponde a la fecha y hora de finalización de la interrupción,
- Abonados Afectados, corresponde al número de abonados que van a ser afectados por la interrupción,
- Estado, corresponde al estado actual de la interrupción.
- Acción, corresponde una acción para visualizar la interrupción seleccionada.
- Otros campos correspondientes a características de la interrupción registrada.

1.1.1.2 Notificación de interrupciones.

La notificación de la interrupción que se haya registrado será remitida mediante correo electrónico a la Administración Regional que le corresponda el análisis de la interrupción.

En el caso de visualizar una interrupción, el proceso consiste en dar clic en una de las interrupciones, esta acción nos devolverá la información registrada, la información mostrada se visualiza de igual manera como se la registra, únicamente con los campos bloqueados, los únicos campos habilitados para manipulación son, cambiar el estado de la interrupción, una entrada para ingresar una observación y una entrada para ingresar la duración total de la interrupción.

En el caso de ingreso de una interrupción, el operador debe llenar una serie de campos dispuestos al largo de la página, el ingreso es manual, dentro de los campos que se deben llenar encontramos, el tipo de interrupción, fecha de inicio y fin de la interrupción, el nivel de la interrupción, luego debe seleccionar los campos de ubicación de la interrupción buscando en todos los checkbox disponibles. es decir, se inicia seleccionando una provincia en al campo de provincias, luego aparece el campo de cantones el cual debe ser elegido del conjunto de resultados y finalmente parroquia con el mismo procedimiento, paso seguido se debe seleccionar los campos de Tipo de órgano, Grado de Afección, y Causas de la interrupción, entre otros campos que describen la interrupción y finalmente se envía la solicitud de registro.

El proceso consecuente, es hacer uso de los aplicativos de visualización expresados anteriormente para dar seguimiento de la interrupción.

1.1.2 SEÑAL MÓVIL ECUADOR

Es un aplicativo desarrollado para la ARCOTEL, mediante las librerías de Bootstrap y jQuery en agosto del 2014 y su última modificación fue realizada en febrero del 2015, el aplicativo tiene por objetivo el visualizar las mediciones de la señal del en el territorio ecuatoriano como se puede observar en la Figura 1-1 y está disponible en [5]

Entre características encontradas tras realizar un análisis del código fuente e interacción con el aplicativo, se determinaron.

- Permite realizar filtros entre tecnologías.
- Permite visualizar mediciones de señal y reportes de servicio.
- Consume el servicio de mapas de Google APIs actual Google Maps Platform.
- El consumo de datos es siempre dependiente del servidor, mediante imágenes PNG (el servidor genera los archivos .png en base a los requerimientos del aplicativo), que son servidas de acuerdo con el nivel de zoom en el aplicativo, es una de las razones identificadas por las que los tres mapas interactúan al mismo tiempo bajo alguna modificación del usuario.
- La metodología de imágenes .png para visualizar contenido dentro del mapa es eficiente cuando de cargar el aplicativo se trata, esta ventaja se da a costa de se generan limitaciones para dinamicidad del aplicativo, esto al ser una imagen que se superpone en un mapa mediante opacidad, genera que solo sea visible, es decir no se puede ejecutar acciones sobre los datos dentro del mapa.
- El sistema de tres mapas para visualizar información resulta útil para realizar comparaciones de dos o más operadoras, pero limita los dispositivos para una visualización “agradable al usuario”, a aquellos que dispositivos de pantalla relativamente grande, por ejemplo, una pantalla 768 x 1024, correspondiente a un iPad como se muestra en la Figura 1-2, ya que para el caso de dispositivos más pequeños la visibilidad es notoriamente deficiente como se muestra en la Figura 1-3. Figura 1-1

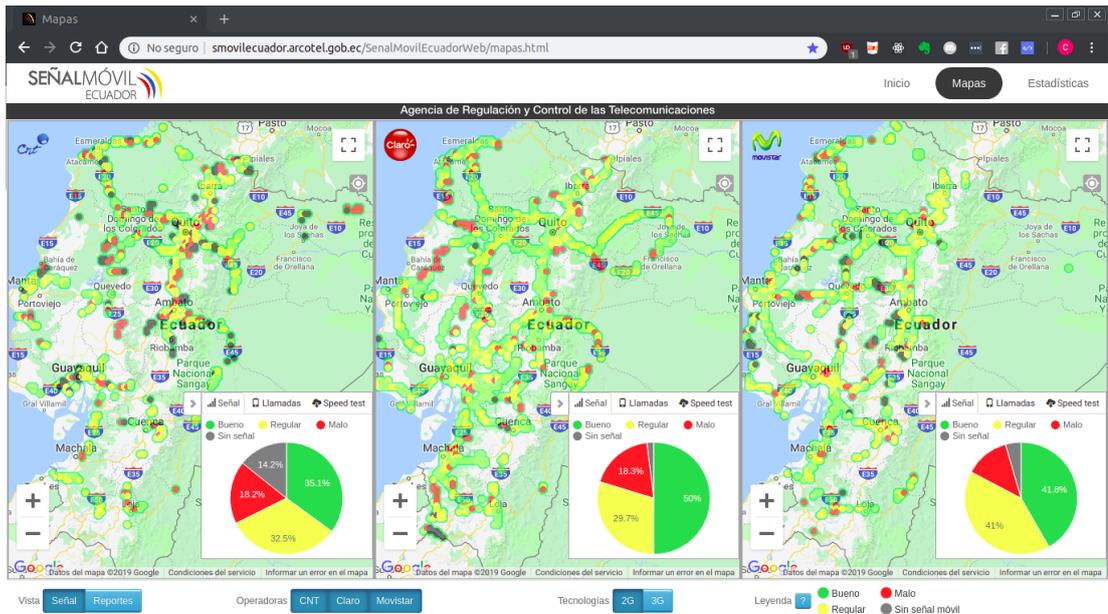


Figura 1-1. Plataforma Señal Móvil Ecuador.

Fuente: autor

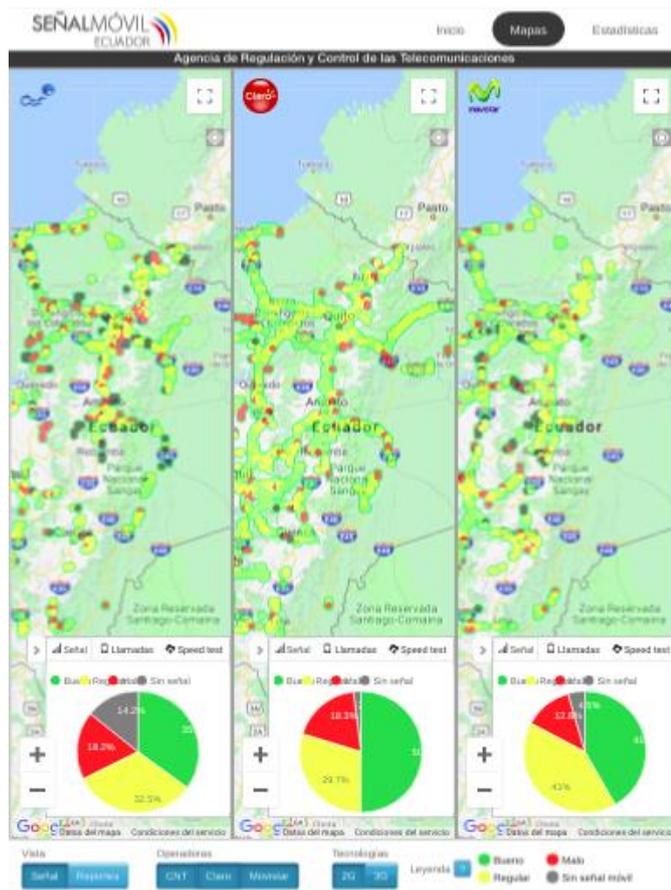


Figura 1-2. Plataforma Señal Móvil Ecuador en iPad.

Fuente: autor

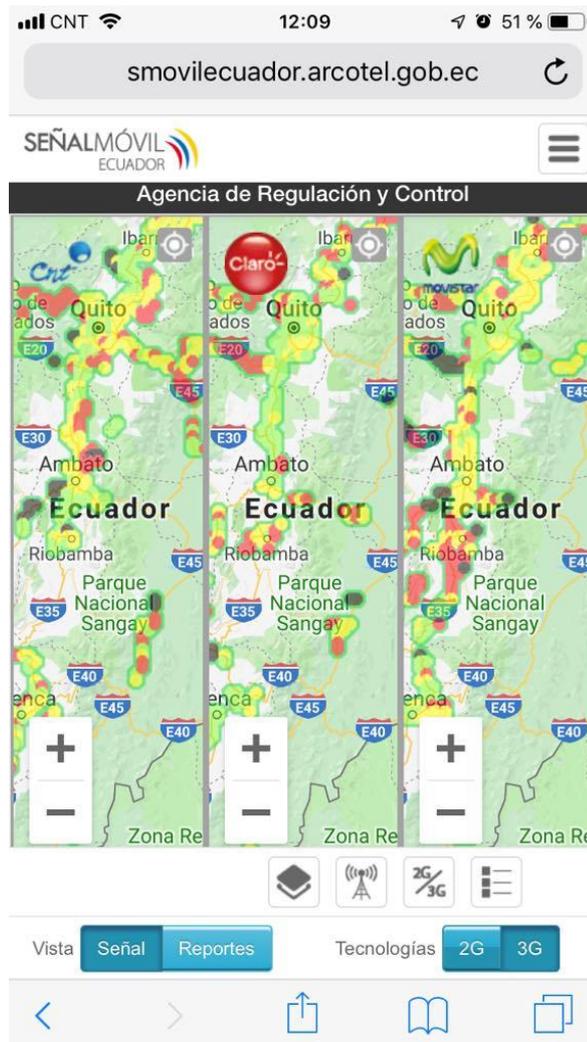


Figura 1-3. Plataforma Señal Móvil Ecuador Smartphone.

Fuente: autor

1.1.3 INFOANTENAS

Es un aplicativo desarrollado para la SECRETARIA DE ESTADO PARA LA SOCIEDAD DE LA INFORMACION Y LA AGENDA DIGITAL de España mediante la librería de jQuery, el aplicativo tiene como objetivo la visualización de estaciones de telefonía móvil que cumplen con los niveles de radiación no ionizante en el territorio español como se muestra en la Figura 1-4 y está disponible en [6]

Entre características encontradas tras realizar un análisis del código fuente e interacción con el aplicativo, se determinaron.

- Permite visualizar información de estaciones móviles en el territorio español.

- Consume un servicio de mapas de licencia abierta llamado OpenLayers, es un api de bajo nivel para manejo de mapas.
- Permite realizar búsquedas de localidades mediante selectores como provincia, municipio y entradas como calle y número de calle, la metodología de filtrado se basa en un sistema de coincidencias de texto, por ello es requerido conocer el nombre de las calles como mínimo requerimiento para tener un conjunto de resultados satisfactorios, caso contrario el mapa se situará a nivel del municipio el cual es un zoom bastante alejado, considerando el siguiente apartado.
- Para poder visualizar las radiobases se requiere realizar un zoom hasta al menos los cuatro últimos niveles como se muestra en la Figura 1-5, esta metodología provoca molestas para identificar la ubicación de las radiobases.
- El consumo de datos es siempre bajo petición al servidor, los datos usados son de tipo JSON que contiene la información de la radiobases, esta metodología permite presentar los datos en el mapa de manera dinámica, es decir permite realizar acciones en los puntos del mapa, ganando interactividad, pero dependiendo de la cantidad de datos el consumo en red varia.

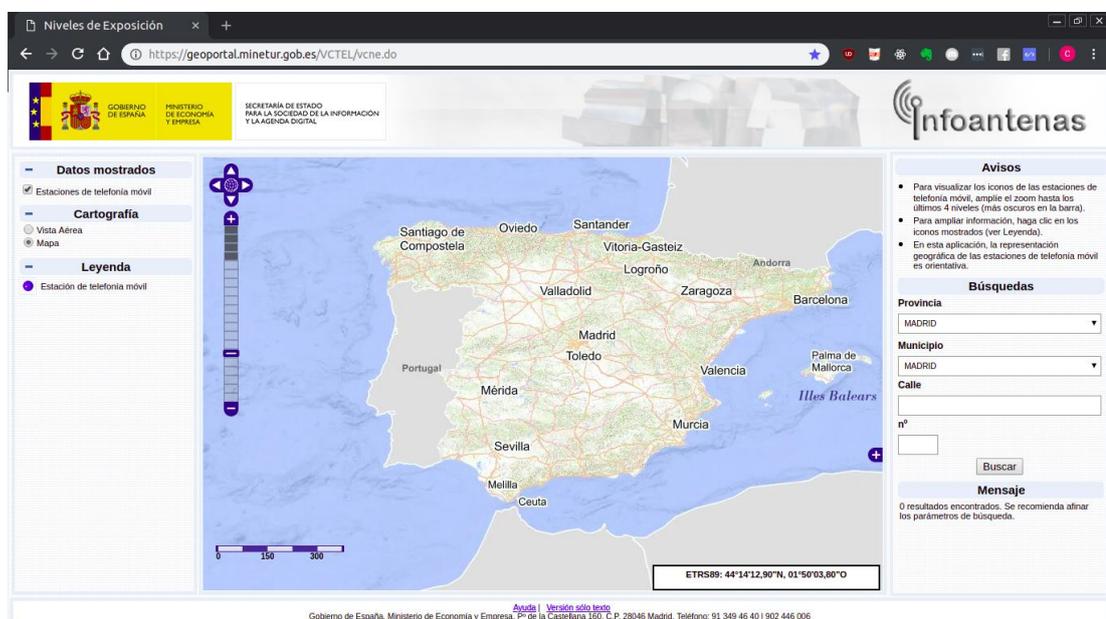


Figura 1-4. Plataforma Infoantenas.

Fuente: autor

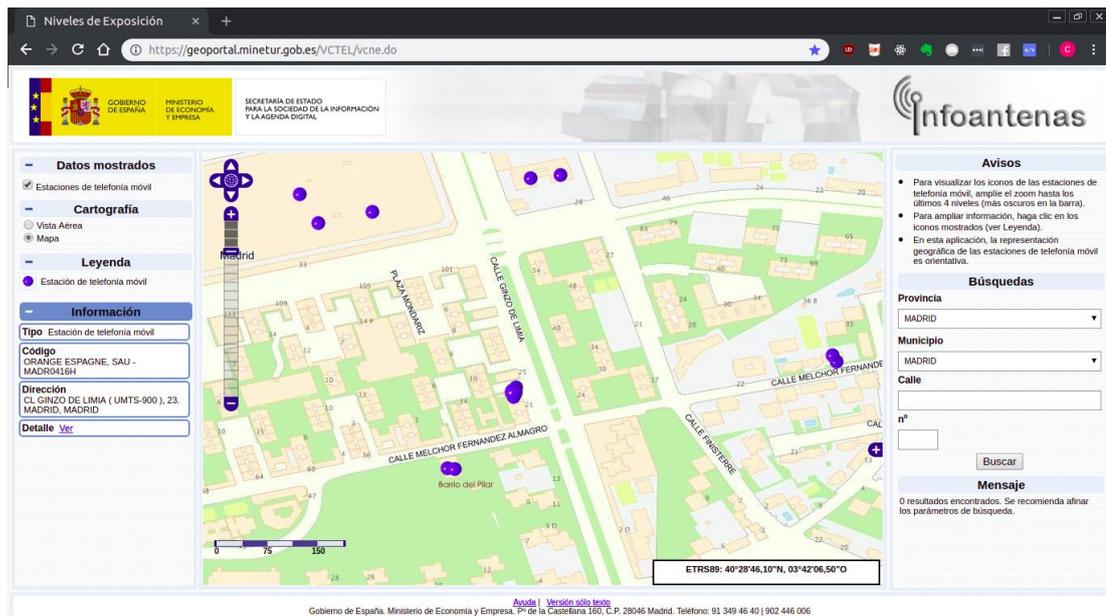


Figura 1-5. Radiobases mostradas en plataforma Infoantenas.

Fuente: autor

ANÁLISIS DE DATOS

En múltiples áreas de investigación, proyectos, medicina, entre otros, en los cuales se trabaja con grandes cantidades de información es necesario buscar el método adecuado para su análisis y obtener así nueva información que se encuentra implícita que pueda aportar al proyecto o investigación que se desarrolla.

El análisis estadístico permite identificar tendencias, dependencias y/o patrones mediante la recolección de información asociada al problema de estudio, mediante distintos métodos como mediciones, observaciones, encuestas, etc., debido a la naturaleza y el problema de estudio hace que la información contenga múltiples tipos de variables para lo cual se requiere acondicionar las variables para que sean útiles en el análisis del problema, mediante una estructuración adecuada. La estructuración de datos depende del analista, por lo cual el paso final del proceso es una validación del análisis realizado.

1.2.1 VARIABLES

Las variables permiten medir un cambio en un fenómeno, una unidad de medición o análisis, dependiendo de la unidad toma distintos tipos. [7]

Existen varios tipos de variables según la unidad de análisis.

1.2.1.1 Categóricas

Indican distintos grupos o categorías, y se dividen en tres tipos.

Nominal: Indica dos o más grupos sin que exista una relación de escala entre estos, (e.g., Sierra, Amazonia, Costa, Región insular, como posibles opciones de una variable Región).

Dicotómica: Cuando existe únicamente dos posibles valores, (e.g., Si, No).

Ordinal: Este tipo de variable presenta un orden jerárquico o una escala la cual tiene valores que dan una magnitud de su valor o importancia, (Muy en desacuerdo, desacuerdo, de acuerdo, muy de acuerdo). [7]

1.2.1.2 Continuas

Cada dato pertenece a un valor dentro de una escala de medición y no de pertenencia a un grupo o categoría, por ejemplo, la temperatura que puede tener valores según la medición y escala de esta. [7]

1.2.2 CODIFICACIÓN DE VARIABLES

Suele ser necesaria una codificación para la simplificación de los datos cuando se trata de variables categóricas que contienen posibles valores que pueden llegar a ser poco eficientes de manejar, esto mediante la asignación de los posibles valores por números. [7]

Para el caso de variables nominales o dicotómicas la codificación consiste en asignar un valor numérico a cada valor posible de la variable, el orden en este caso no tiene ningún significado o valor, por el contrario, en el caso de las variables ordinales la asignación será dada según la escala definida por la variable para cada caso. [7]

Tabla 1-1. Codificación de variable causa.

Causa de interrupción de servicio de comunicaciones móviles	Valor
Intermitencia	0
Caída de enlace microonda	1
Caída de torre	2
Falla del sistema eléctrico	3
Corte de energía eléctrica	4
Corte de fibra óptica	5
Falla de equipos de comunicaciones	6
Falla del sistema de enfriamiento	7
Mantenimiento programado	8
Cambio de Hardware	9
Implementación de nuevo enlace	10

1.2.3 ESTADÍSTICAS DESCRIPTIVAS

El punto de partida para un análisis estadístico más profundo es el conjunto de información o datos acumulados, esto da una visión general de la información y así plantear los posibles análisis más profundos como métodos de regresión asociación, dependencias y demás. [7]

El tipo de método que se debe seleccionar dependerá del tipo de variables que se maneje, para el caso de variables categóricas se tiene tablas de frecuencia y tabulaciones cruzadas. [7]

Las tablas de frecuencia consisten en dar un conteo para cada caso, valor o estado posible de la variable o conjunto de variables dentro de la información, mientras que las tabulaciones cruzadas consisten en tablas que están compuestas por filas y columnas de cada variable con sus posibles valores así con su respectivo conteo, porcentaje en función del total y total [7].

Para datos continuos se utilizan tablas de frecuencias, histogramas, medias, varianzas entre otras medidas de tendencia. [7]

1.2.4 PRUEBA CHI-CUADRADO (χ^2) DE INDEPENDENCIA

Esta prueba es de tipo no paramétrica, que sirve para evaluar cuan relacionadas están una o varias variables categóricas, mediante el cálculo del factor chi-cuadrado,

estas variables son de tipo nominal es decir se asigna un grupo de categorías como posibles valores. [8]

La hipótesis que es evaluada por este método es la de igualdad de frecuencias, es decir la frecuencia observada es igual a la frecuencia esperada. [7]

$$H_0: f_0 = f_e$$

1.2.4.1 Distribución chi-cuadrado

Esta distribución es tipo no paramétrica lo que indica que no es igual a una distribución normal en todos los casos, esta distribución indica los posibles valores que puede obtenerse al calcular el factor chi-cuadrado. [7]

Presenta únicamente valores positivos dado a que su cálculo es en función de números elevados al cuadrado, además, depende también de los grados de libertad. [7]

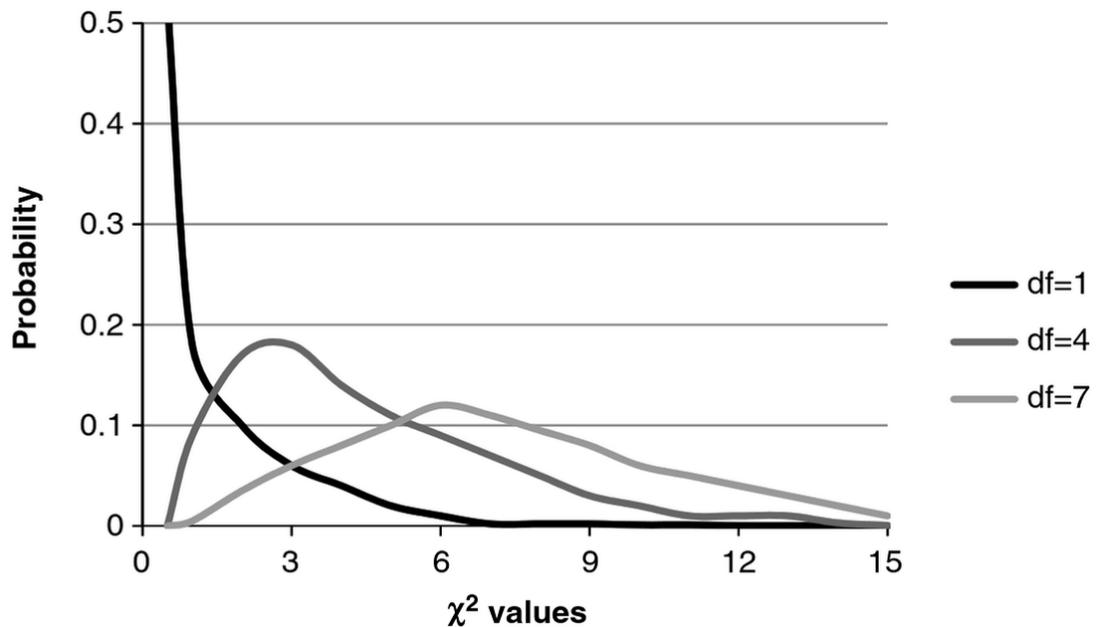


Figura 1-6. Distribución chi-cuadrado.

Fuente: [7]

1.2.4.2 Valor crítico

Este valor indica el límite en el cual existe una diferencia significativa en el valor esperado y el observado, se obtiene a partir de los grados de libertad y el valor de probabilidad dentro de la curva de distribución, el valor de probabilidad se basa en el estándar de $\alpha = 0.05$ que indica un 5% de los valores de factor chi-cuadrado. [7]

Para el cálculo de los grados de libertad se tiene:

$$df = (R - 1)(C - 1)$$

Donde R y C representan las filas y columnas de las variables a analizar. [7]

1.2.4.3 Cálculo de factor χ^2

Para el caso más simple cuando se analiza una sola variable se tiene:

$$\chi_{calc}^2 = \sum \frac{(f_0 - f_e)^2}{f_e}$$

Cuando son varias variables se calcula mediante la misma fórmula y el valor esperado se obtiene con la siguiente formula:

$$f_e = \frac{\text{filas} * \text{columnas}}{\text{total muestras}}$$

1.2.4.4 Desarrollo de análisis

Para el desarrollo del método primero debe tomarse en consideración los tipos de las variables a analizar, deben ser categóricas nominales, debe existir independencia entre las muestras, también la cantidad mínima de muestras debe ser de 5 por cada caso posible o un mínimo del 75% al 80% del total debe tener mínimo 5 muestras. [8]

Luego se definen las hipótesis tanto la hipótesis nula como la alternativa (H_0, H_1).

El siguiente paso es definir el valor de probabilidad α , que en conjunto con el valor de los grados de libertad permitirán determinar el valor crítico para el posterior análisis. [8]

Se procede a calcular el valor de chi-cuadrado mediante la fórmula para χ_{calc}^2 , así como el valor de la frecuencia esperada según las características de la información y sus variables.

Finalmente, según los resultados y el valor crítico se procede a descartar alguna de las hipótesis planteadas. [8]

1.3 HERRAMIENTAS DE DESARROLLO

1.3.1 PLATAFORMA WEB

Un plataforma Web es un conjunto de páginas Web que tienen la finalidad de cumplir requerimientos visuales como de funcionalidad directa o indirectamente impuestos por el usuario final, el objetivo general de una plataforma Web es el de albergar los servicios necesarios para realizar determinados procesos, disminuyendo en la medida de lo posible la necesidad de interacción humana directa con los procedimientos pero permitiéndole desenvolverse de manera flexible a través de ellos, en base a lo mencionado anteriormente, se puede notar la diferencia entre los diferentes tipos de servicios Web, en donde podemos encontrar a las páginas Web que son parte de una plataforma pero no pueden ser consideradas como tal debido a su simplicidad, falta de funcionalidad y limitación para cumplir con todos los requisitos impuestos por el usuario, normalmente estas páginas son conocidas como Landing Pages, por otro lado podemos encontrar otro tipo de servicios como los E-commerce, sitios Web, Web Apps, entre otros, cada uno de estos servicios tienen un objetivo a cumplir, lo que hace que difieran en determinados aspectos unos de otros, pese a que su construcción haga uso de las mismas tecnologías, metodologías, técnicas de diseño y programación, la forma en la cual se estructuran los diferentes componentes y funcionalidades harán marcar dicha diferencia. [9]

1.3.2 HISTORIA DE LOS SERVICIOS WEB

Se podría alegar que la historia de los servicios Web se puede vincular al nacimiento del Internet debido a su dependencia de este último, pero esto es un error común, esto debido que estos servicios/tecnologías son diferentes y cumplen papeles diferentes, por ejemplo, internet es el medio por el cual se brinda en servicio Web. Se puede sostener que los servicios Web vieron su luz en entre los años de 1989 y 1990 en el CERN (por sus siglas en francés Organización Europea para la investigación Nuclear) de la mano Tim Berners-Lee y Robert Cailliau mediante el desarrollo del World Wide Web (WWW). Debido a la estrecha relación de la Web, en la década de los ochenta internet estaba destinado únicamente para fines investigativos por ello apenas en ese momento existían unas cuantas decenas de páginas, pero este era el punto de partida, para enero del 2019 se puede encontrar cerca de 5.12 billones de páginas web indexadas según [10], muchos de estos sitios brindando servicios de manejo y

gestión de procesos de manera digital, lo cual marca una nueva visión para la web moderna. [11]

1.3.3 APLICACIONES EN INTERNET Y PROGRAMACIÓN EN LA RED

En la actualidad existe un sinnúmero de servicios web brindando mediante internet, entre los que podemos encontrar a las plataformas web, correo electrónico, e-commerce entre otros, pero a pesar de que estos servicios difieren tanto en estructura como en finalidad, ninguno es dependiente o es parte de una infraestructura de comunicación específica. Debido a que internet es un mecanismo de comunicación de propósito general capaz de soportar la construcción y distribución estos servicios es el medio predilecto para realizar esta tarea.

1.3.3.1 Modelo Cliente-Servidor

Como se mencionó anteriormente, internet el mecanismo de condiciones de propósito general que es capaz de conectar dos computadores, pero no es capaz de establecer e iniciar la comunicación entre los terminales, para ello los programas de aplicación como puede ser el modelo Cliente-Servidor, permiten iniciar la comunicación entre un computador y uno remoto iniciando la comunicación. Es decir, el modelo Cliente-Servidor ejecuta programas de aplicación en dos computadores independientes de tal manera de que estos coordinen para establecer una comunicación, garantizando una solicitud de conexión simultánea en donde el servidor el primero en iniciar seguido por el cliente. [11]

1.3.3.2 Características de Clientes y Servidores

El flujo de datos entre un cliente y servidor se da de manera bidireccional, aunque es común que el cliente envíe solicitudes y el servidor devuelve respuestas, de esta manera el software del cliente y del servidor pueden tener algunas diferencias las cuales se mencionan a continuación.

El Software del Cliente

Es un programa de aplicación que se encuentra alojado en el servidor, pero temporalmente se envía al cliente y es transformado como tal, requiriendo acceso remoto hasta el servidor de origen, por consiguiente, este tipo de software únicamente

se ejecuta bajo petición del cliente y se ejecuta mientras dura la sesión de un cliente, es capaz de hacer uso de múltiples servicios prestados por el servidor como sea necesario una vez este se haya conectado con el servidor. [11]

El Software del servidor

Es un programa de propósito general que tiene como objetivo proporcionar un servicio a múltiples clientes remotos de manera simultánea, se ejecuta en un computador de capacidades de hardware superiores y dedicadas con respecto a los computadores de uso diario, este software se ejecuta cuando se inicia el sistema o a su vez cuando el administrador levante el servicio, manteniéndose en espera de conexiones de índole arbitraria por parte de un cliente, el programa se está ejecutando continuamente durante sesiones indefinidas. [11]

Estructuras de Software Cliente-servidor

El manejo de software tanto del cliente como del servidor depende de qué tipo de estructura de se maneje en el servicio Web, de manera general para cuestión del proyecto se especifican dos tipos el renderizado al lado del Cliente (CSR) y el renderizado al lado del servidor (SSR), la diferencia notoria entre estos dos está en que el CSR descentraliza el software del Cliente y el del servidor es decir ambos programas están separados tanto por software como opcionalmente de hardware y pueden servirse mediante dos computadores independientes, además que permite una escalabilidad independiente y la comunicación de puede realizar mediante arquitecturas como por ejemplo REST (Transferencia de Estado Representacional), en el caso del SSR es una estructura más centralizada dado como el software tanto del cliente como del servidor se ejecutan de manera dependiente normalmente ejecutan bajo el modelo vista controlador (MVC) o similares, lo cual inhibe que la escalabilidad de ambos es dependiente. Ninguno de los dos modelos es perfecto cada uno tienen sus pros y contras y esto es necesario analizar al implementar un servicio, en [12] se puede encontrar una lectura productiva para tener mayor entendimiento de este apartado. [11]

1.3.4 MODELOS DE PLATAFORMA WEB

1.3.4.1 Modelo del Cliente

JavaScript es un lenguaje de programación orientado a objetos, imperativo, funcional, manejado por eventos que fue creado por Brendan Eich un desarrollador de Netscape, en 1995 bajo el nombre de Mocha fue creado JavaScript, para luego en Diciembre de 1995 adoptar el nombre de JavaScript, la razón por la que JavaScript se mantiene como el lenguaje para Web se deba a su soporte por los navegadores, este hito se dio cuando los navegadores empezaron a implementar y soportar el árbol DOM (Document Object Model) el cual permitía una experiencia de usuario más completa. [13]

Document Object Model (DOM)

El DOM es una jerarquía/árbol de objetos de nodo JavaScript.

Cuando se escribe un documento HTML, se encapsula el contenido HTML dentro de otro contenido HTML a manera de cascada, al hacer esto se configura una jerarquía que puede expresarse como un árbol. El navegador, al cargar el documento HTML, interrumpe y analiza la jerarquía para crear un árbol de objetos de nodo que simula como se encapsula el esquema HTML.

El propósito del DOM es proporcionar una interfaz programática para la creación de Scripts JavaScript en el documento, El DOM originalmente fue una interfaz de programación de aplicaciones para documentos XML y se extendió para el uso de documentos HTML. [14] [15]

React.js

Es una librería de JavaScript desarrollada por Facebook Inc. para construir interfaces de usuario. [16]

React está diseñada para actualizar el DOM del navegador por nosotros, esto ya que con React el desarrollador no interactúa directamente con el API DOM, en su lugar se interactúa con un DOM virtual, o lo que es a su vez un conjunto de instrucciones que REACT utilizara para construir la interfaz de usuario e interactuar con el navegador.

DOM Virtual es un conjunto de instrucciones que un navegador sigue al construir un DOM, los elementos que forman un documento HTML se convierten en elementos DOM cuando el navegador carga HTML y presenta la interfaz de usuario.

La API DOM es un conjunto de objetos que JavaScript puede usar para interactuar con el navegador y así realizar modificaciones en el DOM, el proceso de manipulación y modificación del DOM es bastante sencillo, pero si el objetivo es agregar nuevos elementos, este el proceso se vuelve considerablemente lento debido al uso ineficiente del DOM. La gestión eficiente del DOM con JavaScript es una es una tarea laboriosa y muy complicada por lo que se requiere un conocimiento en JavaScript muy avanzado, de ello nace React.js como una solución a esta problemática.

React es una librería relativamente pequeña como para ser considerada como “Framework” de JavaScript que está diseñada para actualizar el DOM del navegador por nosotros evitando toda la complejidad mencionada anteriormente, de este modo al ya no interactuar con el API DOM directamente, en cambio se interactúa con un DOM Virtual, o conjunto de instrucciones que React utilizara para construir la interfaz de usuario e interactuar con el navegador Web.

El DOM Virtual está conformado por elementos React, que de manera conceptual son semejantes a elementos HTML, pero que en realidad son objetos JavaScript, esto dado que es mucha más rápido y relativamente sencillo trabajar con dicho objetos JavaScript que con la API DOM, por ello React y su DOM Virtual representan de manera más eficaz y eficiente la manipulación del API DOM.

La eficiencia de React y su DOM Virtual se basan en dos ideas simples como son el ejecutar un algoritmo de “diffing” que se encarga de identificar los cambios que se desean realizar en el DOM, luego se realiza la “reconciliación” donde se actualiza el cómo los resultados de diff, lo que permite una optimización de procesos de manipulación de DOM.

La complejidad de la operación “diff” mencionada anteriormente puede referirse a uno operación que devuelve los elementos que han cambiado en un documento se estima en orden $O(n^3)$, lo cual es inaudito para una aplicación de cualquier índole, pero para solucionar este inconveniente los ingenieros de Facebook mediante React.js haciendo uso de un mapa de hash lograron que el coste

computacional se redujera a $O(n)$ el cual es el logro más importante para que React.js sea una de las librerías más populares hoy en día. [17]

1.3.4.2 Modelo del Servidor

RESTful Web Services

Representational State Transfer (REST) es una arquitectura de software para sistemas hipermedia distribuidos permitiendo diseñar aplicaciones de red, además se define como cualquier interfaz entre sistemas HTTP que permite obtener datos en formatos JSON y XML. [18]

Dentro de los conceptos más importantes de REST encontramos, “Interfaz Uniforme”, esta restricción define que la interfaz entre cualquier servidor y todas las posibilidades de los clientes deben ser las mismas, permitiendo desacoplar la arquitectura entre el cliente y el servidor, permitiendo una evolución independiente y generando escalabilidad de acuerdo a las necesidades propias ya sea de cliente o servidor, “Stateless”, significa que cada solicitud debe llevar el estado necesario para mantener la transacción, permitiendo liberar al servidor de mantener un estado de información entre múltiples solicitudes. “Sistema de capas”, indica que, desde el lado del servidor, la escalabilidad puede mejorarse de tal manera que el cliente no se dé cuenta de si se trata de un servidor o una aplicación intermedia. “Código bajo demanda”, Significa que el servidor también puede entregar la lógica al cliente, como el contenido de JavaScript que puede ser interpretado por el cliente, esto aplica cuando se realiza un renderizado al lado del servidor (SSR) pero es totalmente opcional debido a la aparición de las SPA (Single Page Apps) las cuales realizan un renderizado al lado del cliente (CSR). [19] [20]

Implementación REST.

- Usar verbos HTTP significativos.
 - GET, POST, PUT, DELETE, el propósito de estos verbos en los principios de REST se usan como una ventaja para establecer la claridad de la API RESTful.
- Nombre de recursos sensibles
 - El URI (Identificador de Recursos Uniforme) debe ser tan significativo como el verbo HTTP. <https://midominio.com /user/123>.

- Uso de XML y JSON, dependiendo de las necesidades del cliente.
- Crear recursos Fine-Grained
 - En lugar de crear servicios complejos, es recomendable generar recursos simples, y proporcionar operaciones CRUD en ellos.
- Idempotente para PUT y DELETE
 - Esto significa que al enviar una solicitud con el mismo estado (datos) al servidor, tendrá el mismo efecto en el dominio / base de datos.
- Seguridad para HEAD, GET, OPTIONS y TRACE
 - Estos métodos se definen para ser seguros, esto significa, para no alterar el estado en el servidor.

[19]

1.3.5 TECNOLOGÍAS Y LIBRERIAS USADAS EN EL DESARROLLO DE LA PLATAFORMA

1.1.2.1 Front-End

Son aquellas tecnologías que se ejecutan al lado del cliente, es decir en el navegador, las tecnologías usadas se describen a continuación.

HTML5: Lenguaje de marcado, se encarga de describir el contenido en el Navegador Web.

CSS3: Es un lenguaje utilizado para describir la presentación de un documento HTML o XML, describe cómo se va a renderizar el elemento estructurado por HTML en la pantalla.

JavaScript: Es un lenguaje de programación multiparadigma que permite realizar actividades complejas en el navegador Web.

ReactJs: Es una librería de JavaScript open-source para construir interfaces de usuario, mediante una óptima manipulación del DOM.

Redux: Es un contenedor del estado de aplicaciones JavaScript

React Router: Librería de JavaScript open-source desarrollada para la librería de ReactJs para crear enrutamiento tanto dinámico como estático ya sea en el lado del cliente como en el lado del servidor beneficiando al redireccionamiento en las SPA (Single Page Apps).

Leaflet: Es la librería de JavaScript open-source para mapas interactivos, su diseño se basa en los aspectos de simplicidad, rendimiento y alta disponibilidad de personalización y desarrollo en base a su núcleo.

D3.js: Es una librería de JavaScript open-source para manipular documentos basados en datos. D3 permite vincular datos al DOM y luego aplica transformaciones basadas en manipulación del DOM, al igual que ReactJS permite la manipulación eficiente del DOM, pero en este caso este se especializa en el manejo de datos, lo cual evita la representación propietaria y ofrece una flexibilidad al momento de desarrollar permitiendo exponer todas las capacidades de los estándares Web. Debido a que esta librería es de bajo nivel contrario a librerías como chart.js, canvas.js, RAWGraphs, entre otros, permite la manipulación de grandes cantidades de datos con una velocidad extremadamente rápida, además de permitir un comportamiento dinámico para la interacción y animación.

IndexedDB: Es un sistema de bases de datos para el almacenamiento al lado del cliente. Pese a que se puede usar el DOM Storage para almacenamiento local, pero este está diseñado y optimizado para almacenamiento de pequeñas cantidades de datos, debido a este problema aparece IndexedDB el cual es un tipo de base de datos de tipo transaccional (key-value) al lado del cliente que permite almacenar y manipular grandes cantidades de datos estructurados, con la finalidad de disponer de datos persistentes beneficiando de esta manera al desarrollo de aplicaciones Web que requieren funcionamiento en modo sin conexión.

Socket.io-client: Es una librería de JavaScript open-source que permite una comunicación en tiempo real, bidireccional basada en eventos, esta se ejecuta al lado del cliente complementándose con el lado del servidor.

1.2.2.1 Back-End

Son aquellas tecnologías que se ejecutan al lado del servidor, comúnmente se encargan de la lógica de un sitio Web en las páginas tradicionales, aunque en la actualidad con las SPA, ha empezado a compartir esta funcionalidad con el lado del cliente, además se encarga de realizar la manipulación de los datos, las tecnologías usadas se describen a continuación.

Node.js: Es un entorno multiplataforma basado en el motor V8 de Google, open-source usado para la capa del servidor basado en el lenguaje ECMAScript, es de naturaleza asíncrona, con un sistema I/O de datos en una arquitectura orientada a objetos, actualmente desarrollada por Node.js-Foundation y desarrolladores de todo el mundo.

Express.js: Es una librería open-source desarrollada como infraestructura web rápida, minimalista y flexible para Node.js, proporcionando una capa de características de aplicaciones web, lo cual permite una flexibilidad de desarrollo para servidores bajo la estructura de Node.js.

PostgreSQL: PostgreSQL es un motor de base de datos relacionales de propósito general más avanzadas del mundo y la más avanzada en open-source, en sus inicios fue desarrollado para ejecutarse en plataformas basadas en UNIX, pero en su camino fue diseñado para ser multiplataforma, uno de los aspectos más importantes de postgres es que requiere mínimos esfuerzos de mantenimiento debido a su alta estabilidad ya que desarrolladores de todo el mundo contribuyen en esta tarea, además si se desarrolla aplicaciones basadas en PostgreSQL, el coste total de la propiedad es bajo en relación a otros motores de bases de datos, debido a su licencia open-source, además de requerir mínimas extensiones para funcionamiento especializado. [21]

Redis: Es un motor de base de datos basado en memoria open-source, es utilizado como base de datos, caché y agente de mensajes, su característica sobresaliente se debe a su compatibilidad con strings, hashes, listas, conjuntos, conjuntos ordenados con rango de consultas, mapas de bits, hyperloglogs, índices geoespaciales con consultas “radius” y streams.

Puppeteer: Es una librería de Node.js que proporciona una API de alto nivel que permite el control de headless Chrome/Chromium bajo el protocolo DevTools, permite realizar procesos propios del navegador (realizados manualmente), de manera automática mediante código en el servidor.

Handlebars.js: Es una librería de Node.js open-source que permite compilar y renderizar plantillas semánticas de documentos HTML en el servidor.

Knex.js: Es un ORM (Object Relational Mapping, permite convertir los datos de objetos en un formato válido para poder guardar la información en una base de

datos, creándose una base de datos virtual donde los datos que se encuentran en la aplicación se vinculan a la base de datos), open-source para Node.js, en su estructura tiene la capacidad de funcionar con estructuras tradicionales de Node.js, como son los Callbacks y las Promesas de manera indiferente, garantizando un flujo asíncrono y una interfaz stream, consultas de funciones complejas y constructores de esquemas, soporte de transacciones, agrupación de conexiones y respuestas estandarizadas entre diferentes clientes de consultas y dialectos.

JWT: JWT (JSON Web Token) es un estándar abierto (RFC 7519) propuesto por el IETF para representar “claims” (identidad y privilegios) de manera segura entre dos partes, los JWS están diseñados para ser compactos, para poder ser enviados en las URLs, y ser usados en escenarios SSO (Single Sign-On), en el caso de Node.js podemos encontrar jwt.io el cual es una librería muy flexible para el manejo del estándar en las aplicaciones Web.

Socket.io: Es una librería de JavaScript open-source que permite una comunicación en tiempo real, bidireccional basada en eventos, que se ejecuta en el lado del servidor, para ello se basa en WebSocket pero son dos tecnologías distintas ya que este no es una implementación del protocolo WebSocket sino que es una librería de comunicación web en tiempo real que utiliza varios protocolos para conformarse, por ese motivo no soporta interacciones con otros clientes de WebSocket.

CAPÍTULO 2: MARCO METODOLÓGICO

A continuación, se describe la metodología para el desarrollo de los objetivos planteados para este proyecto, en primera instancia fue necesaria la revisión de la normativa vigente para las interrupciones del Servicio Móvil Avanzado, que servirán de base para establecer las características del aplicativo.

REGULACIÓN DEL SERVICIO MÓVIL AVANZADO EN EL ECUADOR

El ente encargado de la administración, gestión, control y regulación de las telecomunicaciones del espectro radioeléctrico en el Ecuador es la Agencia de Control y Regulación de las Telecomunicaciones (ARCOTEL). De acuerdo a la Ley Orgánica de Telecomunicaciones (LOT) en el Título XIV, Capítulo II, Artículo 142, Creación y naturaleza, crea la agencia de Regulación y Control de Telecomunicaciones (ARCOTEL) “como persona jurídica de derecho público, con autonomía administrativa, técnica, económica, financiera y patrimonio propio, adscrita al Ministerio rector de las Telecomunicaciones y la Sociedad de la Información. (...) es la entidad encargada de la administración, regulación y control de las telecomunicaciones y del espectro radioeléctrico y su gestión, así como de los aspectos técnicos de la gestión de medios de comunicación social que usen frecuencias del espectro radioeléctrico o que instalen y operen redes.” [3], de este manera la ARCOTEL ejercer dicho papel como ente regulatorio y de control basado en la Constitución de la República del Ecuador en los artículos Art. 261 numeral 10, Art. 313, Art. 314. En el cual el estado se reserva los competencias exclusivas y los derechos de administrar, regular, controlar y gestionar los sectores estratégicos dentro del cual se encuentra englobado las Telecomunicaciones y sus derivados, en el Art. 315 la constitución ratifica la disposición del Estado para construir “empresas públicas

para la gestión de sectores estratégicos, la prestación de servicios públicos, el aprovechamiento sustentable de recursos naturales o de bienes públicos (...), las empresas públicas estarán bajo la regulación y control específico de los organismo pertinentes, de acuerdo a la ley, (...)” [22].

2.1.1 COMPETENCIAS DE LA ARCOTEL

La ARCOTEL según lo expuesto en el Título XIV, Capítulo I, Artículo 141 Competencias del Órgano Rector de la LOT, a la ARCOTEL se le adjudica 30 competencias como órgano regulatorio y de control de las Telecomunicaciones y de la Sociedad de la Comunicación [3]. El conjunto de competencias que la agencia posee está estructurado de tal manera que se garantiza el cumplimiento de los normas y objetivos institucionales para los cuales fue creado, de ellos a continuación se especifican los concernientes al manejo de interrupciones del Servicio Móvil Avanzado.

LOT, Título XIV, Capítulo II, Artículo 144 Competencias de la Agencia.

Numeral 1, “Emitir las regulaciones, normas técnicas, planes técnicos y demás actos que sean necesarios en el ejercicio de sus competencias, para que la provisión de los servicios de telecomunicaciones cumpla con lo dispuestos en la Constitución de la República y los objetivos y principios previstos en esta ley, (...)” [3].

Numeral 4, “Ejercer el control de la prestación de los servicios de telecomunicaciones, (...), con el propósito de que estas actividades y servicios se sujeten al ordenamiento jurídico y a lo establecido en los correspondientes títulos habilitantes.” [3].

Numeral 7, “Normar, sustanciar y resolver los procedimientos de otorgamiento, administración y extinción de los títulos habilitantes previstos por la ley” [3].

Numeral 18, “Iniciar y sustanciar los procedimientos administrativos de determinación de infracciones e imponer en sus casos, las sanciones previstas por la ley.” [3].

Numeral 23. “Requerir a las y los prestadores de servicios de telecomunicaciones cualquier información que considere conveniente, producida

como consecuencia de la prestación de los servicios y ejecución de los títulos habilitantes dentro del ámbito de sus competencias.” [3].

2.1.2 DISPOSICIÓN DE LA ARCOTEL

La ARCOTEL de acuerdo con la LOT, Título XIV, Capítulo II, Artículo 143 Domicilio y desconcentración, establece que la ARCOTEL tiene sede en el Distrito Metropolitano de Quito, pero presenta una disposición desconcentrada con el fin de una eficaz desconcentración administrativa, todo de acuerdo a el “ordenamiento jurídico vigente”, pese a desconcentración administrativa la Ley Orgánica de Telecomunicaciones no puede desconcentrarse por lo que todas las delegaciones deben regirse en esta.

De esta manera la distribución zonal de la ARCOTEL se basa en las estructuras zonales desconcentradas dada por la Secretaría Nacional de Planificación y Desarrollo (SENPLADES), dicha distribución constituye en agendas zonales dispuestas como se muestran la Figura 2-1.



Figura 2-1 Agendas Zonales dispuestas por la SENPLADES.

Fuente: [23]

De acuerdo con la distribución presentada anteriormente, la distribución zonal que cubre las áreas de cobertura de la ARCOTEL tiene algunas modificaciones, las cuales están mostradas en la Tabla 2-1.

Tabla 2-1 Distribución Zonal de la ARCOTEL.

Coordinación Zonal	Provincias	Agendas Zonales SENPLADES	Sede
CZ02	Carchi, Esmeraldas, Imbabura, Pichincha.	CZ01, CZ09	Quito
CZ03	Cotopaxi, Chimborazo, Pastaza, Tungurahua.	CZ03	Riobamba
CZ04	Manabí, Santo Domingo de los Tsáchilas.	CZ04, CZ02	Portoviejo
CZ05	Bolívar, Guayas, Los Ríos, Santa Elena.	CZ05, CZ8	Guayaquil
CZ06	Azuay, Cañar, Morona Santiago, EL Oro.	CZ06, CZ07	Cuenca

Además, la ARCOTEL dentro de su distribución desconcentrada posee oficinas técnicas las cuales son:

Dirección de Oficina Técnica de Galápagos, pertenece a la Coordinación Zonal 5, su área de cobertura es la región insular.

Oficina Técnica Loja, pertenece a la Coordinación Zonal 6, su área de cobertura es la provincia de Loja y zonas aledañas.

Oficina Técnica de Lago Agrio, pertenece a la Coordinación Zonal 2, su área de cobertura esta englobada las provincias de Sucumbíos, Orellana y Napo.

2.1.3 DERECHOS Y OBLIGACIONES DE USUARIOS Y PRESTADORES DE SERVICIOS DEL SISTEMA DE TELEFONÍA MÓVIL AVANZADA

Los derechos y obligaciones de usuarios para la prestación y suscripción de servicios de telecomunicaciones son muy variados, con el fin de satisfacer los requerimientos impuestos por la Agencia, entre los más significativos en materia de interrupción de servicios podemos citar los siguientes artículos.

LOT, Título II, Capítulo II Artículo 20, La ARCOTEL, “determinará las obligaciones específicas para garantizar la calidad y expansión de servicios de

telecomunicaciones (...) o establecer las limitaciones requeridas para la satisfacción del interés público, todo lo cual será de obligatorio cumplimiento” [3].

Las empresas públicas que presten servicios de telecomunicaciones y las personas naturales o jurídicas delegatarias para prestar tales servicios, deberán cumplir las obligaciones establecidas en esta Ley, su reglamento general y normas emitidas por la Agencia de Regulación y Control de las Telecomunicaciones para garantizar la calidad, continuidad, eficacia, precios y tarifas equitativas y eficiencia de los servicios públicos.” [3].

LOT, Título II, Capítulo II.

Artículo 2, Delegación, La ARCOTEL, para otorgar los títulos habilitantes a prestadores de servicios de telecomunicaciones por delegación.

Inciso a. “Para las empresas de economía mixta en las cuales el Estado tenga la mayoría accionaria, el otorgamiento de títulos habilitantes para el uso o explotación de espectro radioeléctrico o para la prestación de servicios de telecomunicaciones, se sujetará al interés nacional y respetará los plazos y límites fijados en la Ley y en las regulaciones que para efecto emita la Agencia de Regulación y Control de las Telecomunicaciones.” [3].

Inciso b, “Para el caso de empresas públicas de propiedad Estatal de los países que forman la comunidad estatal, la delegación para el uso o explotación del espectro radioeléctrico o para la prestación de servicios públicos de telecomunicaciones, podrá hacerse de forma directa. En todos los casos, la delegación se sujetará al interés nacional y respetará los plazos y límites fijados en esta ley y en las regulaciones que para el efecto emita la Agencia de Regulación y Control de las Telecomunicaciones” [3].

LOT, Título III (Derechos y Obligaciones), Capítulo I (Abonados, clientes y usuarios)

Los derechos de los abonados en materia de interrupción de servicios se expresan a continuación.

Artículo 22. Derechos de los abonados, clientes y usuarios.

Numeral 1. “A disponer y recibir servicios de telecomunicaciones contratados de forma continua, regular, eficiente, con calidad y eficacia.” [3].

Numeral 10. “A que su prestador le informe oportunamente sobre la interrupción, suspensión o averías de los servicios contratados y sus causas.” [3].

LOT, Título II (Derechos y Obligaciones), Capítulo II (Prestadores de Servicios de Telecomunicaciones).

Las obligaciones de los prestadores de servicios de telecomunicaciones en materia de interrupción de servicios se expresan a continuación.

Artículo 24. Obligaciones de los prestadores de servicios de telecomunicaciones.

Numeral 2. “Prestar servicio de forma obligatoria, general, uniforme eficiente, continua, regular, accesible y responsable, cumpliendo las regulaciones que dicte la Agencia de Regulación y Control de la Telecomunicaciones y lo establecido en los títulos habilitantes.” [3].

Numeral 3. “Cumplir y respetar esta ley, sus reglamentos, los planes técnicos, normas técnicas y demás actos generales o particulares emitidos por la Agencia de Regulación y Control de las Telecomunicaciones.” [3].

Numeral 6. “Proporcionar en forma clara, precisa, cierta, completa y oportuna toda la información requerida por la Agencia de Regulación y Control de las Telecomunicaciones (...), en el ámbito de sus competencias, en los formatos, plazos y condiciones establecidos por dichas autoridades.” [3].

Numeral 20. Informar a la ARCOTEL sobre “modificaciones esenciales a las condiciones de las redes, a las condiciones de interconexión, acceso u ocupación y a la prestación de los servicios de conformidad con la normativa aplicable.” [3].

2.1.4 RÉGIMEN SANCIONATORIO

Las sanciones dentro de prestación de servicios de telecomunicaciones se imponen para garantizar el cumplimiento de la ley, además de velar por la satisfacción de prestadores de servicios como de usuarios, debido a que la prestación del servicio tiene por obligación cumplir las expectativas mínimas impuestas por la ARCOTEL.

Las secciones se exponen dentro de la Ley Orgánica de Telecomunicación, para materia de interrupciones en el servicio móvil avanzado, se expone los siguientes puntos.

LOT, Titulo XIII, Capítulo I (Infracciones).

En el apartado b numeral del artículo 117, especifica que se considera una infracción de primera clase, en el numeral 2 “falta de notificación sobre la interrupción total o parcial del servicio por causas programadas o no programadas, de conformidad con el procedimiento que emita la Agencia de Regulación y Control de las Telecomunicaciones a tal efecto o que consten en los títulos habilitantes.” [3], en el numeral 7 “Suministrar al (...) o a la Agencia de Regulación y Control de las Telecomunicaciones información inexacta o incompleta sobre aspectos que estos hayan solicitado, de conformidad con lo dispuesto en esta Ley y sus reglamentos.” [3], en el numeral 12 “la realización de cambios o modificaciones técnica a las estaciones para la prestación de servicios de radiodifusión o a las redes de telecomunicaciones, cuando afecten a la prestación del servicio, sin notificar previamente a la Agencia de Regulación y Control de las Telecomunicaciones y obtener la autorización pertinente.” [3]. En el apartado b artículo 188, se especifica que se considera como infracción de segunda clase, en el numeral 1 “Interrumpir de forma total o parcial el servicio, sin autorización o por causas imputables al prestador de servicios, conforme con lo establecido en la normativa secundaria y en los títulos habilitantes” [3], numeral 21 “Suspender la prestación de servicios de telecomunicaciones sin causa justificada, (...)” [3], numeral 29 “La incidencia en la comisión de cualquier infracción de primera clase, dentro de un periodo de seis meses, (...)” [3]. En el apartado b artículo 119, se especifica que se considera como infracción de cuarta clase numeral 6 “La reincidencia en la comisión de cualquier infracción de tercera clase dentro de un periodo de seis meses, (...)” [3].

LOT, Titulo XIII, Capítulo II (Sanciones).

Las sanciones especificadas dentro de la LOT, se especifican de acuerdo a las clases de infracción presentes a dicha ley, la multa se impondrá de acuerdo al tipo de clase de infracción, de acuerdo con el artículo 122, monto de referencia, cada una de estas clases se tiene un monto de referencia para el cual se obtendrá de acuerdo al monto total que el infractor haya declarado en su última declaración de impuestos, con

relación al servicio o título habilitante en el cual se produjo la infracción, sólo en los casos que no se pueda determinar el monto de referencia las multas se basarán en las expuestas en la Tabla 2-3. Según el artículo 121 las sanciones serán las que se especifican en la Tabla 2-2 [3].

Tabla 2-2 Sanciones para las y los prestadores de Servicio de Telecomunicaciones.

Clase	Primera	Segunda	Tercera	Cuarta
Monto de referencia en salarios básicos	Hasta cien	Entre ciento y trescientos	Entre trescientos uno hasta mil quinientos.	Entre mil quinientos uno hasta dos mil.

Tabla 2-3 Monto de referencia para sanciones.

Clase	Primera	Segunda	Tercera	Cuarta
Multa en porcentaje del monto de referencia	Entre 0.001% y el 0.03%	Entre el 0.031% y el 0.07%	Entre el 0.071% y el 0.1%	1%*

* Válido cuando sea una reincidencia de una infracción que se origine en la tercera clase, en el caso de no originarse la sanción será la revocatoria del título habilitante [3].

2.1.5 NORMATIVA DE INTERRUPCIONES DEL SERVICIO DE TELEFONÍA MÓVIL AVANZADA (SMA)

De acuerdo con lo mencionado en las secciones anteriores, la ARCOTEL entre sus funciones tiene a cargo el control de las interrupciones del servicio móvil avanzado. Una interrupción se considera cuando cualquiera de los servicios voz, datos y/o SMS, cesan su servicio a los usuarios, ya sea de manera parcial o completa, en un área de cobertura para la cual estaba prevista su funcionamiento en el registro de estructuras de la ARCOTEL, la interrupciones pueden producirse de manera programada para labores de mantenimiento o reemplazo de equipos por parte del operador, así como no programadas debido a eventos fortuitos de índole no imputable a el operador de telecomunicaciones, como por ejemplo pueden ser cortes de fibra óptica, fallas en equipos entre otros. El procedimiento consecuente para el manejo de las interrupciones

está basando en el CONTRATO DE CONCESIÓN PARA LA PRESTACIÓN DEL SERVICIO MÓVIL AVANZADO, que contempla todos los escenarios para los dos tipos de interrupciones y las posibles sanciones que se aplican en caso de incumplimiento de alguna de las operadoras de servicio móvil.

Contrato de concesión para la prestación del Servicio Móvil Avanzado.

En el Capítulo Octavo, Prestación de los Servicios Concesionados, establece que el prestador de servicios de telefonía móvil avanzada debe mantener la continuidad de los servicios concesionados, de acuerdo a lo establecido en la LOT, en caso de no cumplir con esta disposición, el operador debe acogerse a la normativa descrita en el Contrato de concesión para la prestación del Servicio Móvil avanzado, en donde se especifica en el Capítulo Octavo, cláusula treinta y cuatro punto dos que la el prestador de servicios no puede interrumpir, discontinuar o suspender dichos servicios ya sea de manera parcial como total, sin antes tener previa autorización de la ARCOTEL, y esta autorización únicamente entrará en vigencia luego de la aprobación dada por este ente. [24]

La interrupción del servicio móvil avanzado únicamente será permitida en los siguientes casos.

2.1.5.1 Interrupción Programada

El prestador de servicios requiera realizar trabajos de mantenimiento o mejoras tecnológicas en las infraestructuras que prestan dichos servicios, de tal forma que interrumpan el normal funcionamiento de la misma por más de una hora, en este caso el prestador de servicios deberá notificar con una anterioridad no menor a tres días a la ARCOTEL de manera escrita y a sus usuarios a través de un medio comunicación masiva y del servicio que va a ser afectado. [24]

2.1.5.2 Interrupción no Programada

Se produzca un evento no planificado por el prestador de servicio, en el que los servicios ofrecidos se vean parcial o totalmente afectados, como consecuencia el prestador de servicios deberá notificar a la ARCOTEL en 30 minutos subsecuentes luego de haberse producida la interrupción, dentro de los próximos 5 días laborales el prestador de servicios deberá presentar pruebas que acrediten la existencia de la

interrupción y en un plazo de 15 días la ARCOTEL calificara si evento presentado corresponde a lo declarado por el prestador de servicios, si la interrupción tiene una duración mayor a 4 horas, el prestador de servicios estará en la obligación de informar a sus usuarios a través de cualquier medio. [24]

2.1.6 SERVICIO MÓVIL AVANZADO EN EL ECUADOR

Servicio móvil avanzado según el Título I, Artículo 3, se puede definir un servicio de telecomunicaciones del servicio móvil terrestre aquel que está destinado a “permitir y facilitar la transmisión y recepción de signos, señales, textos, videos, imágenes, sonidos o cualquier información de cualquier naturaleza, (...)” [3].

Para noviembre del 2018 en el Ecuador a nivel nacional se encuentran activas 15.668.883 líneas en una población de 17.002.872 personas, significando una densidad nacional de líneas de 92.15%, de las cuales 8.211.54 líneas activas pertenecen a CONECEL, 4.649.998 líneas activas pertenecen a OTECEL, y 2.807.331 líneas activas pertenecen a CNT. [1]

Para octubre del 2018 se registraron 16678 radiobases (RBS), que permiten prestar el servicio móvil avanzado para las tecnologías 2G (CDMA, GSM), 3G (UMTS) y 4G (LTE), de las cuales CONECEL posee 2507 RBS GSM, 4321 RBS UMTS y 2193 RBS LTE, OTECEL posee 1464 RBS GSM, 2720 RBS UMTS y 1068 RB LTE y CNT posee 747 RBS LTE y 1658 RBS UMTS. [2]

2.2 IMPLEMENTACIÓN DE PLATAFORMA WEB

La propuesta para el proyecto consta de tres aplicativos dispuestos en una estructura REST API - CSR- SPA (Transferencia de Estado Representacional - Renderizado al Lado del Cliente - Aplicación de una sola página), lo que significa que el aplicativo realiza la renderización de la página en el computador del usuario mediante un archivo HTML incompleto y el JavaScript que se encuentra alojado en el cliente es el encargado de solicitar los datos al servidor y completar el HTML conforme el usuario manipule la aplicación Web, las peticiones al servidor de procesamiento de datos es realizada de acuerdo a los requerimientos del usuario y debido a la arquitectura REST las peticiones normalmente consisten en unos pocos bytes de información requerida.

El CSR se puede considerar subóptimo frente al SSR, debido al tiempo de carga inicial que requiere entre otras claras características. La razón de optar por el CSR, se debe a que el aplicativo debe usar mapas dinámicos, el problema se produce debido a que la mayoría de librerías de código abierto de mapas, no comprueban la existencia del DOM para cargar, lo cual causa conflictos para funcionar en el SSR, existen algunas propuestas para solucionar el problema como por ejemplo crear una abstracción del DOM, Controlar el error de existencia de DOM mediante un control del Error Boundary y volver a cargar automáticamente en el cliente el mapa, importar la librería sin cabeceras mediante importación dinámica, etc., pero estas metodologías son dependientes del entorno y compilación, y es muy probable que no pueda funcionar en algunos casos, por otro lado la diferencia no es considerablemente notoria y en algunos casos, únicamente es notoria cuando se realiza un análisis de rendimiento mediante el DevTool, finalmente es importante mencionar que para realizar un cambio de CSR a SSR, no requiere mayores cambios en los componentes del aplicativo (Salvo Leaflet), únicamente se requiere realizar algunas configuraciones de React y del servidor como se muestra en [25].

2.2.1 ESTRUCTURA DE LA PLATAFORMA

2.2.1.1 Lado del Cliente

Para el desarrollo de la aplicación en el lado del cliente se plantea una estructura CSR/SPA como mencionada anteriormente, para ello se hace uso de dos librerías ampliamente usadas en la actualidad como son React.js y Redux.js. React es una librería para diseño de interfaces de usuario que optimiza el uso del DOM, mediante un DOM virtual y Redux es una librería que actúa como contenedor para el manejo del estado de la aplicación, es decir React maneja la vista de la aplicación mientras Redux se encarga de manejar el estado de la aplicación, para el caso de la propuesta del proyecto se presenta una estructura como se muestra en la Figura 2-2.

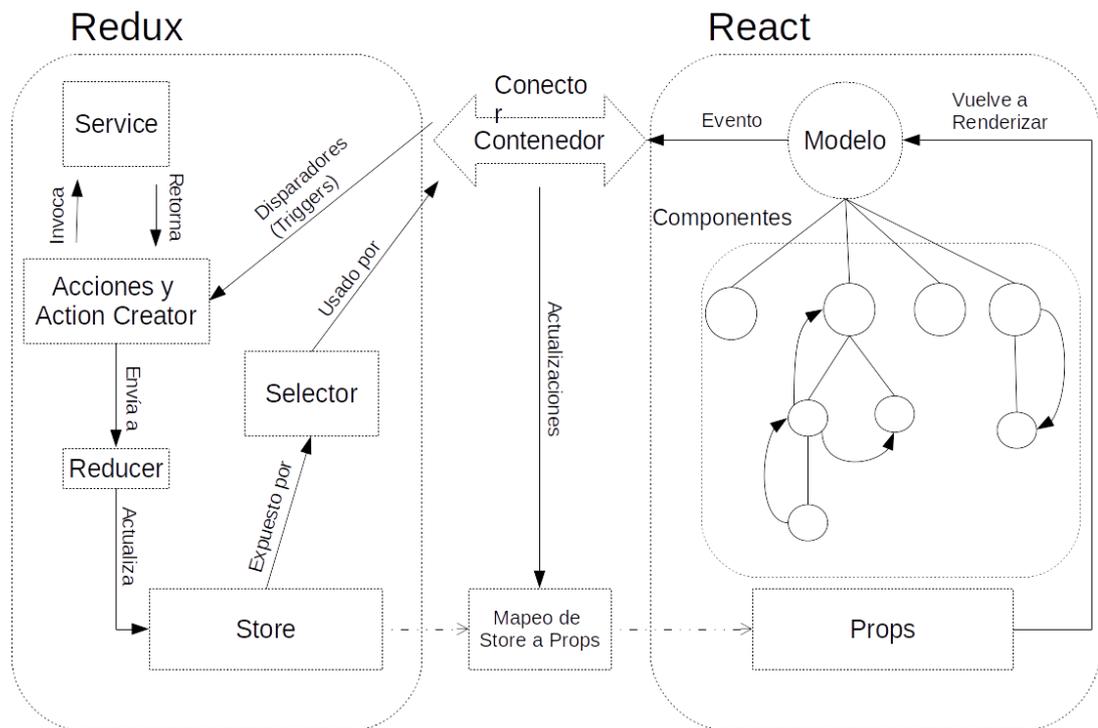


Figura 2-2 Estructura de aplicación para manejo del estado global.

Fuente: autor

React.js

La estructura de React se basa en componentes, los cuales son elementos de código independientes y reutilizables, cada componente corresponde a una parte de la interfaz de usuario, por ende una aplicación se compone de un conjunto de componentes que interactúan a través de una estructura de árbol, comunicándose a través de propiedades y estados de aplicación, estas propiedades y estados dependen del tipo de componente que se vaya a usar, en React podemos encontrar de manera general con dos tipos de componentes, los tipos funcionales que solo tienen propiedades y los componentes de clase que tienen ciclos de vida y propiedades, una división más específica en cuanto al manejo de componentes se presenta, los tipos de acuerdo al tipo de comportamiento que vaya a realizar el componente, en donde se puede encontrar a los componentes Stateless que son componentes de React que no tienen ni manejan un estado, Stateful son componentes que utilizan la encapsulación en clases y que contienen un estado por ende lo manejan, PureComponent es un tipo de componentes que tiene encapsulación en clases pero no contienen ni manejan un

estado y HOC o componente de orden superior son funciones que toman como parámetro un componente y retornando con funcionalidad extendida. [26]

Otro concepto importante en React es el estado, el término se puede abstraer como “la condición de componente con respecto a sus circunstancias o atributos”, el concepto es importante debido a que la aplicación para su funcionamiento requiere un manejo de estado, el caso de React el manejo de estado se realiza mediante el método “setState” el cual es propio del componente, el estado puede ser pasado a otros componentes (únicamente de padre a hijo y viceversa) a través de las propiedades (props), el funcionamiento es bastante sencillo debido a la estructura “one way data-binding” de React, pero conforme el aplicativo crece el manejo de estado se va complicando debido a los múltiples anidamientos de componentes que puede tener el aplicativo dependiendo del tamaño del proyecto, limitando la escalabilidad, ya que un manejo de estado puede requerir realizar varios saltos entre componentes para poder entregar o recibir información de otros componentes, el proceso puede resultar muy complejo para garantizar un óptimo manejo del flujo de propiedades como se muestra en la Figura 2-3 a, para solventar estos inconvenientes existen contenedores de manejo de estado como Redux o Flux, los cuales reducen la complejidad del flujo de propiedades mediante un único store para el manejo de estado como se muestra en la Figura 2-3 b. [26]

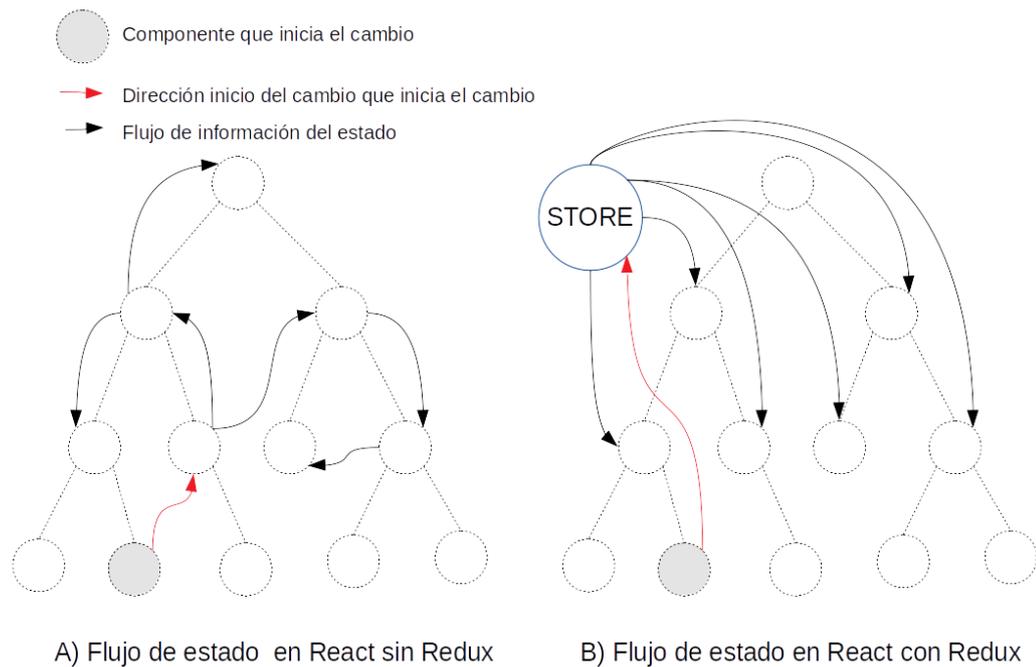


Figura 2-3 Flujo es estado en aplicación React.

Fuente: autor

Redux

Es un librería para el manejo de estado de la aplicación, y este se basa en tres principios de funcionamiento expresadas como, “Todo estado de la aplicación está contenido en un único store”, “El estado debe ser inmutable por el usuario y la única forma de modificar el estado es a través de la emisión de una acción que indique el cambio”, “Para controlar como el store es modificado por las acciones se usan únicamente reducers que sean funciones puras”, es importante mencionar que el estado que está almacenado en un único store es accesible desde cualquier lugar dentro de la aplicación lo que solventa el problema de manejo de estado debido al anidamiento de componentes que es generado por React. [27]

Par el caso del aplicativo planteado se hace uso de las dos tecnologías, React para vistas y Redux para manejo de estado del aplicativo, además en determinados componentes que no requieren recorrer su estado más allá de una componente padre a hijo o viceversa, se mantendrá el manejo del estado mediante el método “setState”, esto debido a que Redux permite este funcionamiento dual para el manejo de estos.

2.2.1.2 Lado del Servidor

De acuerdo a la estructura planteada se definió una aplicación con arquitectura REST, la arquitectura permite desarrollar los servicios del cliente y del servidor de manera independiente, por otro lado REST presenta la característica de peticiones que requieren pequeñas cantidades de bits para renderización en el cliente, debido a esto las solicitudes deben ser resueltas lo más rápido posible, para minimizar el impacto del servicio de datos y contenido por parte del servidor, en experiencia de usuario, por esta razón se propone un servidor Node.js

Node.js

Es un entorno de ejecución de JavaScript desarrollado basado en el motor V8 de Google, usa un modelo de entrada/salida no bloqueante lo que le permite ser liviano y eficiente, esto debido a que solo posee un solo hilo de ejecución, esta característica de Node le permite ser muy rápido en responder solicitudes, además le permite establecer una cantidad significativa de conexiones simultaneas al servidor, ya que todos las solicitudes están disponibles mediante el bucle de eventos de JavaScript, por lo que la cantidad de conexiones simultaneas al servidor de Node se encuentra limitada por lo recursos de hardware y configuración del sistema operativo. Node está diseñado para arquitecturas de aplicaciones que no procesan muchos datos, para uso como middleware de servicios que realizan el trabajo pesado, es decir permite puede consumir servicios desarrollados de manera externa al servidor y sobre como ya se mencionó antes, es usado cuando se requiere servir a una gran cantidad de solicitudes simultáneas. [28] [29]

2.2.1.3 Estructura de la Plataforma del Cliente

El aplicativo desarrollado para el manejo de interrupciones desarrollado mediante la librería de React, tiene una estructura de árbol como se muestra en la Figura 2-4, la estructura mostrada corresponde únicamente a los componentes desarrollados por los autores, es decir se excluyen componentes correspondientes a librerías de terceros a excepción de las librerías de enrutamiento dinámico React Router y al contenedor de estado React Redux (Debido a la importancia que tienen en la estructura planteada), esta exclusión de componentes se debe a la extensión que llegaría a tener el diagrama, cada componentes está diseñado para simplificar el flujo

de información inter componente, ya sea cuando se requiere almacenar el estado en el Store (contenedor global) o cuando se requiera usar un estado local mediante el paso de propiedades entre componentes, esto debido a que si todo el estado de la aplicación es enviado al Store de la aplicación, la estructura del estado se complica para un seguimiento debido a adición de estados que son usados de manera local únicamente los cuales actúan como estados muertos en el resto de la aplicación, en la Figura 2-5 se muestra el estado global de la aplicación cuando se está utilizando la plataforma es decir que este irá variando dependiendo del proceso que se esté realizando en este momento, pero una mejor visualización referirse a APÉNDICE E: ÁRBOL DE ESTADOS GLOBAL.

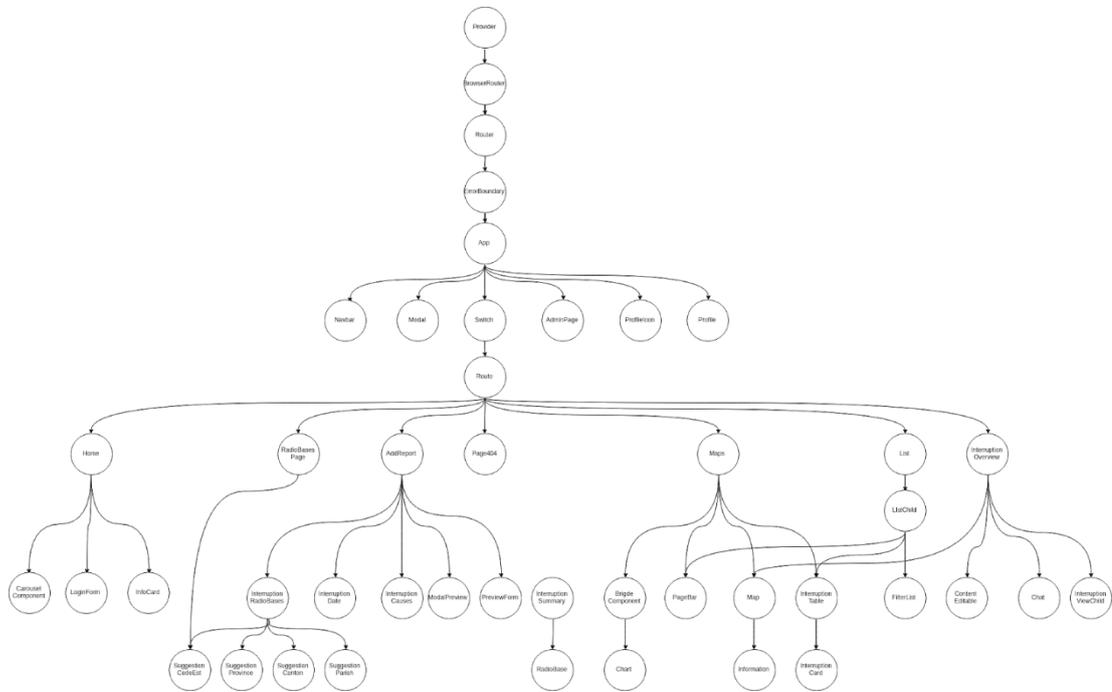


Figura 2-4 Diagrama de árbol de componentes del aplicativo SMA_APP.

Fuente: autor

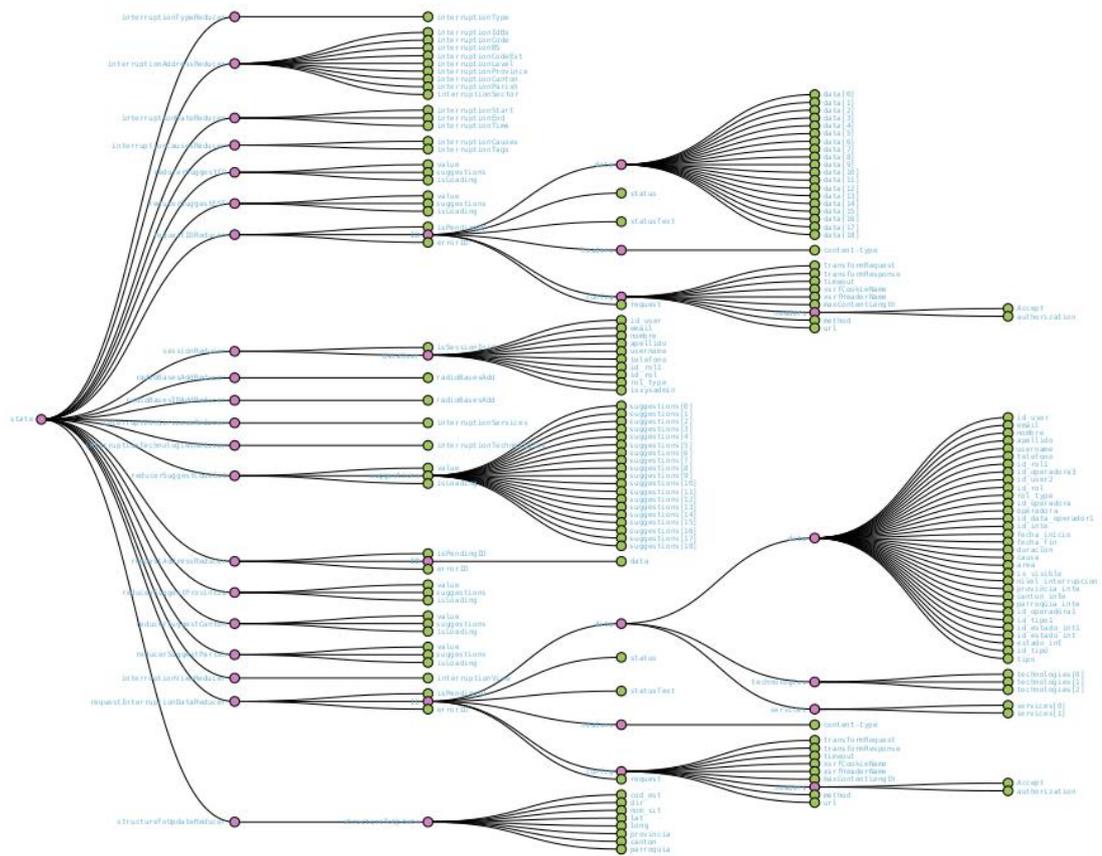


Figura 2-5 Flujo de estado de aplicación, cuando se usa la plataforma.

Fuente: autor

De igual manera para el aplicativo de mapas, la estructura de árbol presentada en la Figura 2-6 corresponde únicamente a los componentes desarrollados por los autores, en este aplicativo el árbol se encuentra menos ramificado, debido que no se usa un contenedor de estado, sino que hace uso de flujo de datos convencional de React, de esta manera se facilita el flujo de datos en el aplicativo pero el proceso de mantener el código se vuelve más complicado debido que existen componentes que tienen una extensión considerable.

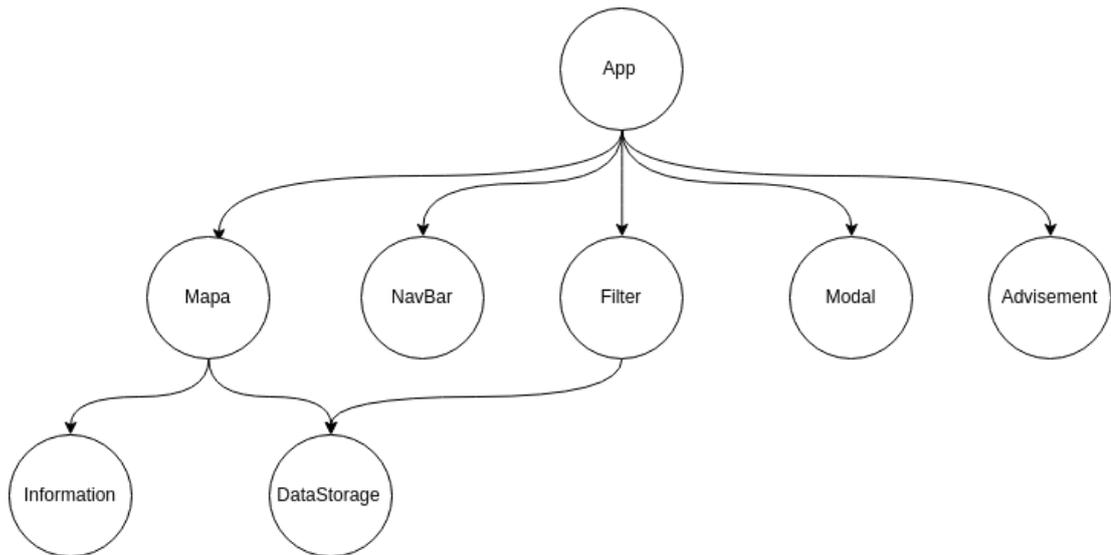


Figura 2-6 Diagrama de árbol de componentes del aplicativo BSS.

Fuente: autor

2.2.1.4 Estructura de la aplicación del Servidor

En la Figura 2-7 se presenta la estructura simplificada de los componentes del servidor, la estructura planteada es propuesta por los autores, debido a que no existe un estándar para un diagrama de un aplicativo de Express.js, dentro del árbol del aplicativo se puede encontrar, App el cual es el punto de entrada del aplicativo es decir toda petición entra y sale por este módulo, DockerConfig corresponde a la forma abreviada de los archivos correspondientes a la configuración del contenedor docker, package.json corresponde al archivo de configuración de la aplicación, README.md corresponde al archivo que contiene a la descripción del aplicativo, Routes corresponde al conjunto métodos de ruta que se derivan de los métodos HTTP y son adjuntados a una instancia de clase express, estos métodos de rutas han sido agrupados por servicio como se observa en la Figura 2-8, Services corresponde al conjunto de servicios no relacionados a un método de ruta pero que son consumidos de acuerdo a requerimientos, en la Figura 2-9 se observa las clases correspondientes a dichos servicios, Core corresponde a la lógica de negocios asociada a cada método de ruta de express como se muestra en la Figura 2-10, DataBase corresponde al conjunto de archivos DDL, DML y Shell que definen la base de datos, configuran y la acondicionan la bases de datos para el funcionamiento inicial del aplicativo, esto archivos se muestran en la Figura 2-11.

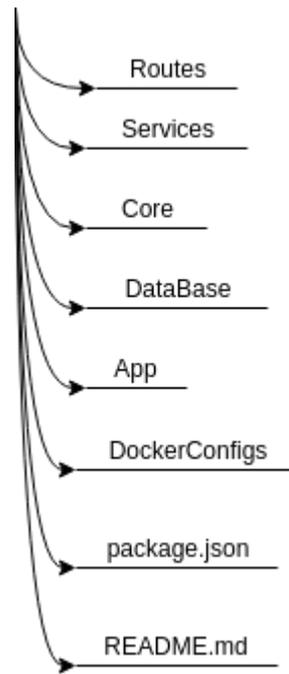


Figura 2-7 Estructura principal propuesta para el aplicativo SMA_API.

Fuente: autor

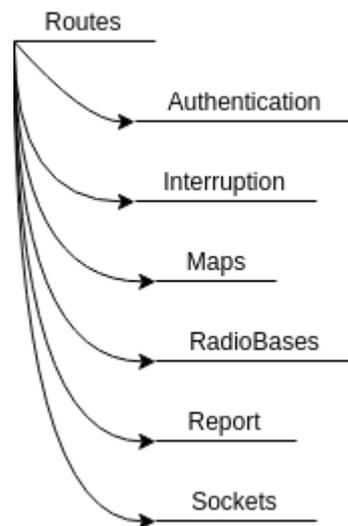


Figura 2-8 Estructura propuesta de la carpeta Routes.

Fuente: autor

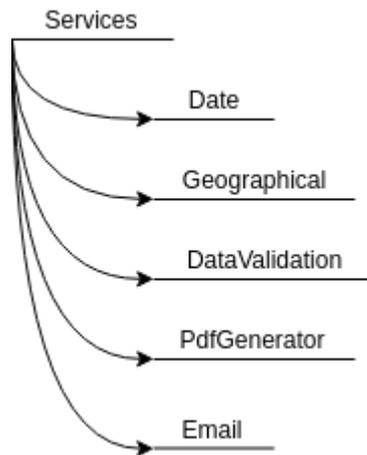


Figura 2-9 Estructura propuesta de la carpeta Services.

Fuente: autor

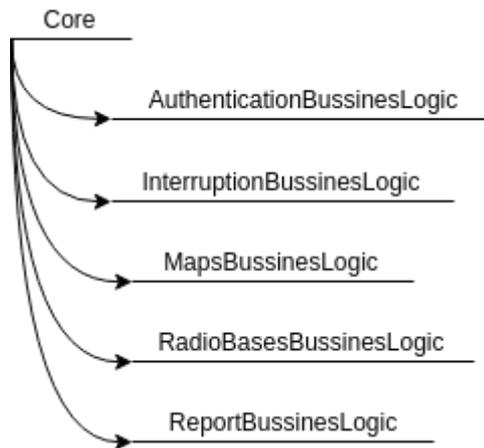


Figura 2-10 Estructura propuesta de la carpeta Core.

Fuente: autor

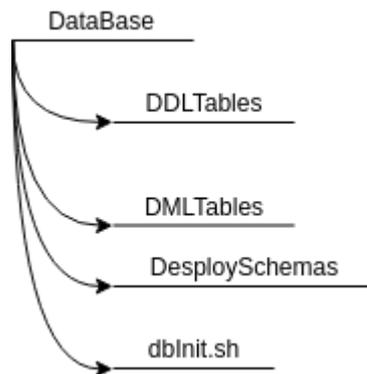


Figura 2-11 Estructura propuesta de la carpeta DataBase.

Fuente: autor

2.2.2 DISEÑO DE BASE DE DATOS

El proceso de diseño de base de datos hace referencia al proceso de simplificar un problema mediante la toma de decisiones de diferentes niveles como son el conceptual, lógico y físico, para desglosar el problema en subproblemas que puedan ser resueltos de manera independiente, el cual puede ser esquematizado mediante un grafo en donde se identifican los elementos correspondientes del problema y las relaciones que existen entre estos elementos. El diseño de base de datos a nivel lógico propuesto para el problema de la plataforma se muestra en la Figura 2-12 para una mejor visualización referirse al APÉNDICE A: ESTRUCTURA DE BASE DE DATOS, no se muestran los niveles conceptuales debido a que únicamente es una abstracción del problema y el físico, debido a la extensión considerable que tiene, el nivel lógico presentado muestra un diseño normalizado a excepción de los campos Provincia, Cantón y Parroquia, debido al problema de asignación, dado que los datos preexistentes deben ser asignados a un registro de la tabla de manera automática, por ello para realizar el proceso de normalización de estos campos mediante la propuesta inicial se propuso de un algoritmo con complejidad $O(n^3)$, que generaba tiempos relativamente largos además de uso de recursos de hardware elevados, el diseño además contempla la seguridad a nivel de acceso a la información como se describe en el apartado de gestión de sesión del aplicativo.

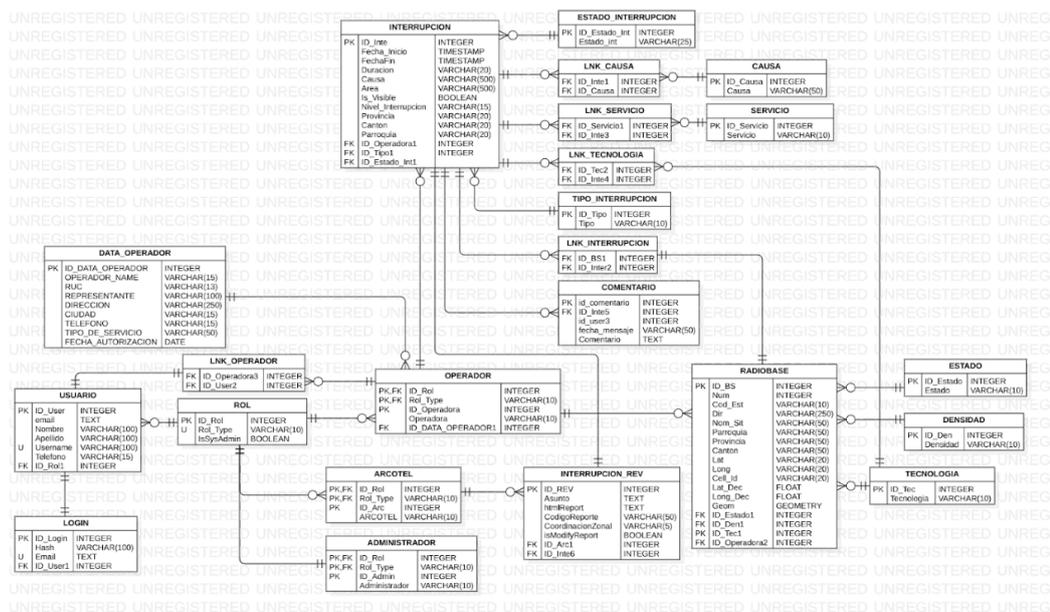


Figura 2-12 Diseño de la base de datos propuesto.

Fuente: autor

2.2.3 GESTIÓN DE SESION DEL APLICATIVO

La gestión de sesión hace referencia al registro, acceso y manejo de sesiones del usuario, el proceso propuesto se divide en dos niveles de validación de roles y acceso, los cuales son el nivel de base de datos y el de programa de aplicación, estos niveles se describen a continuación

2.2.3.1 Base de Datos

Los diferentes roles (Defínase como rol al tipo de usuario perteneciente a la plataforma) tienen una base de datos común para servirse, por esa razón es primordial restringir el acceso a los datos de acuerdo a sus credenciales, para cumplir con esta tarea se requiere que un rol específico sea identificado como tal y acceda solo a los datos que le compete, evitando desde la base de datos que el usuario acceda a datos a los cuales no tiene acceso, ya sea por errores lógicos del código como por ataques de inyección de SQL, para ello se menciona dos técnicas ampliamente usadas en el diseño de base de datos para solventar esta problemática.

Exclusividad (Exclusive Relationship ARC)

La exclusividad se usa cuando una entidad puede estar relacionada con dos o más entidades, pero no de manera simultánea, es decir las entidades dentro de un arco exclusivo se excluyen mutuamente, su funcionalidad es análoga a la de una compuerta lógica XOR. [30]

La exclusividad es un técnica ampliamente utilizada, de manera teórica es una opción factible para resolver el problema, pero indagando en la comunidad en línea, se encuentran controversias en su implementación, en donde se menciona que esta metodología debe ser usado en casos puntuales y su desarrollo es poco flexible, ya que cuando usamos un arco exclusivo el procedimiento describe que la entidad raíz tiene dos o más claves foráneas donde solo una de ellas puede ser no nula, este tipo de restricción es de programa de aplicación, por lo que no existe realmente una restricción real en la base de datos, entonces nada impide que se pueda crear un registro corrupto generando más de una clave foránea no nula lo cual causaría, no sólo problemas lógicos sino de seguridad.

Entidad Supertipo/Subtipo

Una entidad supertipo, se define como la entidad que agrupa múltiples entidades diferentes dentro de sí y que contiene atributos comunes a sus subtipos, mientras que las entidades anidadas se definen como entidades subtipo. [Barkers ERD Notation]

Las entidades subtipo deben cumplir con dos reglas, las cuales son, regla de exhaustividad que cual dicta que cada entidad supertipo debe ser una instancia de una entidad subtipo y la regla de exclusión mutua, la cual dicta que cada entidad del supertipo puede ser una instancia de una sola entidad subtipo, similar a la funcionalidad de la exclusividad pero con la característica de que no existe claves foráneas nulas. [30]

En base a lo especificado se opta por aplicar el método de Entidad Supertipo/Subtipo debido a sus beneficios frente a la Exclusividad, el diagrama entidad relación para el manejo de registro e inicio de sesión se muestra en la Figura 2-13 en el cual se ve implementada la metodología, como consecuente este diseño de base de datos se implementa mediante Lenguaje de Definición de Datos (DDL).

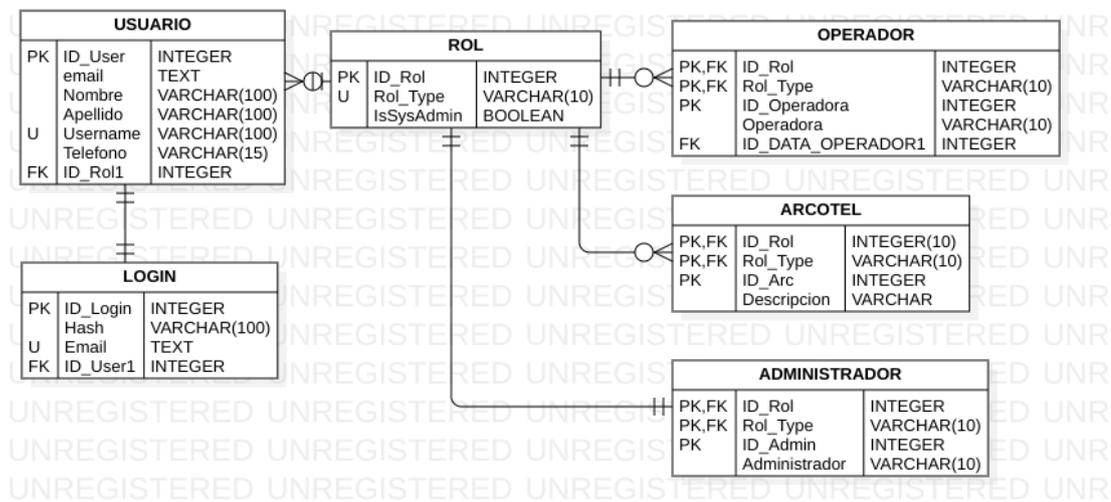


Figura 2-13 Diseño de base de datos para restricción a la información según rol de usuario.

Fuente: autor

2.2.3.2 Programa de Aplicación

El aplicativo está desarrollado con una estructura SPA/CSR (Aplicación de una sola página con renderizado al lado del cliente) y comunicación al servidor de la realiza mediante una arquitectura REST, lo que significa que el servidor de interfaces y el de datos y lógica de servidor están separados, por ende el aplicativo solicitara pequeñas cantidades de datos al servidor cada vez que sea necesaria, debido a esto se requiere una autenticación en cada una de esas solicitudes, este proceso se puede dividir en dos niveles, uno del lado del servidor y otro del lado del cliente, los cuales se describen a continuación.

Lado del Cliente

En el cliente la autenticación para brindar servicios se basa en identificadores de usuario, que son brindados por el servidor, el cual tiene almacenado en sus bases de datos (PostgreSQL y Redis), los identificadores de identidad primarios son el identificador de usuario y el identificador de rol de usuario, además de estas identificadores primarios, existen otros identificadores secundarios (expresados en el diseño de la base de datos en el APÉNDICE A: ESTRUCTURA DE BASE DE DATOS que son usados dependiente de los servicios a consumir.

Lado del Servidor

La identificación del cliente en el servidor se realiza mediante las credenciales directas del cliente (usuario y contraseña), identificadores de sesión e identificadores de usuario, los identificaciones directos son verificados en la base de datos (es importante mencionar que las contraseñas son encriptadas con un algoritmo SHA-256 los identificadores de usuario corresponden a las credenciales que fueron enviados al cliente bajo petición y que almacena en su estado de aplicación (podríamos expresar como identificadores volátiles), los cuales son enviados al servidor para brindar acceso y servicios de acuerdo a estos identificadores y el identificador de sesión es el JWT (JSON Web Token), el cual se usa tanto para mantener el control de inicio de sesión como para autenticar el usuario en cada petición al servidor, este proceso hace uso de middleware en el servidor, el cual es primer nivel de seguridad para ingresar al mismo, permitiendo hacer peticiones al servidor, de esta forma se garantiza que el

acceso a servicios, únicamente sea obtenido por los usuarios que dispongan todas las credenciales correctas, para un determinado servicio.

2.2.4 ESTADO DE ESTACIONES BASE (BSS)

La construcción se basa en la librería de Leaflet. Leaflet es una librería para construir mapas iterativos de Open-Source, presenta ventajas frente a Google Maps Platform, debido a que no tiene costo alguno, además de que al ser una librería de bajo nivel permite un desarrollo en el núcleo de Leaflet es decir se pueden crear componentes propios integrándolos en el mapa sin ningún tipo de restricción, por ello se convierte en una de las mejores alternativas para el desarrollo del aplicativo.

2.2.4.1 Almacenamiento

Este aplicativo para el funcionamiento en la versión que se encuentra actualmente, requiere servirse de datos del servidor con un tamaño aproximado 25 MB, el cual en un servicio Web esta cantidad es considerable, para solucionar el inconveniente se propone el uso de almacenamiento de datos en el lado del cliente, el objetivo que se plantea para esta metodología es que la aplicación tenga la capacidad de funcionar de manera transparente en ambos casos.

El almacenamiento de datos en el lado del cliente dependiendo del tamaño de los datos puede hacer uso de diferentes tecnologías entre las cuales encontramos el “DOM Storage”, “Web SQL” e “IndexedDB”, el DOM Storage es usado para datos relativamente pequeños, en el presente aplicativo se hace eso de este tipo de almacenamiento para manejo de credenciales de sesión, pero no tienen utilidad para el manejo grandes cantidades de datos, Web SQL por su parte es una tecnología de base de datos al lado del cliente que permite almacenar grandes cantidades de datos, pero pese a que los navegadores aún soportan Web SQL, es una tecnología que fue declarada como obsoleta en 2010 por la W3C, por otro lado IndexedDB es un API de bajo nivel que permite el almacenamiento de grandes cantidades de datos al lado del cliente, por lo que es la tecnología a usar para cumplir esta tarea. [31]

IndexedDB

Es una base de datos no relacional orientada a objetos, estructura de funcionamiento es mediante el modelo Key-Value (clave-valor), por lo que pueden

estructurarse objetos complejos mediante índices que son asignados a un propiedad del objeto, uno de los aspectos más significativos de IndexedDB es su política de “same-origin” la cual indica que cada base de datos pertenece a un único origen o dominio, lo que significa que la base de datos sólo puede ser accedida por el aplicativo que creó dicha base de datos.

De esta manera el paradigma de estructura de la base de datos del cliente y del servidor son completamente diferentes, por ello la manera de realizar las operaciones CRUD son completamente diferentes, una vez comprendido estos conceptos se implementó la base de datos en el lado cliente como se muestra en la Figura 2-14 el código implementado para el funcionamiento no se muestra debido a su considerable cantidad de líneas (aproximadamente 500 líneas).

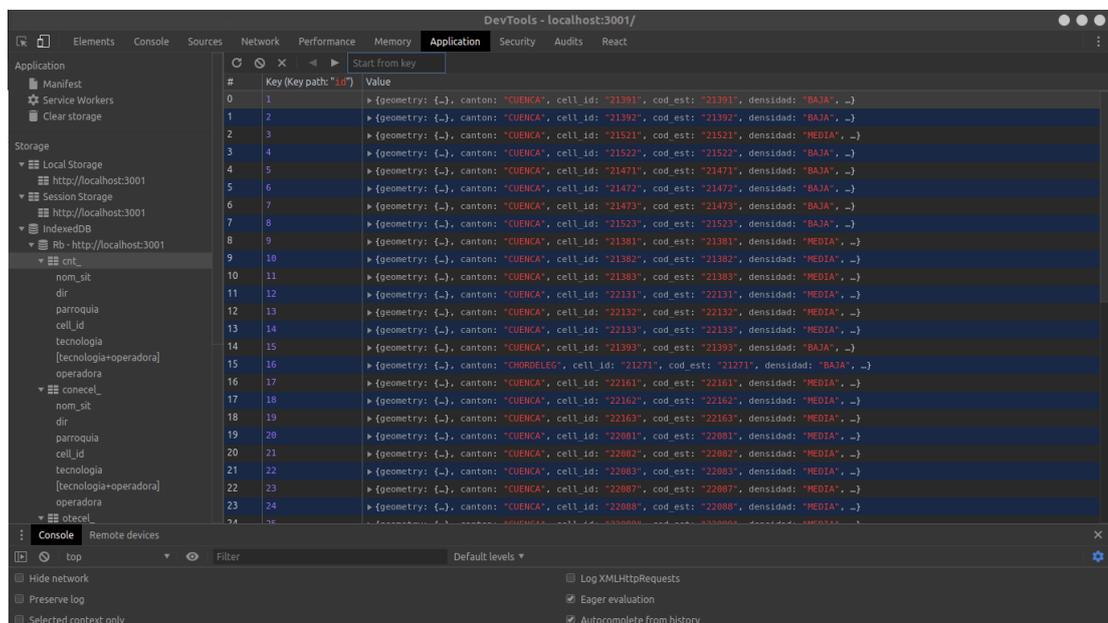


Figura 2-14 Base de datos al lado del cliente mediante IndexedDB.

Fuente: autor

Con las metodologías ya implementadas, el aplicativo es capaz de funcionar de manera y transparente tanto con dependencia del continuo servicio de datos por parte del servidor como de manera aislado del servicio de datos por parte del servidor, lo que permite tener la capacidad de realizar todas las funciones que se presentaron anteriormente de manera “Offline”.

2.2.5 CONSUMO DE SERVICIOS

2.2.5.1 Estructura de Consumo de servicios

La plataforma está dispuesta de manera general en tres servicios a los que hemos nombrado, “BSS”, “SMA_APP” y “SMA_API”, dispuestos como se muestra en la Figura 2-15, la cual expresa el flujo de los servicios de la plataforma, donde BSS es un servidor/servicio de interfaces de usuario para la visualización de radiobases en mapa iterativo, SMA_APP, es un servidor/servicio de interfaces de usuario para el manejo de la interrupciones del SMA y SMA_API es un servidor/servicio de datos y procesamiento para los aplicativos SMA_APP y BSS, además para que la plataforma funcione como se planteó anteriormente, requiere conectarse a servicios externos los cuales son, un servidor SMTP de correo electrónico (para pruebas se creó un correo Gmail) y el servidor de mapas en este caso se hace uso de OpenStreetMap que es de licencia abierta.

Es importante mencionar que los servicios SMA_APP y BSS pueden ser distribuidos de manera aislada o directamente por el servidor SMA_API, sin necesidad de modificar el código fuente de ninguno de los servicios en mayor medida, solo es necesario asignar la carpeta “build” que contienen los archivos estáticos de cada uno de los servicios a un método de ruta del dominio o subdominio del servidor. Para fines de explicativos se mantendrá la estructura presentada.

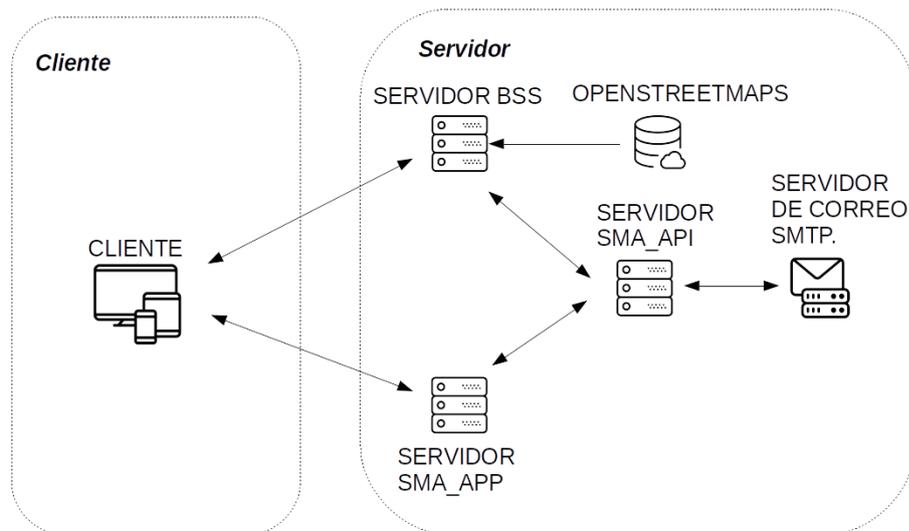


Figura 2-15 Flujo de servicios de los aplicativos.

Fuente: autor

2.2.5.2 Estructura de Flujo y Almacenamiento de datos

En apartados anteriores se presentaron algunas metodologías de consumo de datos, ahora se describe el flujo y almacenamiento de datos a través de los servicios, la estructura general se muestra en la Figura 2-16, en donde se muestra dos notaciones, la notación con terminación en punta de flecha invertida corresponde al flujo de datos de SMA_APP y la notación con terminación en rombos pertenece a BSS.

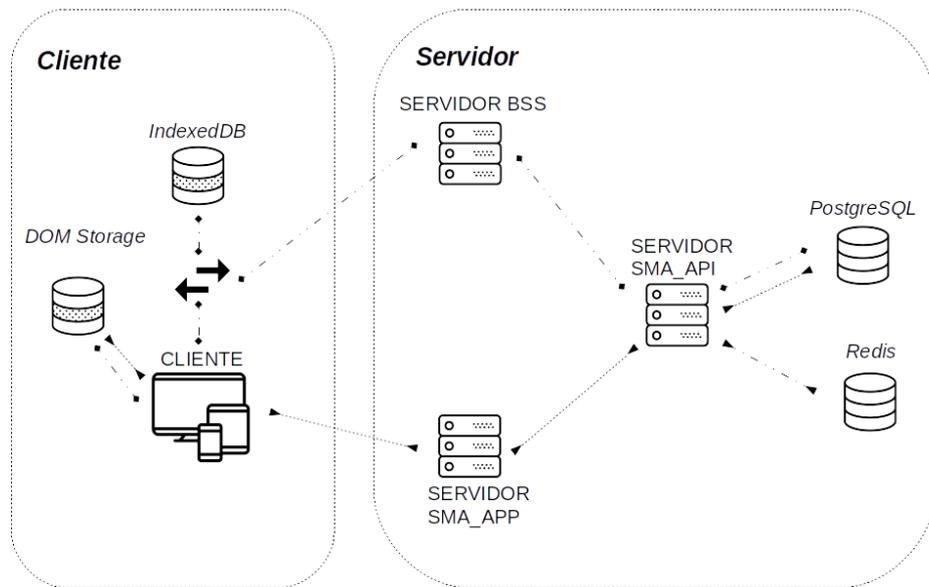


Figura 2-16 Flujo de datos de los aplicativos.

Fuente: autor

Almacenamiento

SMA_APP

El aplicativo hace uso de tres tecnologías de almacenamiento, en las que podemos encontrar el DOM Storage, PostgreSQL y Redis.

DOM Storage corresponde al lado del cliente y su función es la de almacenar las credenciales de sesión del aplicativo como los JWT, esto ya que una de las medidas de seguridad propuestas dicta, que toda petición de un servicio necesita identificarse para ser resuelto, por ello esta credencial no puede ser almacenada en variables locales o el cache debido a su volatilidad, además que este sistema de almacenamiento sigue la política de “same-origin” mencionada anteriormente.

PostgreSQL corresponde al lado del servidor, cumple la función de ser la base de datos principal del aplicativo es la encargada de servir y almacenar datos a los aplicativos, además y cumple un papel importante en el manejo de algunas credenciales de servicios, ya que es la encargada de almacenarlos y en algunos de los casos de verificar su validez mediante operaciones CRUD.

Redis corresponde al lado del servidor y actúa como la contraparte del DOM Storage, ya que es la encargada de almacenar las credenciales de sesión pero en el servidor, la ventaja de Redis frente a PostgreSQL para el manejo de este tipo de datos, es que se encuentra optimizada para almacenamiento de este tipo de datos ya que se basa en un almacenamiento en tablas de hashes pero con la ventaja de ser persistente, además que es una base de datos estructurada en memoria lo que le permite ser rápida en sus operaciones CRUD.

BSS

El aplicativo hace uso de tres tecnologías de almacenamiento, en las que podemos encontrar el DOM Storage, PostgreSQL e IndexedDB.

Las tecnologías de almacenamiento DOM Storage y PostgreSQL tienen el mismo comportamiento que lo descrito en el caso de SMA_APP, por lo que no se mencionan.

Por otro lado IndexedDB es una tecnología que corresponde al lado del cliente y es usada para almacenar grandes cantidades de datos, para el caso de la aplicación propuesta, se usa para obtener una especie de pseudo replicación de base de datos del servidor, la pseudo replicación propuesta consiste en copiar los registros que competen la información de las radiobases de una base de datos relacional (PostgreSQL) en el servidor a una compatible base de datos no relacional (IndexedDB) en el cliente, con el fin de servir de datos al cliente sin requerir petición de datos al servidor, sino que obtenerlos de manera local si el cliente desea aceptar las condiciones de almacenamiento.

Flujo de Datos

El flujo de datos de la aplicativos se muestra en la Figura 2-16 y se explica a continuación.

SMA_APP

El flujo de datos para el aplicativo SMA_APP es bastante simple, el cliente hace una petición del servicio Web a SMA_APP y el servidor le devuelve la interfaz de usuario, cuando el cliente requiere iniciar sesión envía sus credenciales directas de acceso (correo y contraseña), entonces el servidor recibe las credenciales y valida estas credenciales con las que se encuentran la base de datos de PostgreSQL, luego genera un JWT que es una credencial de sesión que almacena en la base de datos Redis y una copia es enviada hacia el cliente para ser almacenada en el DOM Storage, aparte de la credencial de sesión, el servidor envía una serie de credenciales secundarias las cuales son identificadores de servicio y de usuario, estas últimas son almacenados como variables volátiles de acuerdo al servicio que ocupe el usuario, cuando el usuario ejecute algún componente del aplicativo que requiera una petición de datos al servidor, este enviará la credencial de sesión almacenada en el DOM Storage, las credenciales de usuario y credenciales del servicio a consumir, al llegar la solicitud al servidor este verifica todas las credenciales antes mencionadas, extrayendo los datos desde el servidor de datos PostgreSQL y finalmente retorna los datos del servicio al cliente que lo solicitó.

BSS

El flujo de datos se diferencia del aplicativo SMA_APP porque este no hace uso manejo de sesión, por ello el proceso varía en algunos pasos como se explica a continuación, el cliente hace una petición al servicio Web BSS y el servidor devuelve la interfaz de usuario, cuando la interfaz de usuario termina de renderizar el cliente se encuentra con modal que pide elegir el método de flujo de datos. En el método 1, servicio de datos desde el servidor, en este caso la petición al servidor es enviada, el servidor recibe las credenciales del servicio como un usuario invitado, extrae los datos desde la base de datos PostgreSQL y los retorna hacia el cliente, este proceso es realizado cada vez que el cliente requiere datos desde el servidor. En el método 2 servicio de datos de manera local, en este caso el usuario realiza la petición al servidor, este recibe las credenciales como usuario invitado, extrae los datos desde la base de

datos PostgreSQL y retorna hacia el cliente, al llegar los datos hasta el cliente este almacena las credenciales de almacenamiento local en el DOM Storage y genera una pseudo replicación de los datos en la base de datos IndexedDB, cuando el cliente requiera servirse de datos bajo algún requerimiento de BSS, extraerá las credenciales del DOM Storage y las verifica, luego extraerá los datos desde la base de datos local IndexedDB, el proceso se mantiene hasta que el usuario decida eliminar la base de datos local y por lo tanto las credenciales, volviéndose a repetir el primer paso del proceso.

2.2.6 CONFIGURACIÓN Y EJECUCIÓN DE LA PLATAFORMA

Para ejecución de la plataforma, se propone la ejecución en un solo comando, esto significa que para ejecutar los servicios ya sea como servidor en desarrollo como en producción se lo haga con el mínimos de pasos posibles, debido que internamente los servidores están preconfigurados como se muestra a continuación.

2.2.6.1 Cliente

Para el caso de los clientes (SMA_APP y BSS) el proceso es simple, debido a que la librería “*create-react-app*” desarrollado por Facebook, nos brinda toda la configuración necesaria para compilar y generar los archivos estáticos para producción, por otro lado podríamos optar por configurar un empaquetador de JavaScript como Webpack, Parcel JS o browserify etc. y un transpilador como Babel, TypeScript o Kotlin, etc., en ambos casos únicamente debemos ejecutar el comando “*npm run build*” que nos generará la carpeta “*build*” de archivos estáticos, que como se mencionó anteriormente, solo es necesario asignar la carpeta “*build*” que contienen los archivos estáticos de cada uno de los servicios a un método de ruta del dominio o subdominio de un servidor web.

2.2.6.2 Servidor

Para el caso del servidor el proceso es relativamente más complicado, debido a que se tiene un conjunto de tecnologías que requieren ejecutarse de manera simultánea para poner en marcha el sistema, por esta razón es necesario implementar alguna metodología que permita ejecutar todas estas tecnologías con la menor cantidad de comandos posibles.

Dentro de las tecnologías que debemos conectar y configurar encontramos.

Node.js, es un entorno de ejecución de JavaScript construido con el motor de v8 de Chrome, dentro del aplicativo, esta tecnología es usada para ejecutar el programa de servidor, su instalación y configuración es muy sencilla, como se puede encontrar en su página Web oficial, únicamente se debe seguir los pasos indicados por el asistente de instalación en el caso de Windows y MacOS y un solo comando para el caso de Linux.

PostgreSQL, es una base de datos relacional Open Source, dentro del aplicativo cumple la función de ser la base de datos principal para el manejo de datos dentro del aplicativo, su instalación y configuración es bastante sencilla, en donde para Windows y MacOS se debe seguir los pasos indicados por el asistente de instalación, que en este caso el asistente es desarrollado por EDB POSTGRES, para el caso de Linux dependiendo de la distribución que se esté usando, se recomienda seguir una serie de pasos descritos en su página web, luego debe procederse a configurar las credenciales de acceso y crear la base de datos que se vaya a usar.

PostGIS, es un extensor de base de datos espacial para PostgreSQL, es decir brinda el soporte para ejecutar CRUD SQL a objetos geográficos, para el caso del aplicativo es usado para servir los datos geográficos para los mapas que se ejecutan en el cliente, su instalación y configuración depende del sistema operativo y el procedimiento es un tanto más complejo que los casos anteriores, para el caso de la configuración se debe agregar la extensión luego de haber creado la base de datos de PostgreSQL.

Redis es una base de datos estructurada en memoria Open Source, debido a su velocidad y optimización en almacenamiento de datos tipo hash, en el aplicativo es usada para almacenar las credenciales de sesión, su instalación y configuración son bastantes sencillas, consiste en ejecutar una serie de comando para los sistemas operativos de Linux y MacOS, pero Redis no es compatible con Windows por lo que si se desea ejecutar en este sistema operativo, existen dos opciones la primera consiste en instalar un versión antigua que es desarrollada por terceros para su funcionamiento en la sistema operativo de Windows y la segunda opción es instalar Redis en un subsistema Linux en Windows (WSL) y seguir los pasos de instalación para este sistema operativo.

Para que el aplicativo funcione como se plantea, debe estar funcionando todas estas tecnologías y otras instalaciones y ejecuciones que son requeridas como Puppeteer, instalación de dependencias del aplicativo, definición de estructuras de almacenamiento mediante lenguaje DDL, entre otros, para la propuesta de ejecución de un solo comando, esto es no es viable porque requiere una serie de configuraciones, además de no ser perfectamente compatible con todos los sistemas operativos, por ello se propone una solución haciendo uso de un contenedor de software, para esta propuesta se hace uso de Docker.

Docker

Es un proyecto de código abierto de la familia de los contenedores de software, desarrollado por Solomon Hykes, que permite la automatización del despliegue de aplicaciones dentro contenedores de software, una de las características más significativas de Docker es que usa características de aislamiento de recursos del kernel de Linux, para permitir que contenedores independientes se ejecuten dentro de una sola instancia de Linux, evitando la sobrecarga inicial y el mantener máquinas virtuales.

Un contenedor es una unidad estándar de software que empaqueta el código y sus dependencias para que la aplicación se ejecute cualquier entorno informático.

De esta forma Docker nos permite ejecutar todos los servicios en un contenedor, que contiene las imágenes del software que se describió con anterioridad, finalmente únicamente se requiere configurar Docker para que los servicios se instalen, configuren e inicien de manera automática mediante un solo comando. [32]

Configuración

La configuración de Docker que se propone para ejecutar los servicios similares a lo que se muestra en la Figura 2-17

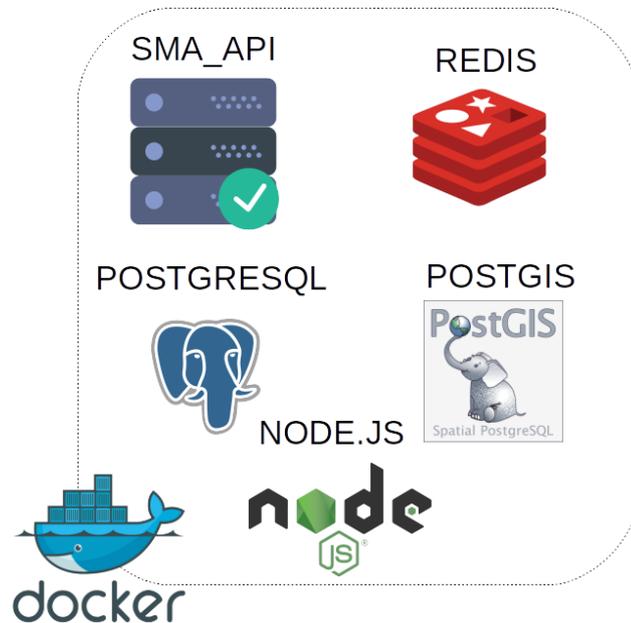


Figura 2-17 Agrupación de Servicios de SMA_API propuesta en contenedor de software Docker.

Fuente: autor

Para configurar Docker se inicia generando el archivo `docker-compose.yml`, el cual es el encargado de unir todas las imágenes de software y ejecutar los servicios, la configuración realizada se muestra a continuación.

```
#Versión de Docker
version: "3.7"

#Configuración de servicios de Docker
services:

# SMA_API

#Configura SMA_API, instalando las dependencias y Node.js para
funcionamiento.

sma_api:

container_name: sma_api
```

```
build: ./
volumes:
- ./usr/src/sma_api
- /usr/src/sma_api/node_modules
command: npm start
working_dir: /usr/src/sma_api
environment:
POSTGRES_URI:
postgres://usuario:contraseña@postgres:5432/db_nombre
REDIS_URI: redis://redis:6379
user: root
```

link es una característica heredada y puede ser eliminada por lo que debe ser usada solo en casos necesarios.

```
# links:
# - postgis
# - postgres
# - redis
ports:
- "3000:3000"
# Postgres
#Instala y configura PostgreSQL y PostGIS
postgres:
container_name: postgres
build: ./postgres
environment:
POSTGRES_USER: postgres
POSTGRES_PASSWORD: secret_password
POSTGRES_DB: sma_api
POSTGRES_HOST: postgres
ports:
```

```

- "5432:5432"

# Redis

#Instala y configura Redis

redis:

  container_name: redis

  image: redis

  ports:

    - "6379:6379"

```

En el archivo de configuración anterior se ve observa que se requiere de archivo Dockerfile para ejecutar y configurar SMA_API, esta configuración en encuentra a continuación.

```

FROM node:10.14.1

# Creamos un directorio para alojar la aplicación.

RUN mkdir -p /usr/src/sma_api

WORKDIR /usr/src/sma_api

#Configuración para que Puppeteer función en Docker

RUN apt-get update && apt-get install -yq libgconf-2-4

RUN apt-get update && apt-get install -y wget --no-install-recommends \

  && wget -q -O - https://dl-ssl.google.com/linux/linux_signing_key.pub | apt-key add - \

  && sh -c 'echo "deb [arch=amd64] http://dl.google.com/linux/chrome/deb/ stable main" >> /etc/apt/sources.list.d/google.list' \

  && apt-get update \

  && apt-get install -y google-chrome-unstable fonts-ipafont-gothic fonts-wqy-zenhei fonts-thai-tlwg fonts-kacst ttf-freefont \

  --no-install-recommends \

  && rm -rf /var/lib/apt/lists/* \

  && apt-get purge --auto-remove -y curl \

```

```

&& rm -rf /src/*.deb

ADD https://github.com/Yelp/dumb-
init/releases/download/v1.2.0/dumb-init\_1.2.0\_amd64
/usr/local/bin/dumb-init

RUN chmod +x /usr/local/bin/dumb-init

#Instalación de las dependencias del aplicativo

COPY package.json /usr/src/sma_api

RUN npm install

#Agrega usuarios para que no se necesite --no-sandbox.

RUN groupadd -r pptruser && useradd -r -g pptruser -G audio,video
pptruser \

&& mkdir -p /home/pptruser/Downloads \

&& chown -R pptruser:pptruser /home/pptruser \

&& chown -R pptruser:pptruser /usr/src/sma_api/node_modules

# Vuelve a ejecutarse los comandos como usuario no-privilegiado.

USER pptruser

ENTRYPOINT ["dumb-init", "--"]

CMD ["google-chrome-unstable"]

# Copiamos el contenido de la aplicación en el directorio que aloja la
aplicación

COPY . /usr/src/sma_api

# Construye los argumentos de la versión de Node.js

ARG NODE_VERSION=10.14.1

# Declaramos una variable de entorno para ejecutar Node.js

ENV NODE_VERSION $NODE_VERSION

```

En el archivo de configuración docker-compose.yml también se observa que se requiere de archivo Dockerfile para ejecutar y configurar los servicios PostgreSQL y PostGIS, esta configuración en encuentra a continuación.

```

FROM mdillon/postgis:latest

ADD /restoreDB//docker-entrypoint-initdb.d/restoreDB

ADD /restoreDB/rbcompleto /docker-entrypoint-initdb.d

```

```
COPY ./dbInit.sh ./
```

```
ADD /tables//docker-entrypoint-initdb.d/tables/
```

```
ADD deploy_schemas.sql /docker-entrypoint-initdb.d/
```

Una vez se agregan los archivos necesarios al contenedor de postgres, se requiere ejecutarlos y lo hace ejecutando internamente el archivo `deploy_schemas.sql` que se muestra a continuación, como consecuencia de ejecutar este archivo se ejecutan los archivos contienen la configuración DDL que permite estructurar la base de datos en PostgreSQL y además agregan la extensión PostGIS, estos últimos no se mostrarán debido a su considerable longitud.

```
\i '/docker-entrypoint-initdb.d/restoreDB/init.sql'
```

```
\i '/docker-entrypoint-initdb.d/tables/db_tables.sql'
```

Teniendo la estructura de la base de datos creada se procede escribir los datos iniciales que requiere el aplicativo para funcionar, este proceso debe ser realizado una sola vez, por esa razón debe ser ejecutada desde comandos externos, esto pese a que los archivos que contienen el CRUD a ejecutar ya se encuentran en el contenedor de postgres, el archivo que permite la ejecución es el `dbInit.sh` el cual se muestra a continuación, este ejecuta los archivos de CRUD que no se mostrarán debido a su considerable longitud.

```
cd docker-entrypoint-initdb.d/restoreDB
```

```
pg_restore -U postgres --dbname=sma_api --verbose --clean <  
rbcompleto
```

```
psql -U postgres -d sma_api -f '/docker-entrypoint-  
initdb.d/tables/db_write_tables.sql'
```

```
exit
```

Con la configuración anteriormente descrita, el administrador del servidor solo debe ejecutar un comando para instalar, configurar y ejecutar los servicios que la plataforma requiere, este proceso se describe a continuación.

Para ejecutar los servicios, se requiere tener instalado los servicios Docker, Docker Compose y Docker CE. Una vez el administrador tenga los servicios instalados debe dirigirse a la carpeta del proyecto y ejecutar el comando “*docker-compose up --build*” como se muestra en la Figura 2-18 el proceso completo puede

tardar unos 15 minutos aproximadamente dependiendo de la velocidad de la conexión a internet y características del computador, este tiempo de ejecución sólo será la primera vez: debido a la descarga de las imágenes de software caso contrario le tomará menos de 1 minuto para levantar todos los servicios, una vez termine el proceso debemos ver en la consola todos los servicios corriendo como se muestra en la Figura 2-19, pero si es la primera vez que ejecutamos el código debemos llenar las tablas como mencionamos anteriormente para ello ingresamos al contenedor de postgres mediante el comando “*docker exec -it postgres bash*”, una vez dentro ejecutamos el comando “*sh dbInit.sh*” como se muestra en la Figura 2-20 y esperamos que termine el proceso, si el proceso es correcto la consola debe verse como se muestra en la Figura 2-21 y salimos mediante el comando “*exit*”, con esto habremos configurado todo SMA API y está lista para brindar los servicios. Es importante mencionar que si no se quiere hacer uso de Docker se puede ingresar a la carpeta “bin” y ejecutar el comando “node www”, pero en este caso necesitamos instalar, configurar e iniciar todos los servicios manualmente y funcionará de manera transparente en ambos casos.

```

cmarcag@cmarcag-X555LAB: ~/Documentos/SMA_APP/sma_api_docker
Archivo Editar Ver Buscar Terminal Pestañas Ayuda
Terminal
x cmarcag@cmarcag-X555LAB: ~/Documentos/SMA_APP/sma_api_docker x
Total reclaimed space: 1.087GB
cmarcag@cmarcag-X555LAB:~/Documentos/SMA_APP/sma_api_docker$ clear
cmarcag@cmarcag-X555LAB:~/Documentos/SMA_APP/sma_api_docker$ sudo docker-compose up --build
Creating network "sma_api_docker_default" with the default driver
Building sma_api
Step 1/16 : FROM node:10.14.1
--> 75a384428e1d
Step 2/16 : RUN mkdir -p /usr/src/sma_api
--> Using cache
--> a50944f96b62
Step 3/16 : WORKDIR /usr/src/sma_api
--> Using cache
--> adf39c7f9abf
Step 4/16 : RUN apt-get update && apt-get install -y libgconf-2-4
--> Using cache
--> 9554d4422486
Step 5/16 : RUN apt-get update && apt-get install -y wget --no-install-recommends && wget -q -O - https://dl-ssl.google.com/linux/linux_signing_key.pub | apt-key add - && sh -c 'echo "deb [arch=amd64] http://dl.google.com/linux/chrome/deb/ stable main" >> /etc/apt/sources.list.d/google.list' && apt-get update && apt-get install -y google-chrome-unstable fonts-ipafont-gothic fonts-wqy-zenhei fonts-thai-tlwg fonts-kacst ttf-freefont --no-install-recommends && rm -rf /var/lib/apt/lists/* && apt-get purge --auto-remove -y curl && rm -rf /src/*.deb
--> Using cache
--> 7488feafd6a3
Step 6/16 : ADD https://github.com/Yelp/dumb-init/releases/download/v1.2.0/dumb-init_1.2.0_amd64 /usr/local/bin/dumb-init
--> Using cache
--> 120ddd3da004
Step 7/16 : RUN chmod +x /usr/local/bin/dumb-init
--> Using cache
--> 2d3860b05133
Step 8/16 : COPY package.json /usr/src/sma_api
--> Using cache
--> a2acf17cd858
Step 9/16 : RUN npm install
--> Using cache
--> 02c4d0d102d4
Step 10/16 : RUN groupadd -r pptruser && useradd -r -g pptruser -G audio,video pptruser && mkdir -p /home/pptruser/Downloads && chown -R pptruser:pptruser /home/pptruser && chown -R pptruser:pptruser /usr/src/sma_api/node_modules

```

Figura 2-18 Ejecución del contenedor de software Docker para iniciar los servicios de SMA API.

Fuente: autor

```

cmarcag@cmarcag-X555LAB: ~/Documentos/SMA_APP/sma_api_docker
Archivo Editar Ver Buscar Terminal Pestañas Ayuda
Terminal x cmarcag@cmarcag-X555LAB: ~/Documentos/SMA_APP/sma_api_docker x
postgres | CREATE EXTENSION
postgres | Loading PostGIS extensions into sma_api
postgres | NOTICE: extension "postgis" already exists, skipping
postgres | CREATE EXTENSION
postgres | CREATE EXTENSION
postgres | CREATE EXTENSION
postgres | /usr/local/bin/docker-entrypoint.sh: ignoring /docker-entrypoint-initdb.d/rbcompleto
postgres | /usr/local/bin/docker-entrypoint.sh: ignoring /docker-entrypoint-initdb.d/restoreDB
postgres | /usr/local/bin/docker-entrypoint.sh: ignoring /docker-entrypoint-initdb.d/tables
postgres | waiting for server to shut down...2019-01-11 19:15:57.419 UTC [43] LOG: received fast shutdown request
postgres | ..2019-01-11 19:15:57.527 UTC [43] LOG: aborting any active transactions
postgres | 2019-01-11 19:15:57.529 UTC [43] LOG: worker process: logical replication launcher (PID 50) exited with exit code 1
postgres | 2019-01-11 19:15:57.531 UTC [45] LOG: shutting down
postgres | ...2019-01-11 19:16:01.275 UTC [43] LOG: database system is shut down
postgres | done
postgres | server stopped
postgres | PostgreSQL init process complete; ready for start up.
postgres |
postgres | 2019-01-11 19:16:02.394 UTC [1] LOG: listening on IPv4 address "0.0.0.0", port 5432
postgres | 2019-01-11 19:16:02.395 UTC [1] LOG: listening on IPv6 address "::", port 5432
postgres | 2019-01-11 19:16:02.543 UTC [1] LOG: listening on Unix socket "/var/run/postgresql/.s.PGSQL.5432"
postgres | 2019-01-11 19:16:02.844 UTC [99] LOG: database system was shut down at 2019-01-11 19:16:01 UTC
postgres | 2019-01-11 19:16:03.335 UTC [1] LOG: database system is ready to accept connections
sma_api |
sma_api | > sma-api@0.0.0 start /usr/src/sma_api
sma_api | > nodemon ./bin/www
sma_api |
sma_api | [nodemon] 1.18.9
sma_api | [nodemon] to restart at any time, enter `rs`
sma_api | [nodemon] watching: *.*
sma_api | [nodemon] starting `node ./bin/www`

```

Figura 2-19 Compilación correcta de contenedor de software Docker.

Fuente: autor

```

cmarcag@cmarcag-X555LAB: ~/Documentos/SMA_APP/sma_api_docker
Archivo Editar Ver Buscar Terminal Pestañas Ayuda
cmarcag@cmarcag-X555LAB: ~/Documentos/SMA_APP/sma_api_docker x cmarcag@cmarcag-X555LAB: ~/Documentos/SMA_APP/sma_api_docker x
cmarcag@cmarcag-X555LAB:~/Documentos/SMA_APP/sma_api_docker$ sudo docker exec -lt postgres bash
[sudo] contraseña para cmarcag:
root@521781a47f6e:/# sh dbinit.sh
pg_restore: connecting to database for restore
pg_restore: dropping INDEX radiobases_gist
pg_restore: [archiver (db)] Error while PROCESSING TOC:
pg_restore: [archiver (db)] Error from TOC entry 4161; 1259 21057 INDEX radiobases_gist postgres
pg_restore: [archiver (db)] could not execute query: ERROR: index "radiobases_gist" does not exist
Command was: DROP INDEX public.radiobases_gist;

pg_restore: dropping CONSTRAINT radiobasesstatus radiobasesstatus_pkey
pg_restore: [archiver (db)] Error from TOC entry 4165; 2606 21056 CONSTRAINT radiobasesstatus radiobasesstatus_pkey postgres
pg_restore: [archiver (db)] could not execute query: ERROR: relation "public.radiobasesstatus" does not exist
Command was: ALTER TABLE ONLY public.radiobasesstatus DROP CONSTRAINT radiobasesstatus_pkey;

pg_restore: dropping CONSTRAINT radiobases radiobases_pkey
pg_restore: [archiver (db)] Error from TOC entry 4163; 2606 21054 CONSTRAINT radiobases radiobases_pkey postgres
pg_restore: [archiver (db)] could not execute query: ERROR: relation "public.radiobases" does not exist
Command was: ALTER TABLE ONLY public.radiobases DROP CONSTRAINT radiobases_pkey;

pg_restore: dropping SEQUENCE SET radiobasesstatus_no_seq
pg_restore: dropping SEQUENCE SET radiobasesstatus_id_seq
pg_restore: dropping SEQUENCE SET radiobasesstatus_cell_id_seq
pg_restore: dropping SEQUENCE SET radiobases_no_seq
pg_restore: dropping SEQUENCE SET radiobases_id_seq
pg_restore: dropping SEQUENCE SET radiobases_cell_id_seq
pg_restore: dropping TABLE DATA spatial_ref_sys
pg_restore: dropping TABLE DATA rbtodo
pg_restore: dropping TABLE DATA radiobasesstatus
pg_restore: dropping TABLE DATA radiobases
pg_restore: dropping DEFAULT rbtodo cell_id
pg_restore: [archiver (db)] Error from TOC entry 4168; 2604 21099 DEFAULT rbtodo cell_id postgres
pg_restore: [archiver (db)] could not execute query: ERROR: relation "public.rbtodo" does not exist
Command was: ALTER TABLE public.rbtodo ALTER COLUMN cell_id DROP DEFAULT;

pg_restore: dropping DEFAULT rbtodo no
pg_restore: [archiver (db)] Error from TOC entry 4159; 2604 21098 DEFAULT rbtodo no postgres
pg_restore: [archiver (db)] could not execute query: ERROR: relation "public.rbtodo" does not exist

```

Figura 2-20 Ingreso a la línea de comandos de la imagen de PostgreSQL.

Fuente: autor

```

cmarcag@cmarcag-X555LAB: ~/Documentos/SMA_APP/sma_api_docker
Archivo Editar Ver Buscar Terminal Pestañas Ayuda
cmarcag@cmarcag-X555LAB: ~/Documentos/SMA_APP/sma_api_docker x cmarcag@cmarcag-X555LAB: ~/Documentos/SMA_APP/sma_api_docker x
pg_restore: creating SEQUENCE OWNED BY "public.radiobases_no_seq"
pg_restore: creating TABLE "public.radiobasesstatus"
pg_restore: creating SEQUENCE "public.radiobasesstatus_cell_id_seq"
pg_restore: creating SEQUENCE OWNED BY "public.radiobasesstatus_cell_id_seq"
pg_restore: creating SEQUENCE "public.radiobasesstatus_id_seq"
pg_restore: creating SEQUENCE OWNED BY "public.radiobasesstatus_id_seq"
pg_restore: creating SEQUENCE "public.radiobasesstatus_no_seq"
pg_restore: creating SEQUENCE OWNED BY "public.radiobasesstatus_no_seq"
pg_restore: creating TABLE "public.rbtodo"
pg_restore: creating DEFAULT "public.radiobases id"
pg_restore: creating DEFAULT "public.radiobases no"
pg_restore: creating DEFAULT "public.radiobases cell_id"
pg_restore: creating DEFAULT "public.radiobasesstatus id"
pg_restore: creating DEFAULT "public.radiobasesstatus no"
pg_restore: creating DEFAULT "public.radiobasesstatus cell_id"
pg_restore: creating DEFAULT "public.rbtodo id"
pg_restore: creating DEFAULT "public.rbtodo no"
pg_restore: creating DEFAULT "public.rbtodo cell_id"
pg_restore: processing data for table "public.radiobases"
pg_restore: processing data for table "public.radiobasesstatus"
pg_restore: processing data for table "public.rbtodo"
pg_restore: processing data for table "public.spatial_ref_sys"
pg_restore: executing SEQUENCE SET radiobases_cell_id_seq
pg_restore: executing SEQUENCE SET radiobases_id_seq
pg_restore: executing SEQUENCE SET radiobases_no_seq
pg_restore: executing SEQUENCE SET radiobasesstatus_cell_id_seq
pg_restore: executing SEQUENCE SET radiobasesstatus_id_seq
pg_restore: executing SEQUENCE SET radiobasesstatus_no_seq
pg_restore: creating CONSTRAINT "public.radiobases radiobases_pkey"
pg_restore: creating CONSTRAINT "public.radiobasesstatus radiobasesstatus_pkey"
pg_restore: creating INDEX "public.radiobases_gist"
WARNING: errors ignored on restore: 25
BEGIN
INSERT 0 6
INSERT 0 4
INSERT 0 47080
COMMIT
root@521781a4f76e:/#

```

Figura 2-21 Acondicionamiento correcto de la base de datos.

Fuente: autor

2.2.7 PROCESO DE USO DE LA PLATAFORMA

La plataforma está diseñada para ser minimalista y de fácil acceso para los usuarios, el proceso a seguir para el manejo de interrupciones mediante la plataforma se esquematiza en la Figura 2-22, referirse a APÉNDICE C: DIAGRAMA DE PROCESOS PARA MANEJO DE APLICATIVO en el cual se muestra el diagrama extendido.

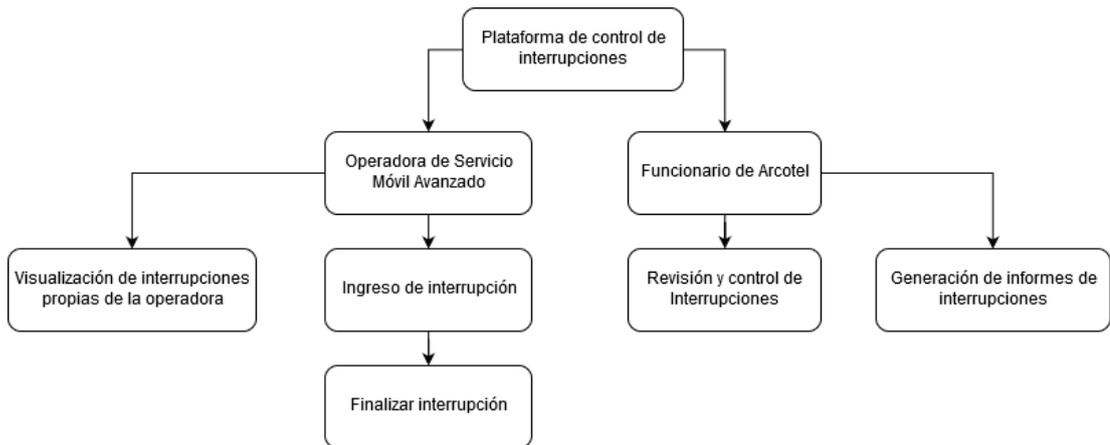


Figura 2-22 Esquema simplificado de funcionalidad de aplicativo.

Fuente: autor

2.3 DETERMINACIÓN DE MÉTODOS PARA ANÁLISIS ESTADÍSTICO

La función del análisis de la información de las interrupciones del Servicio Móvil Avanzado consiste en determinar alguna relación de las causas de dichas interrupciones con localidad, fechas, tiempos y tipos de interrupción, así como plantear algún modelo predictivo que permite inferir alguna posible interrupción que pueda darse a futuro.

Existen múltiples métodos de análisis como regresiones, correlaciones, análisis de varianza, pruebas de independencia o asociación y diferencias entre grupos, y según la naturaleza de la información presente se seleccionará el adecuado.

En este caso se cuenta con una historial de las interrupciones producidas en el transcurso del año 2016 hasta el 2018, se dispone información del tipo de interrupción, que puede ser programada o no programada, así como la causa, fecha, duración y localidad.

La cantidad de muestras es de 159 en total, con 69 programadas y 90 no programadas.

Tabla 2-4. Tipos de variables de información de interrupciones.

Variable	Tipo
Tipo	Dicotómica
Causa	Multinomial
Localidad	Multinomial
Duración	Continua
Fecha	Continua

Esta pequeña cantidad de muestras, así como los tipos de variables hace que la implementación de un análisis de asociatividad entre causa y localidad sea la opción más adecuada, ya que para un modelo predictivo mediante regresión se requiere que cumplan con varias características como en el caso de regresión lineal, en el cual se requieren propiedades de relación lineal entre las variables, las cuales no se cumplieron para los datos de las interrupciones.

Para el caso de pruebas de dependencia se requieren variables nominales, se cuenta con causa y localidad como principales variables a analizar, las dos son de tipo nominal con múltiples categorías.

Existen varios tipos de análisis de asociatividad entre dos variables, para el caso de variables categóricas se tiene el método de chi-cuadrado, y específicamente para dos variables multinomiales el método de chi-cuadrado de dependencia $R \times C$, por tanto, es el más adecuado debido a la naturaleza de la información que se presenta.

2.3.1 PROCEDIMIENTO DE ANÁLISIS

El primer paso para el desarrollo del análisis será la adecuación de la información con técnicas de codificación para un manejo más eficiente de la información, ya que el trabajar con datos de tipo textual resulta mucho más complejo que trabajar con variables numéricas, para esto se asignará un valor numérico a cada valor de tipo textual de las variables categóricas.

El siguiente paso consiste en desarrollar las estadísticas descriptivas, que son las básicas como promedios, histogramas, etc., para así poder tener una idea de las posibles relaciones que puedan existir y un panorama general de los datos.

Posteriormente se procederá con el análisis que se ajuste a las necesidades y naturaleza de los datos, tipos de variables, cantidad de muestras, pruebas de validez, etc., según el método seleccionado deberá contar con las validaciones necesarias para que el método sea el correcto, así también cada método tiene sus propias suposiciones, análisis y resultados.

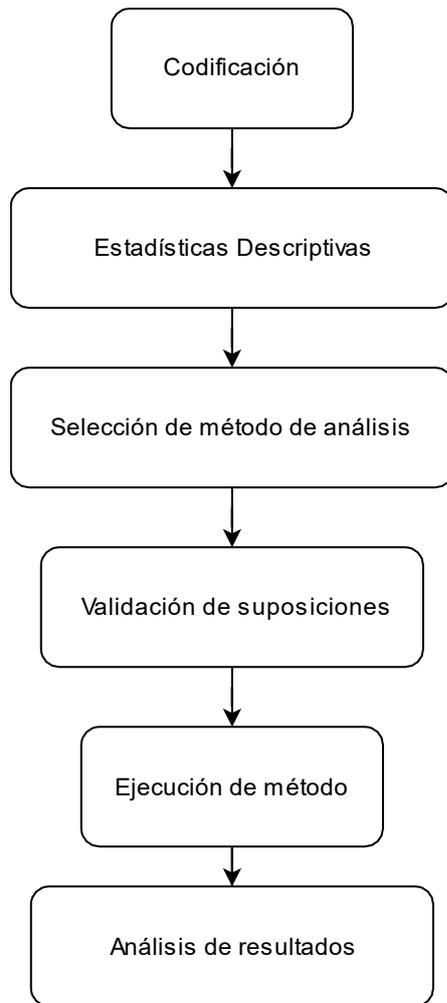


Figura 2-23. Diagrama de flujo para desarrollo de análisis.

Fuente: autor

CAPÍTULO 3: IMPLEMENTACIÓN Y ANÁLISIS DE RESULTADOS

3.1 DESCRIPCIÓN DE LA PLATAFORMA Y COMPONENTES DEL SISTEMA DE MANEJO DE INTERRUPCIONES DEL SMA

El presente proyecto se compone de aplicativos Web que tienen como objetivo implementar un sistema de manejo de interrupciones del Servicio Móvil Avanzado y un sistema de visualización de estado de estaciones base, para lo cual se construyó componentes que se describen a lo largo del capítulo.

La implementación global del proyecto de manera general presenta las siguientes características:

- Visualización de información de estructuras de radiobases en mapa interactivo.
- Búsqueda y ubicación de estructuras radiobases mediante filtros y cuadros de sugerencia intuitivos y simplistas.
- Funcionamiento Offline para búsqueda y ubicación de estructuras radiobases.
- Sistema online de registro de usuarios.
- Visualizadores de interrupciones (tabulado dinámico, tablero dinámico).
- Registro de interrupciones.
- Notificaciones mediante correo electrónico.
- Visualizador de proceso de interrupción.
- Generador de informe con editor de texto simple.
- Sistema de comentarios por interrupción, mediante sistema de mensajería en tiempo real.

En la Figura 3-1 se muestra el proceso general de la plataforma, el cual será descrito a lo largo del capítulo.

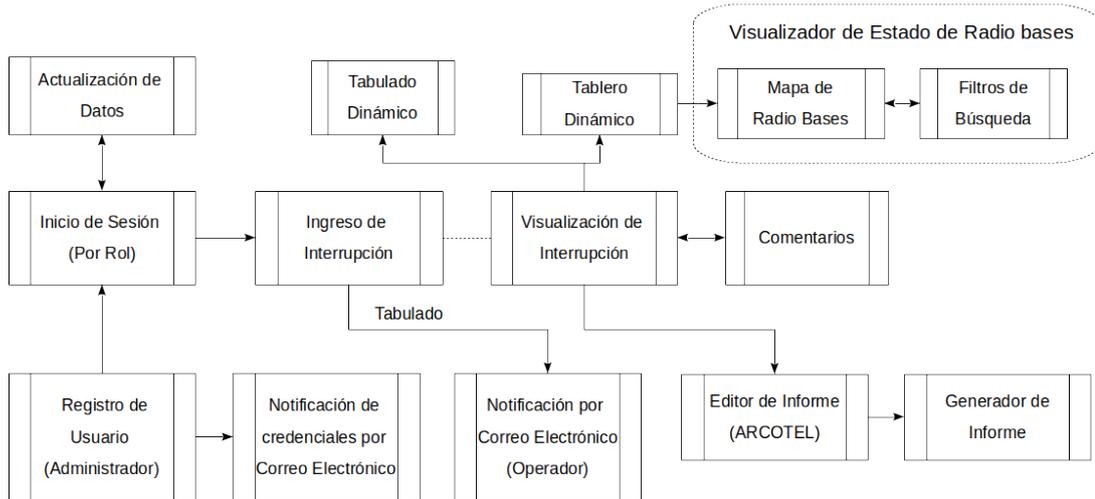


Figura 3-1 Esquema de procesos de la plataforma.

Fuente: autor

3.2 COMPONENTES GENERALES DE LA PLATAFORMA

3.2.1 REGISTRO E INICIO DE SESIÓN DE USUARIO

3.2.1.1 Registro

El registro de un nuevo usuario a la plataforma será a través del usuario administrador el cual es registrado automáticamente en la primera ejecución del servidor, el registro se realizará únicamente con el correo electrónico del usuario a ser registrado, esta metodología se propone debido a que el acceso a la plataforma será restringido a usuarios definidos por la agencia.

El proceso inicia cuando el usuario brinda el correo electrónico al administrador de la plataforma, seguido el administrador accede a la plataforma y registra al usuario otorgando un “rol” como se muestra en la Figura 3-2. Una vez se envía la solicitud de registro, el servidor procede a generar las credenciales del usuario y las almacena en la base de datos, seguido se envían las credenciales de correo y contraseña (la contraseña generada es aleatoria) al usuario como se muestra en la Figura 3-3. Una vez recibido el correo con las credenciales se recomienda al usuario cambiar las credenciales por defecto.

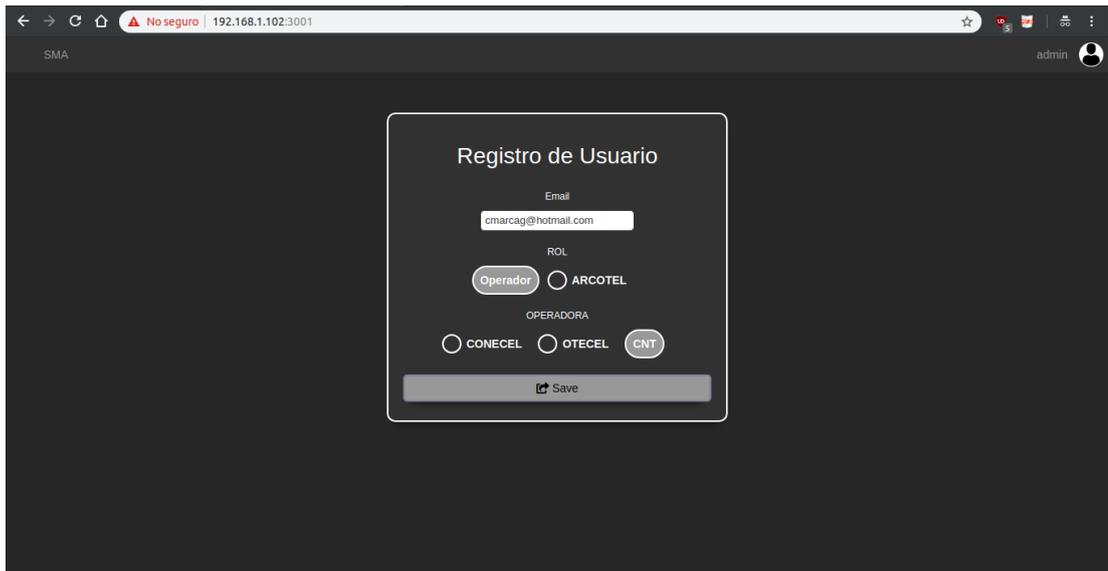


Figura 3-2 Página de administrador para registro de usuarios.

Fuente: autor



Figura 3-3 Correo electrónico entregado al usuario luego del registro.

Fuente: autor

3.2.1.2 Inicio de Sesión

El proceso de inicio de sesión consiste en ingresar a la landing page de la plataforma mostrada en la Figura 3-4, e ingresar las credenciales correo y contraseña entregados al usuario luego de realizado el registro mencionado anteriormente, es importante mencionar que los usuarios pueden cambiar su contraseña y otros datos del usuario en la página como se muestra en la Figura 3-5.

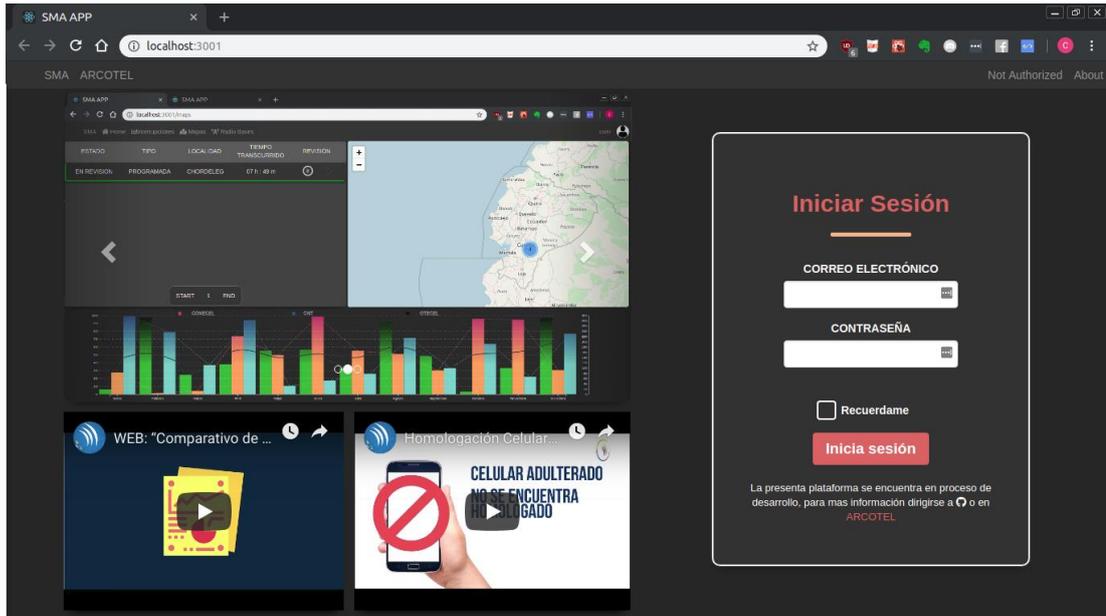


Figura 3-4 Landing page (Página de Inicio) de la plataforma.

Fuente: autor

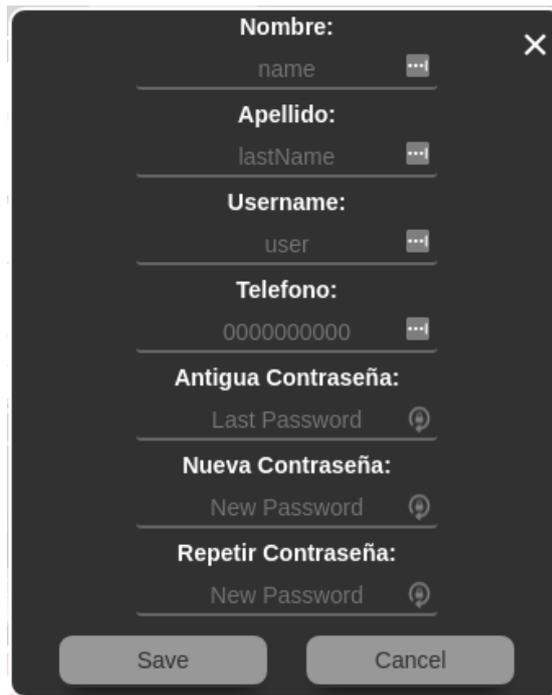
A dark-themed modal window for updating user data. It contains several input fields with labels in Spanish: 'Nombre:' (name), 'Apellido:' (lastName), 'Username:', 'Telefono:' (0000000000), 'Antigua Contraseña:' (Last Password), 'Nueva Contraseña:' (New Password), and 'Repetir Contraseña:' (New Password). Each field has a small icon to its right. At the bottom, there are two buttons: 'Save' and 'Cancel'. A close button (X) is in the top right corner.

Figura 3-5 Modal de actualización de datos de usuario.

Fuente: autor

3.2.2 INGRESO DE INTERRUPCIÓN

El proceso de registrar una interrupción actualmente se lo realiza de manera escrita mediante un correo electrónico, en donde se describe las características generales de la interrupción, cada operadora envía en un formato propio este registro, este procedimiento genera un cuello de botella al analizar la interrupción para el registro de esta.

Debido a ello se propone una aplicación de registro intuitiva y de fácil acceso a los componentes de una interrupción como se muestra en la Figura 3-6.

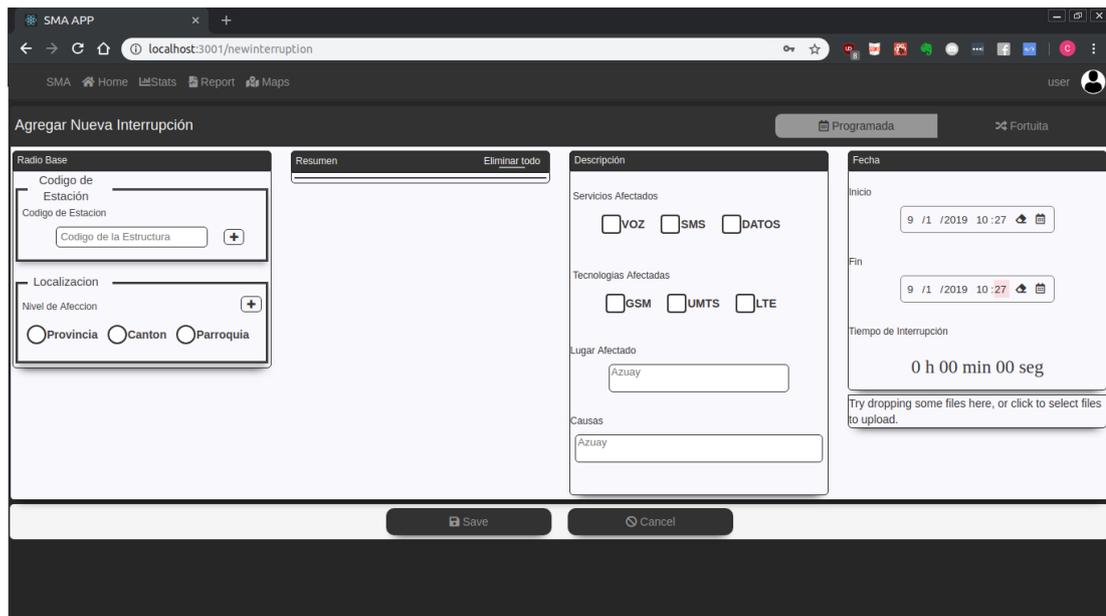


Figura 3-6 Aplicación de registro de interrupción.

Fuente: autor

Este aplicativo está desarrollado de tal forma que pueda brindar las de facilidades al operador para registrar una interrupción, dentro de las herramientas para cumplir este objetivo podemos encontrar, un campo que calcula el tiempo aproximado de la interrupción (En este campo se consideran todos los días como días hábiles para la interrupción, los datos de los parámetros de fecha (ingresado a través de un modal de calendario), un conjunto de cuadros de sugerencia generados mediante filtros de datos, que actúan de acuerdo a la manipulación del usuario para registrar la localidad de afección, de manera similar el registro de una sola estructura en caso especiales se lo realiza mediante un cuadro de sugerencias, además se presenta un cuadro de resumen con las estructuras afectas que permite una sencilla adición y eliminación de estructuras a registrar. Es importante mencionar que la metodología de selección por cuadro de sugerencias no está únicamente implementado para facilidad del operador sino que también contienen métodos implementados de validación de datos que permiten al servidor validar y registrar una interrupción sólo si el operador corresponde a las estructuras que pretende registrar, ya que el estado de la aplicación contiene los parámetros de autenticación necesarios para completar el proceso, en la Figura 3-7 se muestra el formulario de interrupción llenado .

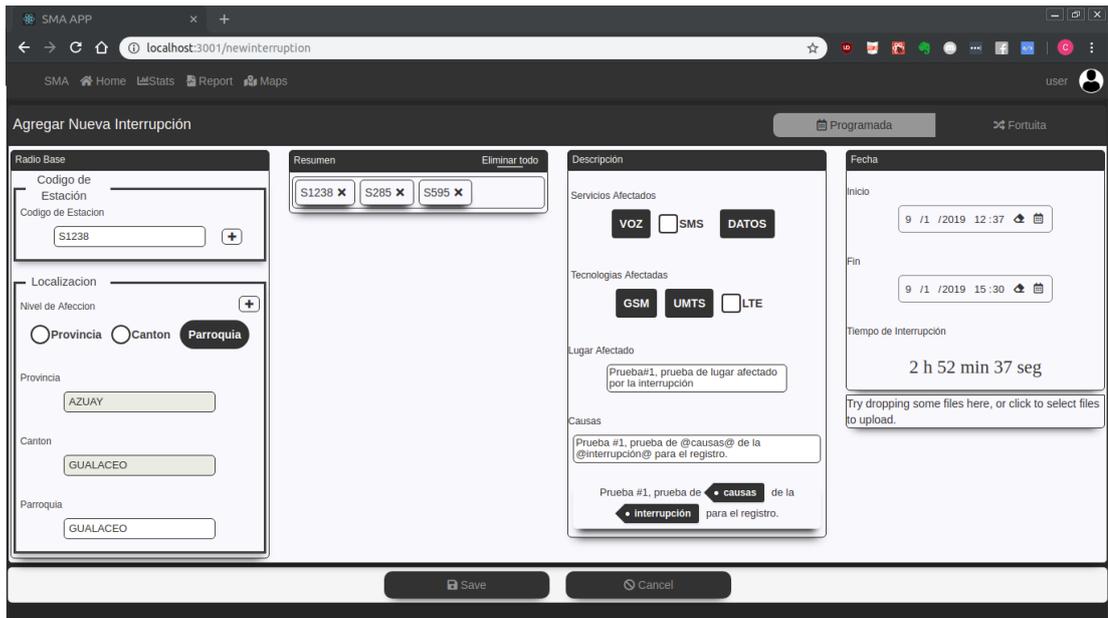


Figura 3-7 Campos del registro de interrupción completados.

Fuente: autor

Una vez completamos todos los campos se procede a registrar la interrupción, antes de completar el proceso se muestra un modal que contiene un resumen de los datos ingresados de la interrupción para verificación, además requiere ingresar uno o varios correos electrónico de remitentes a los que se enviará una copia del registro de la interrupción, como se muestra en la Figura 3-8, habiendo completado el proceso se registra la interrupción y se envía una copia a los remitentes ingresados como se muestra en la Figura 3-9.

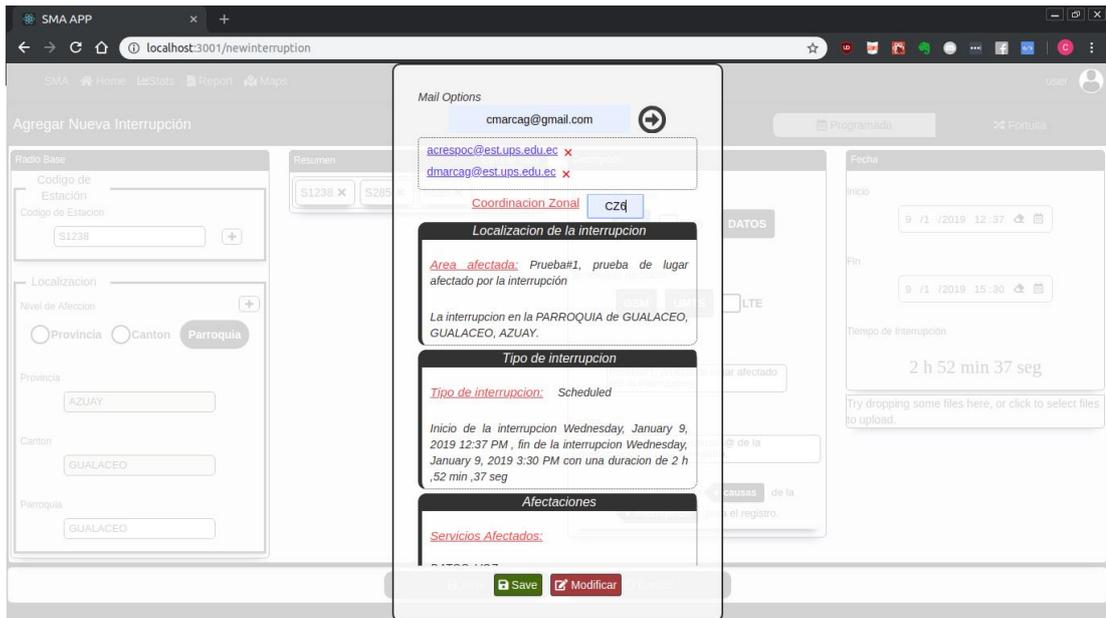


Figura 3-8 Modal de resumen de interrupción e ingreso de destinatarios de correo electrónico.

Fuente: autor

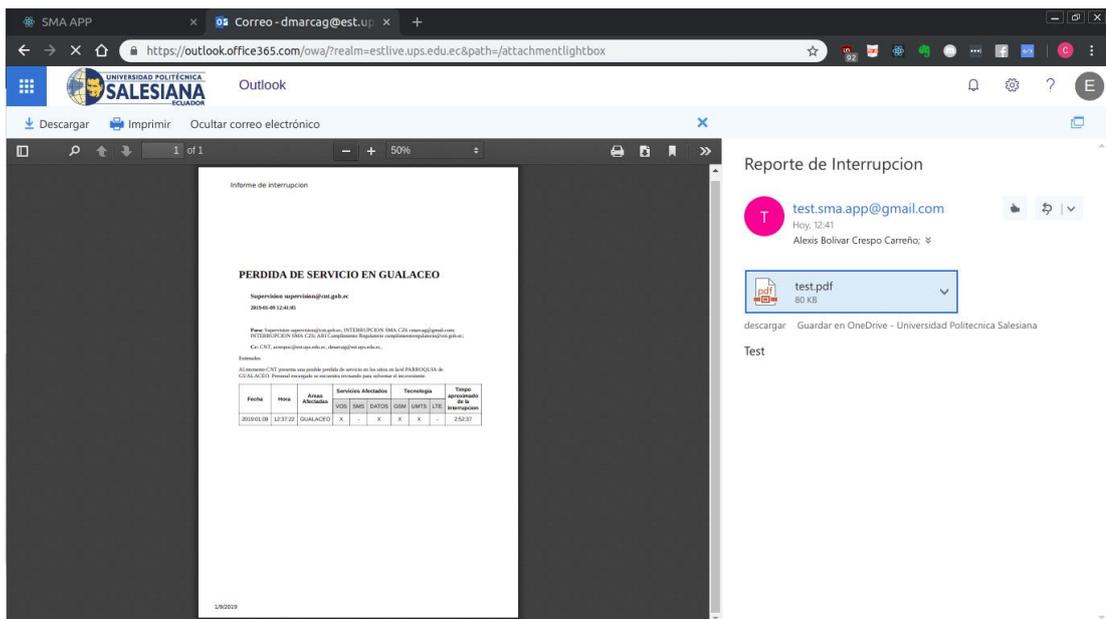


Figura 3-9 Correo Electrónico de registro de interrupción recibido.

Fuente: autor

3.2.3 VISUALIZACIÓN DE INTERRUPCIONES

Como se mencionó anteriormente el seguimiento de la interrupción se realiza a través un registro de correo electrónico, los cuales, al momento de hacer un seguimiento a las interrupciones, genera un proceso laborioso debido a la dificultad de

verificar el estado de cada uno de ellos, para solventar el problema se plantea dos aplicaciones para la visualización de las interrupciones las cuales son la tabulación dinámica y el tablero dinámico.

3.2.3.1 Tabulación Dinámica

Este aplicativo consiste en sistema de tabulación de datos como se muestra en la Figura 3-10 el cual contiene herramientas de filtrado y búsqueda y un mecanismo de etiquetado de interrupciones.

Id inte	Tipo	Canton inte	Fecha inicio	Localidad	Tiempo Transcurrido	Revision
1	PROGRAMADA	GUALACEO	2019-01-09T17:37:22.431+00:00	GUALACEO	55 m	Ir >
2	PROGRAMADA	CHORDELEG	2019-01-03T17:37:22.431+00:00	CHORDELEG	06 d : 00 h : 55 m	Ir >
3	PROGRAMADA	PORTOVIEJO	2018-12-31T17:37:22.431+00:00	PORTOVIEJO	09 d : 00 h : 55 m	Ir >>
4	PROGRAMADA	PAUTE	2018-12-26T17:37:22.431+00:00	PAUTE	14 d : 00 h : 55 m	Ir >>>
5	PROGRAMADA	SIGSIG	2018-12-12T17:37:22.431+00:00	SIGSIG	28 d : 00 h : 55 m	Ir >>>
6	PROGRAMADA	ZAMORA	2019-01-07T17:37:22.431+00:00	ZAMORA	02 d : 00 h : 55 m	Ir >>>

Figura 3-10 Aplicativo de tabulación dinámica para visualización de interrupciones.

Fuente: autor

Herramientas de Filtrado y Búsqueda

Las herramientas buscan facilitar la ubicación dentro de la tabla de interrupciones registradas, en donde podemos encontrar las herramientas de, ordenar en forma ascendente y descendente los campos de las interrupciones, filtros de búsquedas de interrupciones por fechas, filtro de búsqueda mediante entrada de texto para las localidades y funciones de agregar y eliminar campos a la tabla dependiendo de las necesidades del usuario, estas herramientas están desarrolladas tanto en el servidor como en cliente, dependiendo de las necesidades de procesamiento y uso de datos para ser ejecutado con el fin de garantizar una fluidez en la búsqueda.

Etiquetado de Interrupciones

El mecanismo de etiquetado se basa en un proceso simple, las etiquetas son establecidas de acuerdo al tiempo transcurrido desde el inicio de la interrupción, el cálculo de este tiempo toma en consideración únicamente los días laborales mediante un algoritmo simple, el tiempo transcurrido y las etiquetas cumplen con los tiempos establecidos en [24], el etiquetado es netamente visual en donde podemos encontrar tres colores base de semaforización, los cuales con “INICIO” color “VERDE”, “INTERMEDIO” color “NARANJA” y “FINALIZADO” color “ROJO”, para diferenciar entre el estado del proceso del operador o de la agencia se diferencian mediante una etiqueta “A” como se muestra en la Figura 3-10.

3.2.3.2 Tablero Dinámico

La metodología anteriormente presentada para visualización de interrupciones es simplista, pero puede ser poco intuitiva en el aspecto visual del conjunto de interrupciones, por lo cual se propone una aplicación para visualizar las estructuras que son afectadas por las interrupciones.

El aplicativo mostrado en la Figura 3-11, en donde se puede encontrar una metodología de tabulado similar al presentado en el apartado anterior, pero este no posee las herramientas de filtrado y búsqueda, esto debido al tamaño dispuesto para el componente en el actual aplicativo, además se puede encontrar un componente de mapas que permite ubicar geográficamente las interrupciones registradas, finalmente un componente de gráficos dinámicos, la finalidad de este componente es visualizar las estadísticas de cantidad de interrupciones reportadas a lo largo del año en curso, estos componentes están pensados para que el usuario tenga una visión más completa de las interrupciones que se registran.



Figura 3-11 Aplicativo de tablero dinámico para visualización de interrupciones.

Fuente: autor

3.2.3.3 Visualizador y Administrador de Interrupción

La administración de la interrupción se la hace mediante el seguimiento de la misma a través de informes que son entregados por parte del operador a través del sistema de Quipux el cual es un servicio web para gestión documental dispuesto por el Ministerio de Telecomunicaciones y de la sociedad de la Información y su uso es exclusivo para las instituciones públicas de la Función Ejecutiva [<https://www.gestiondocumental.gob.ec/>], por lo que el acceso al mismo por parte de los autores es nulo, pero el sistema planteado presenta las características necesarias para la integración a futuro.

Como alternativa para la administración de las interrupciones se presenta una aplicación de visualización y administración de interrupción, el cual consiste en un visualizador de las características de la interrupción con temporizadores en tiempo real, un editor de texto simple de tipo WYSIWYG HTML de alta capacidad de escalabilidad, un editor de HTML para edición de texto complementaria y un sistema de comentarios en tiempo real.

El visualizador de características de la interrupción es un componente de la aplicación que se encarga de extraer la información de la interrupción a una tasa de refresco de 1 segundo.

El editor de texto simple de tipo WYSIWYG HTML, es un editor de texto que posee características básicas para la edición de textos, permite editar el informe de manera online, el cual se genera automáticamente al momento de realizar la primera visualización de la interrupción con los datos bases de la misma, los cambios hechos por el usuario se van registrando conforme este lo desee, el editor HTML está conectado a este editor, permitiendo realizar una edición de bajo nivel, brindando todas las capacidades de un documento HTML, este tipo de implementación permite un alto grado de escalabilidad para desarrollar editores de texto avanzados. Existen muchas librerías que cumplen esta labor en donde podemos encontrar a react-quill, Draft.js, Slate.js entre otros, en el presente proyecto no se implementó ninguna de estas librerías debido a que el editor se presenta como una propuesta, y puede ser escalado o reemplazado por dichas librerías sin mayor complejidad o modificación del código fuente.

El sistema de comentarios es un sistema de mensajería simple en tiempo real, basado en salas de mensajería, lo que significa que múltiples usuarios que poseen las credenciales de accesos a una interrupción registrada pueden acceder a la sala de comentarios correspondientes, cuando el usuario abandone la sala el servidor detecta este cambio y elimina las credenciales de acceso del usuario a la sala hasta que este vuelva a conectarse a la sala mediante las credenciales de la interrupción.

Este aplicativo debido a sus componentes dinámicos, fue implementado de tal manera que brinde flexibilidad de movimiento y redimensionamiento de los componentes a los largo y ancho de la pantalla, además se implementó menús contextuales con el fin de minimizar el impacto visual en el aplicativo, estos menús, permiten realizar funciones como, volver a el estado original del informe, cambiar el estado de estructuras de radiobases, generar el informe y reenviar al correo electrónico, entre otros.

Es importante mencionar que el editor de texto y generador de informes es accesible para aquellos usuarios que tengan las credenciales de agente de la agencia, por lo tanto, un operador no puede tener acceso a estos datos, por esa razón para el operador se procedió a habilitar el servicio de mapa para la visualización como se muestra en la Figura 3-12 y en la Figura 3-13 se muestra componentes antes descritos para la agencia.

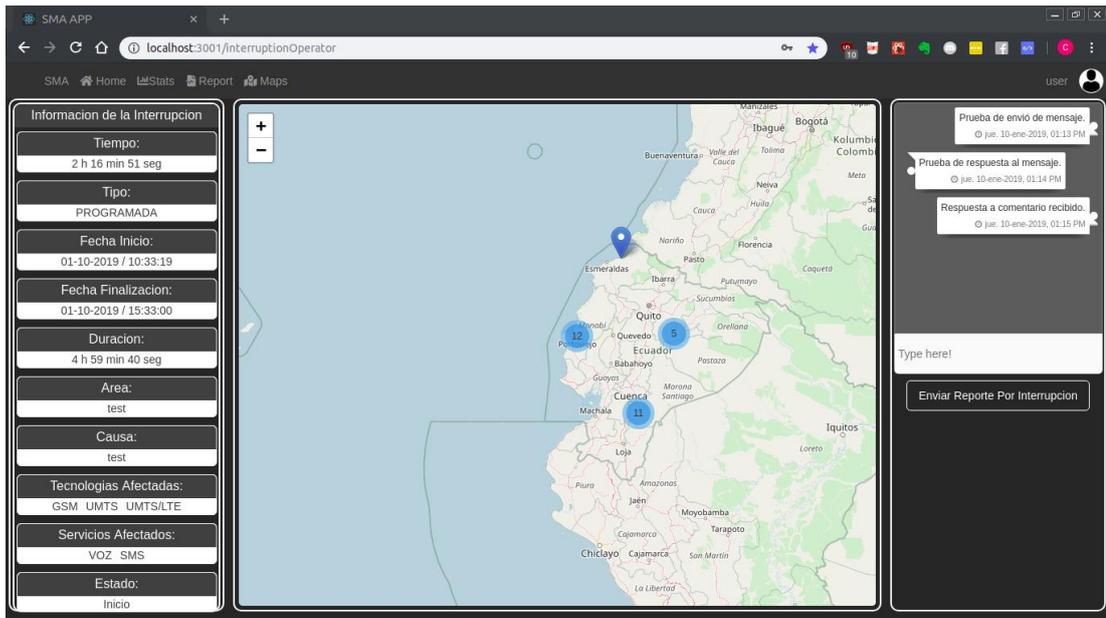


Figura 3-12 Visualizador de interrupción del operador.

Fuente: autor

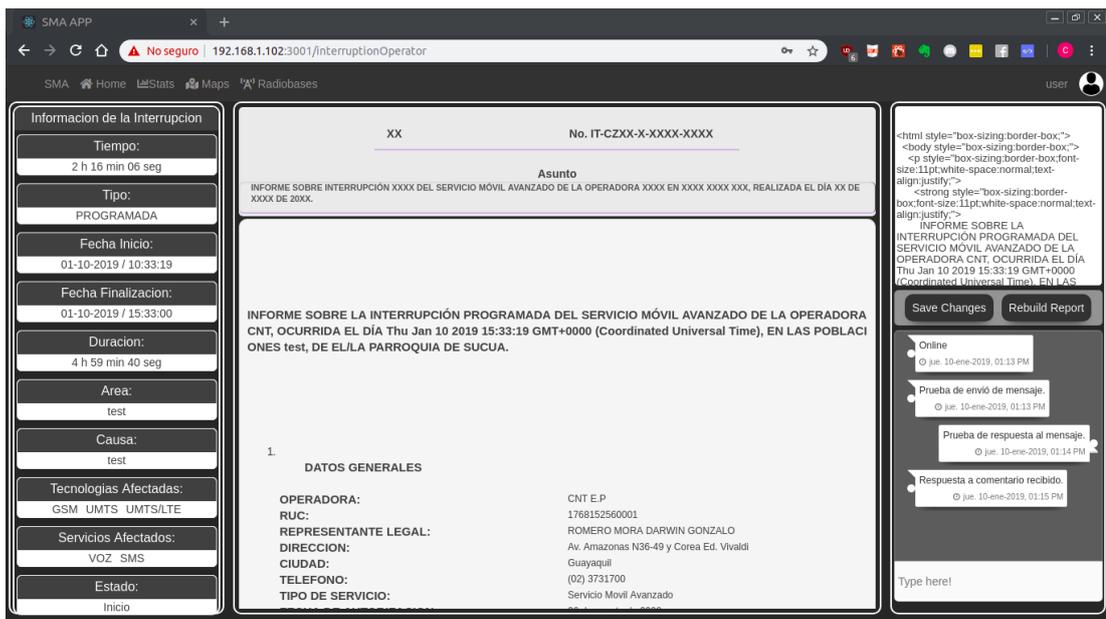


Figura 3-13 Visualizador y administrador de interrupción de funcionario.

Fuente: autor

3.2.3.4 Visualizador de estado de radiobases (BSS)

Es un aplicativo que permite visualizar las radiobases de las operadoras del servicio móvil avanzado en el territorio ecuatoriano y se está compuesto por las siguientes funcionalidades.

Mapa

El mapa tiene la función de exponer visualmente la localización de las radiobases de brindan el servicio de telefonía móvil avanzada, las radiobases son presentadas como un conjunto de puntos dentro del mapa, es decir un punto no representa una estructura física, sino que expresa características de una radiobase como un sector y/o una tecnología que pertenece a la estructura, por esta razón la cantidad de puntos que podemos encontrar es bastante elevada.

La cantidad de puntos que existen en el mapa generan una contaminación visual, por ello se procedió a usar y configurar un plugin desarrollado para Leaflet, que mediante algunos cálculos de densidad de puntos y nivel de zoom realiza un clúster de los puntos, generando un aspecto visual más amigable para el usuario, como se muestra en la Figura 3-14

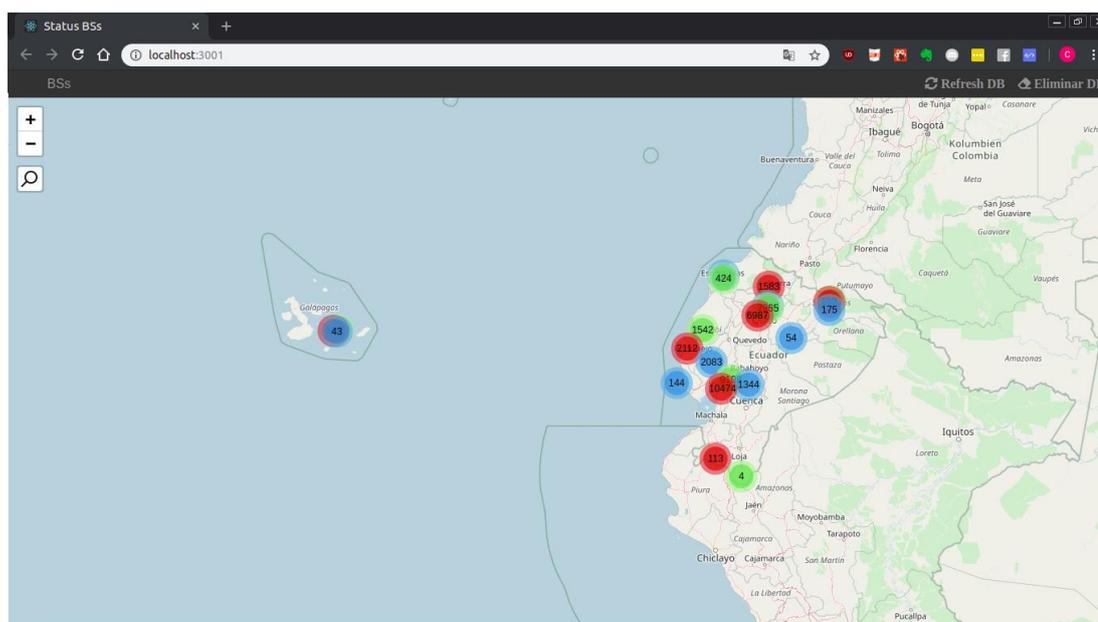


Figura 3-14 Visualizador de radiobases.

Fuente: autor

Para ver la información de la radiobase correspondiente a un punto en el mapa, solo es necesario dar clic al punto que deseemos ver su información, y esta es expuesta en una barra lateral, como se presenta en la Figura 3-15.

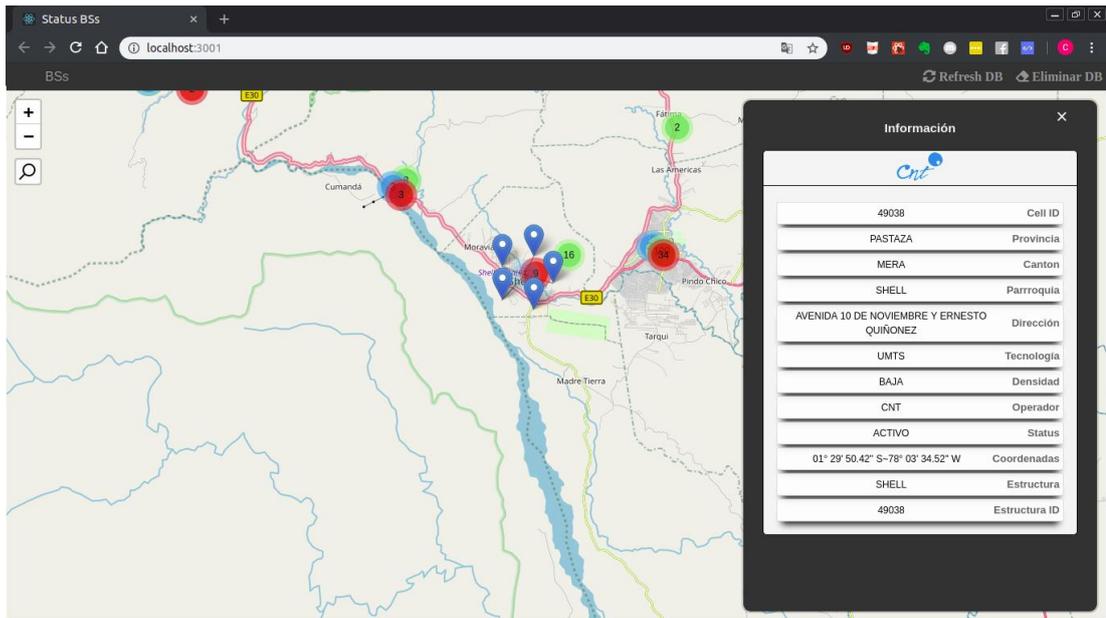


Figura 3-15 Modal de información de radiobases.

Fuente: autor

Filtros

Una de las principales características del aplicativo es el sistema de filtrado implementado, el cual consiste en un conjunto de entradas de texto que se encuentran conectados internamente, permitiendo, mediante un cuadro de sugerencias seleccionar la radiobases que se desea buscar, la búsqueda es realizada en la base de datos y genera las sugerencias con una estructura que contiene los campos nombre de estación, Identificador de la celda, dirección de la estructura y parroquia, estas funciones se generan en base a lo que el usuario escriba una de las entradas como se muestra en la Figura 3-16.

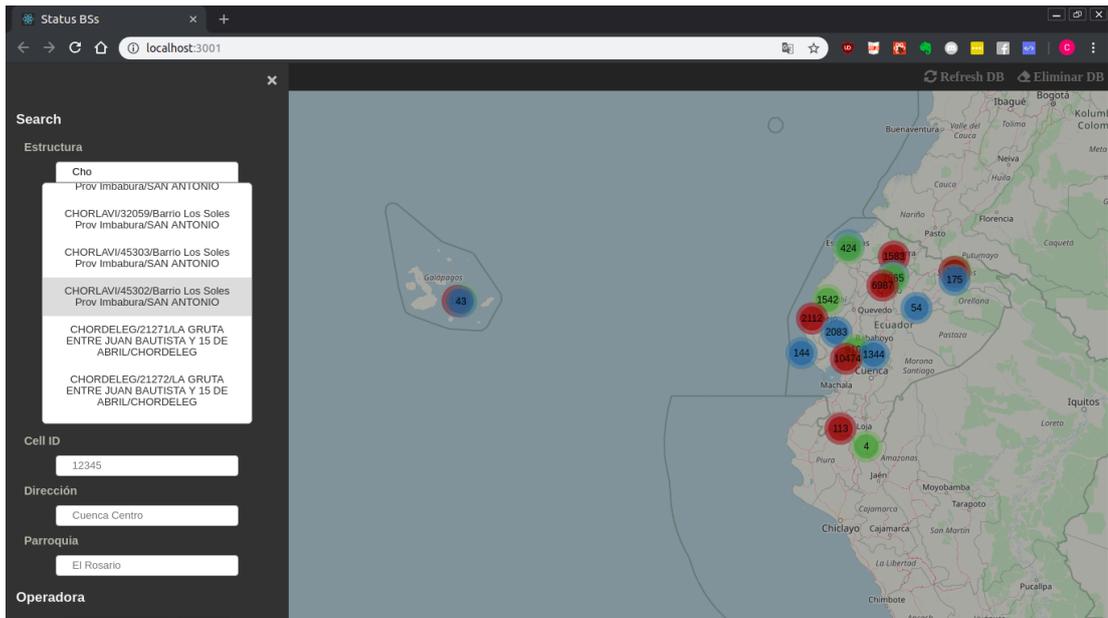


Figura 3-16 Cuadro de sugerencias para ubicación de estructuras.

Fuente: autor

Dentro del sistema de filtros encontramos la botonera, que nos permite optimizar los resultados de la búsqueda mediante la reducción de tamaño del objeto de sugerencias, lo cual optimiza la búsqueda de las entradas de sugerencia presentado anteriormente como se muestra en la Figura 3-17.

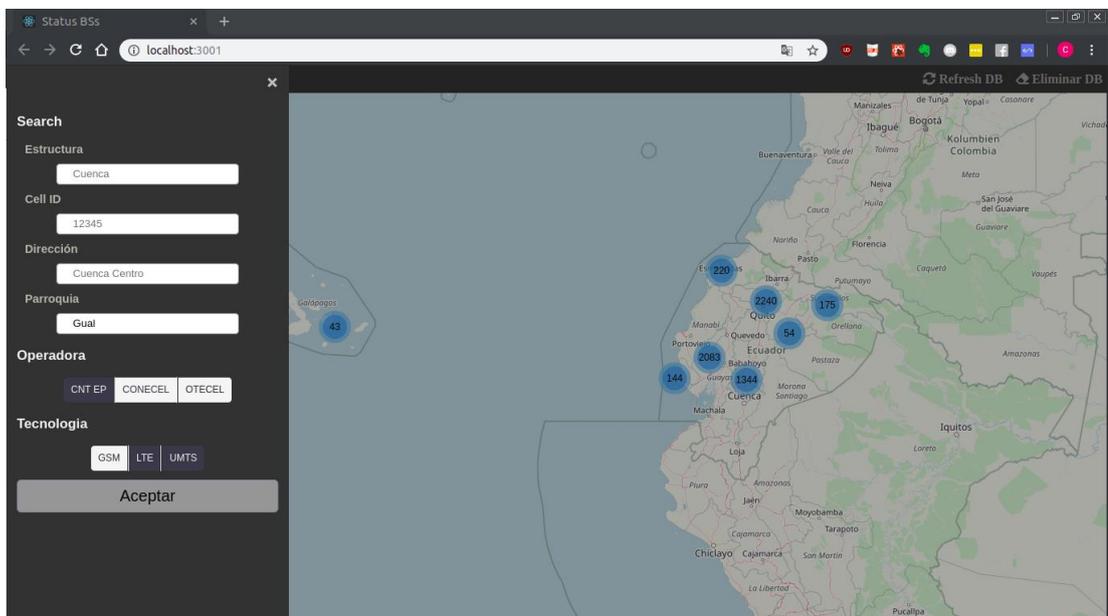


Figura 3-17 Herramientas de filtrado para aplicativo de mapas.

Fuente: autor

CAPÍTULO 4: ANÁLISIS ESTADÍSTICO DE INTERRUPCIONES DEL SERVICIO MÓVIL AVANZADO

Para determinar alguna relación entre los factores que componen las interrupciones se realizará un análisis de la información, este análisis permitirá determinar si existe una dependencia entre los variables implicadas en la información de las interrupciones del servicio móvil.

DESCRIPCIÓN DE LA PLATAFORMA Y COMPONENTES DEL SISTEMA DE MANEJO DE INTERRUPCIONES DEL SMA

El presente proyecto se compone de aplicativos Web que tienen como objetivo implementar un sistema de manejo de interrupciones del Servicio Móvil Avanzado y un sistema de visualización de estado de estaciones base, para lo cual se construyó componentes que se describen a lo largo del capítulo.

DESCRIPCIÓN DE LA INFORMACIÓN

Los datos de las interrupciones son de los años 2016, 2017 y 2018, pertenecientes a la coordinación zonal 6 de la ARCOTEL, debido a que la ARCOTEL no tiene obligación de llevar un historial, solo se tuvo acceso a información que está limitada a la facilitada por dicha coordinación.

La coordinación zonal 6 de la ARCOTEL está a cargo de las provincias de Loja, Cañar, Azuay, El oro, Zamora Chinchipe y Morona Santiago, por tanto, regiones que se encuentran en las tres regiones principales del país.

A continuación, se presenta una tabla de la cantidad de radiobases operativas en todo el país y en cada provincia pertenecientes a la coordinación zonal 6.

Tabla 4-1. Cantidad de radiobases a nivel de coordinación zonal 6.

Provincia	Cantidad de radiobases
Azuay	2101
Cañar	465
Loja	1037
El oro	1527
Morona Santiago	187
Zamora Chinchipe	150
Total nacional	47080

Lo que representa un 11.6% del total de todas las radiobases del país, dado que cada región, provincia y ciudad tiene una realidad distinta debido a muchos factores como son la población, geografía, tipo de clima, etc. el resultado que se obtenga será representativo únicamente para dichas zonas.

Las variables presentes en las muestras son las siguientes:

Tabla 4-2. Variables y tipos.

Variable	Tipo
Tipo	Dicotómica
Causa	Multinomial
Localidad	Multinomial
Duración	Continua
Fecha	Continua

|CODIFICACIÓN DE VARIABLES

Se codificará la información de las variables categóricas debido a que la complejidad de manera valores de tipo carácter es mayor e ineficiente que manejar valores numéricos asignados a cada grupo o categoría de las variables.

La primera variable corresponde al tipo de interrupción la cual se asignará los valores según la siguiente tabla.

Tabla 4-3. Codificación de variable Tipo.

Tipo	Valor
No programada	0
Programada	1

Como se indica en la normativa una interrupción no programada se produce cuando una estación base o un conjunto deja de proveer el servicio de llamadas, mensajes o datos, mientras que una programada se produce por una programación de esta para labores de mantenimiento o reparación según sea el caso.

Tabla 4-4. Codificación de variable causa.

Causa de interrupción de servicio de comunicaciones móviles	Valor
Intermitencia	0
Caída de enlace microonda	1
Caída de torre	2
Falla del sistema eléctrico	3
Corte de energía eléctrica	4
Corte de fibra óptica	5
Falla de equipos de comunicaciones	6
Falla del sistema de enfriamiento	7
Mantenimiento programado	8
Cambio de Hardware	9
Implementación de nuevo enlace	10

Las causas listadas en la tabla corresponden tanto a interrupciones programadas y no programadas, las programadas corresponden a los números 8, 9 y 10, el resto corresponde a causas de interrupciones no programadas.

Por otro lado, la variable región se derivó de la necesidad de agrupar las distintas zonas donde se producen las interrupciones para poder realizar el análisis, en este caso las ciudades, parroquias o provincias se asignaron a cada región correspondiente, Sierra, Costa, Amazonia o varias cuando existan más de una región afectada por la interrupción.

Tabla 4-5 Codificación de variable región.

Región	Valor
Costa	0
Sierra	1
Amazonia	2
Varias	3

ESTADÍSTICAS DESCRIPTIVAS

Se procederá a realizar el análisis de la cantidad de interrupciones programadas como no programadas, así como promedios de duración y en qué mes se producen

mayormente para poder tener una visión general de la distribución de las distintas causas según región en la cual se producen.

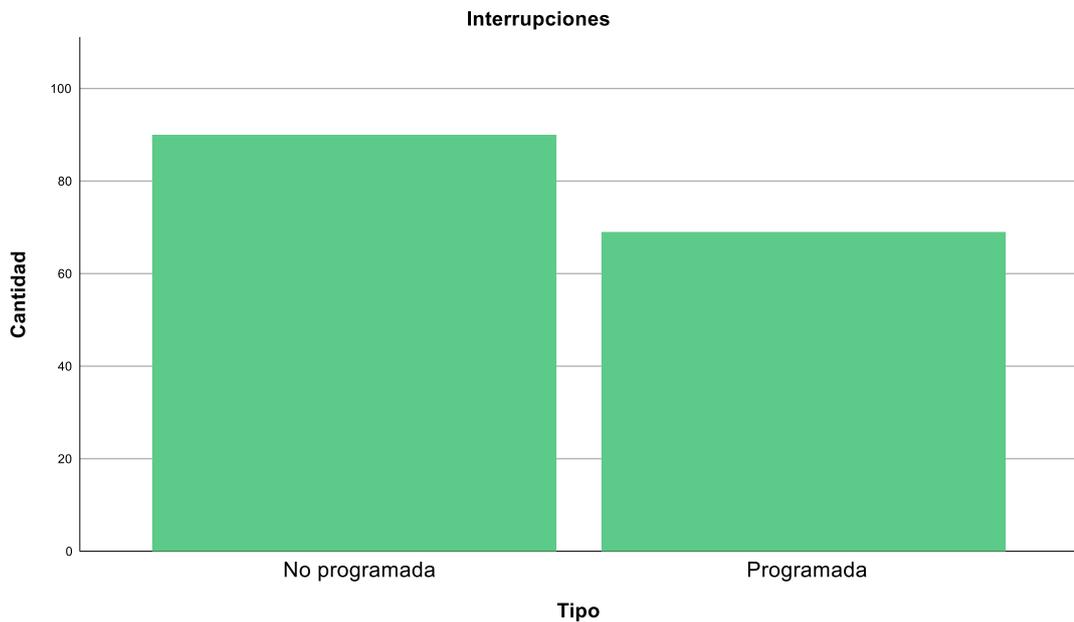


Figura 4-1. Cantidad de interrupciones.

Fuente: autor

Se puede observar que mayoritariamente la cantidad de interrupciones no programadas son las que se producen con un total de 90, mientras que las programadas con 69, las no programadas representan un 56% del total de interrupciones.

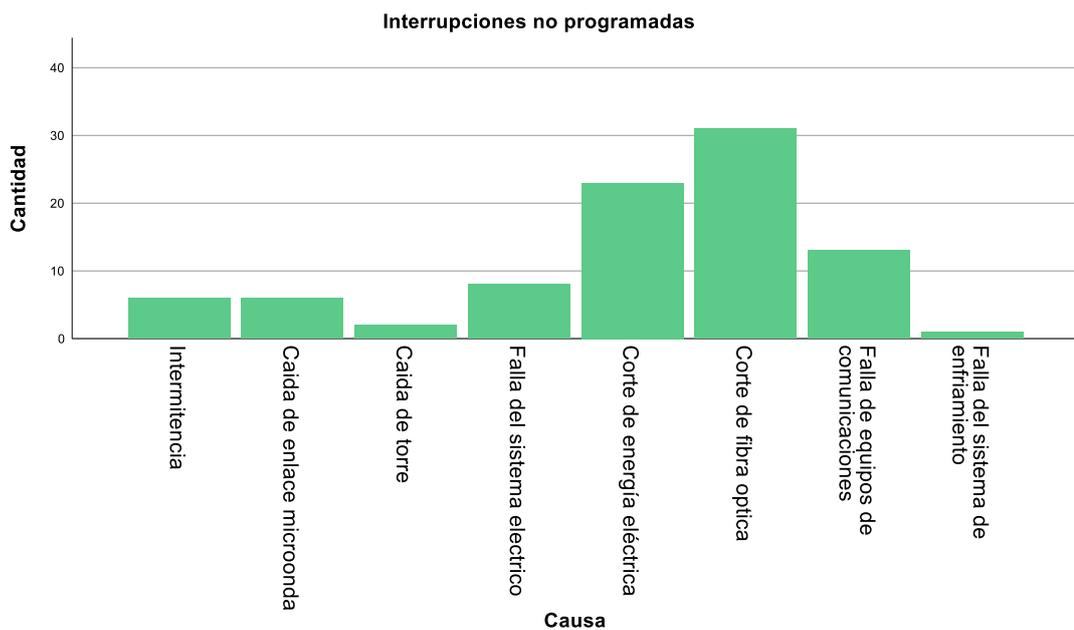


Figura 4-2. Interrupciones no programadas según la causa.

Fuente: autor

En el caso de las no programadas son mayoritariamente producidas por cortes de fibra óptica con un total de 31 que representan un 34% del total de interrupciones no programadas, la segunda causa con mayor incidencia corresponde a cortes de energía eléctrica con un total de 23 que representa un 25%, mientras que la cause menos recurrente es la de falla del sistema de enfriamiento con 1 interrupción únicamente.

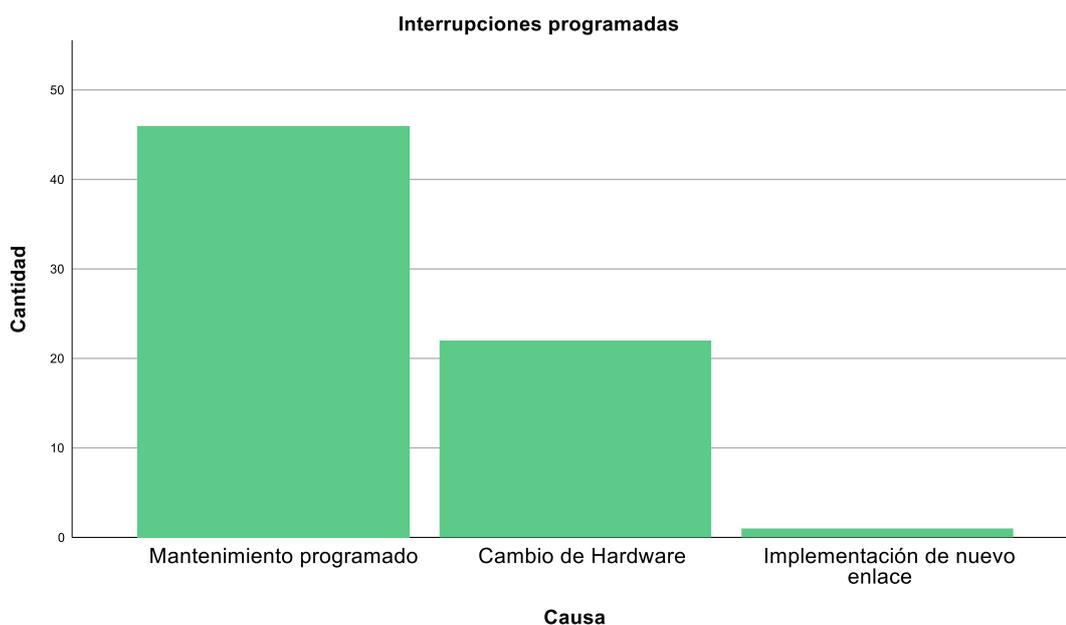


Figura 4-3. Interrupciones programadas según la causa.

Fuente: autor

En el caso de las programadas mayoritariamente se producen para mantenimientos programados con un total de 46 que representa un 66% del total, mientras que en último lugar se encuentra la implementación de un nuevo enlace con una únicamente.

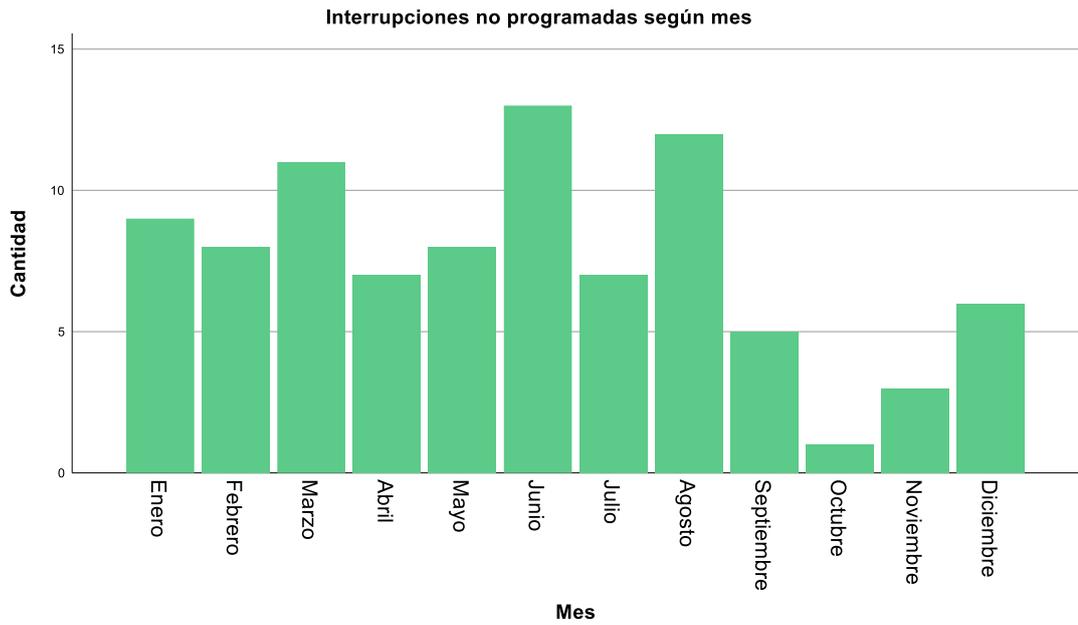


Figura 4-4. Interrupciones no programadas por mes.

Fuente: autor

En el caso de las no programadas se producen mayoritariamente en el mes de con 13 con un 14% del total, mientras que en octubre se produce la menor cantidad con una únicamente.

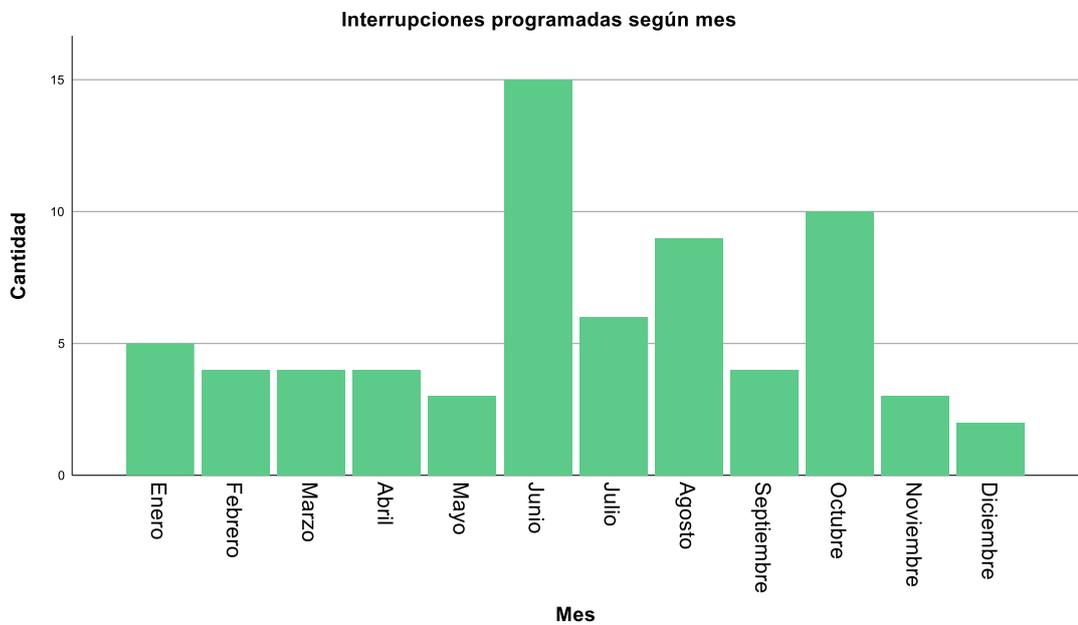


Figura 4-5. Interrupciones programadas por mes.

Fuente: autor

Para el caso de las interrupciones programadas la mayor cantidad se encuentra en junio con un total de 15 que representan un 21%, y la menor cantidad en diciembre con un total de 2.

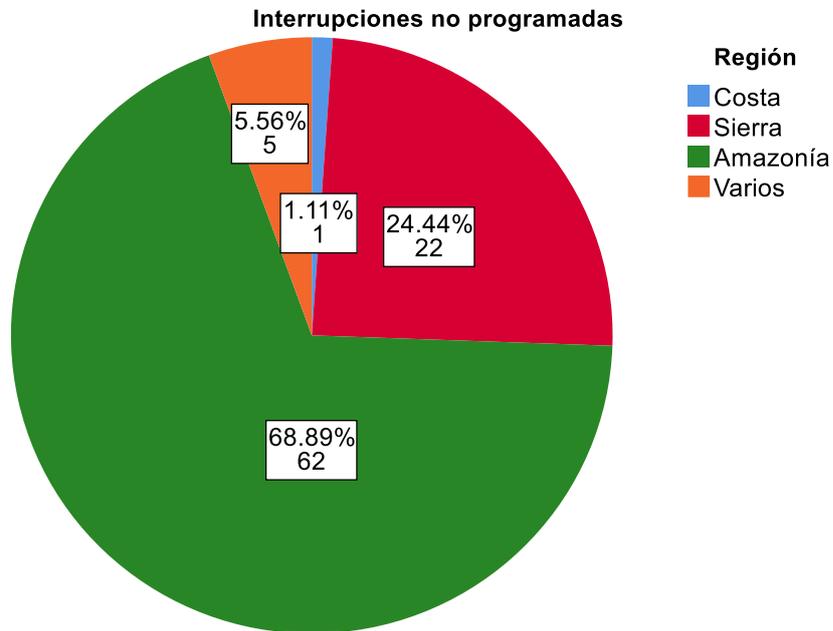


Figura 4-6. Interrupciones no programadas por región.

Fuente: autor

En el caso de distribución por región, mayormente se producen en la región Amazónica con un total de 62 que representa un 68% del total, y una menor cantidad en la región Costa con una interrupción.

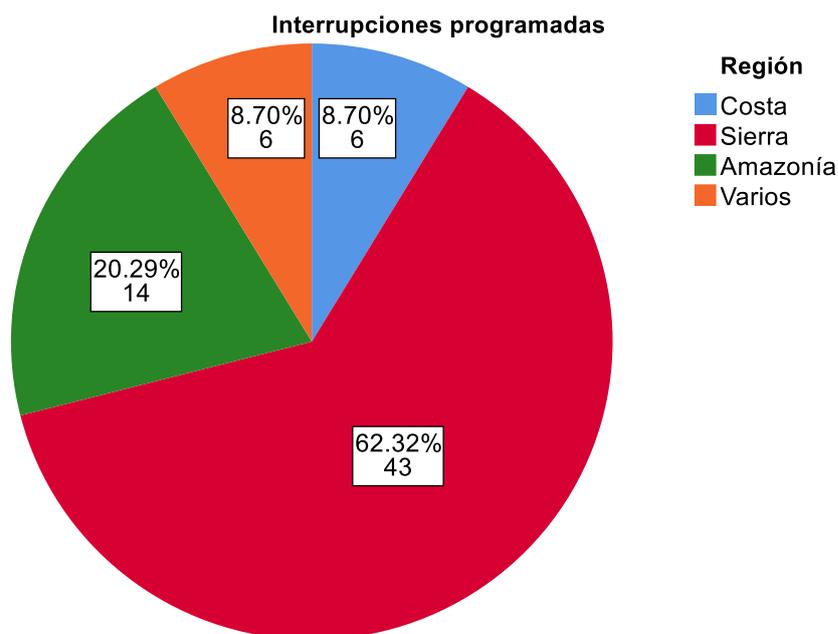


Figura 4-7. Interrupciones programadas por región.

Fuente: autor

Para el caso de las programadas se tiene mayor número en la región Sierra con un total de 43 con un 62% del total y con igual incidencia en múltiples regiones al mismo tiempo, así como en región Costa con 6 interrupciones en cada categoría.

Tabla 4-6. Duración de interrupciones.

Tipo	Rango	Mínimo	Máximo	Media	Desviación Std.	Varianza
No programadas	23:18:00	00:10:00	23:28:00	04:31:41	06:05:37	01:32:50
Programadas	09:00:00	02:00:00	11:00:00	08:12:30	02:33:14	00:16:18

Las duraciones para cada tipo de interrupción como se observa en la tabla anterior son de un promedio de 4 horas y 30 minutos para las no programadas y 8 horas con 12 minutos para las programadas, con rango de duración mucho más amplio para las no programadas de más de 23 horas, y con un rango de 9 horas para las programadas.

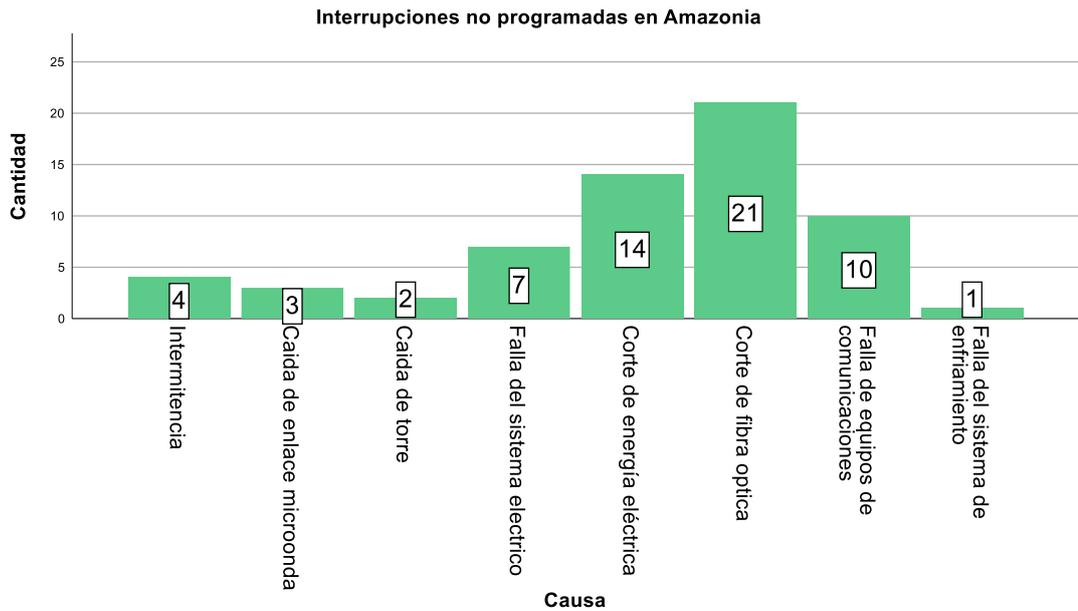


Figura 4-8. Interrupciones no programadas en región Amazónica.

Fuente: autor

Como pudo observarse en los analizado anteriormente las interrupciones no programadas están mayormente localizadas en la región Amazónica y con la causa más reincidente de corte de fibra óptica, lo que indica posiblemente una relación entre esas dos variables.

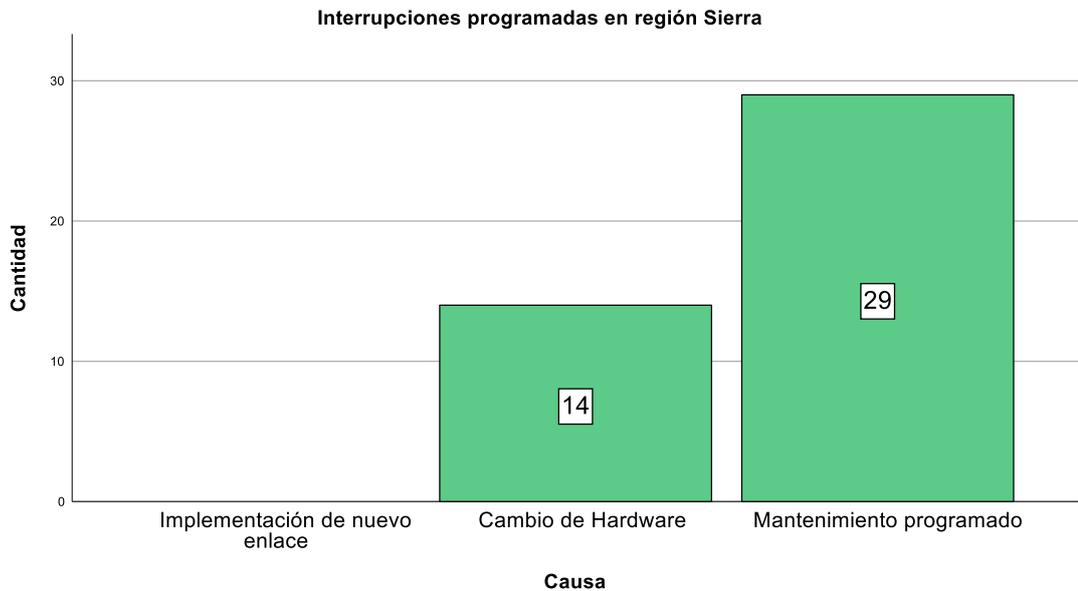


Figura 4-9. Interrupciones programadas en región Sierra.

Fuente: autor

Para el caso de las programadas existe una mayor cantidad en región Sierra para mantenimiento programado.

Para determinar si existe una relación entre la ubicación y las causas de interrupciones se desarrollará el método de chi-cuadrado de independencia.

MÉTODO CHI-CUADRADO DE INDEPENDENCIA

El análisis de independencia se lo realizara únicamente para las interrupciones no programadas dado que son de carácter aleatorio, no predecibles, y es de mayor interés determinar alguna relación o información que permita comprender en alguna medida la naturaleza.

Las variables de Región y de Causa son las seleccionadas para el análisis, las dos variables son de tipo multinomial con múltiples categorías posibles cada una como se definió en la descripción inicial de la información.

4.4.1 HIPÓTESIS

Se establece la hipótesis de nulidad y la hipótesis alternativa que son las que se confirmaran o descartaran según el resultado de la comparación de los cálculos y la significancia estadística.:

H_0 : La variable Región y la variable Causa son independientes.

H_A : Existe una asociación entre la variable Región y la variable Causa.

4.4.2 VALOR DE α

El valor de α será de 0.05, lo que indica una probabilidad del 5%, este valor está definido para establecer los valores de chi-cuadrado que estarán fuera del intervalo que correspondería a la hipótesis nula. El valor está definido como estándar para una significancia estadística valida. [33]

4.4.3 VALOR CRÍTICO

El valor critico está en función del valor α y de los grados de libertad, para este caso se tiene el cálculo de los grados de libertad, mediante este valor se podrá descartar la hipótesis nula en caso de que el valor calculado sea menor que el valor critico obtenido de la tabla de distribución chi-cuadrado.

Para los valores de R y C se modificó las variables, debido a que algunas celdas no contaban con datos suficientes para la evaluación se procedió a omitir varias categorías, teniendo así únicamente las siguientes variables.

Tabla 4-7. Variables modificadas.

Variable	Grupo
Causa	Falla del sistema eléctrico
	Corte de energía eléctrica
	Corte de fibra óptica
	Falla de equipos de comunicaciones
Localidad	Sierra
	Amazonía

$$df = (R - 1)(C - 1)$$

$$df = (4 - 1)(2 - 1) = 3$$

Con esto se seleccionará el valor crítico a partir de la tabla de distribución de probabilidad (Anexo 1: Valores críticos de χ^2):

Tabla 4-8. Valores críticos.

α	df	χ^2_{crit}
0.05	3	7.81

4.4.4 TABULACIÓN CRUZADA

Se realizó una prueba de independencia de chi-cuadrado entre Causa y Región. Las frecuencias de celdas esperadas fueron mayores que cinco para un 75%. Así se procedió con la tabulación cruzada mediante el cálculo de los valores esperados y se los contrastó con los valores medidos.

Tabla 4-9. Tabulación cruzada Región*Causa.

Causa		Región		Total
		Sierra	Amazonía	
Falla del sistema eléctrico	Frecuencia	0	7	7
	Frecuencia esperada	1.8	5.2	7.0
	% Causa	0.0%	100.0%	100.0%
	% Región	0.0%	13.5%	10.0%
	Residual ajustado	-1.6	1.6	
Corte de energía eléctrica	Frecuencia	9	14	23
	Frecuencia esperada	5.9	17.1	23.0
	% Causa	39.1%	60.9%	100.0%
	% Región	50.0%	26.9%	32.9%
	Residual ajustado	1.8	-1.8	
Corte de fibra óptica	Frecuencia	6	21	27
	Frecuencia esperada	6.9	20.1	27.0
	% Causa	22.2%	77.8%	100.0%
	% Región	33.3%	40.4%	38.6%
	Residual ajustado	-0.5	0.5	
Falla de equipos de comunicaciones	Frecuencia	3	10	13
	Frecuencia esperada	3.3	9.7	13.0
	% Causa	23.1%	76.9%	100.0%
	% Región	16.7%	19.2%	18.6%
	Residual ajustado	-0.2	0.2	
Total	Frecuencia	18	52	70
	Frecuencia esperada	18.0	52.0	70.0
	% Causa	25.7%	74.3%	100.0%
	% Región	100.0%	100.0%	100.0%

En la tabla anterior que el porcentaje presente de las interrupciones es de un 74.3% para la región Amazónica y solo un 25.7% para la región Sierra, así también la causa más recurrente es el corte de fibra óptica, con un total de 77.8% producidas en la Amazonía y un 22.2% para la Sierra.

Tabla 4-10. Frecuencias.

Causa		Región		Total
		Sierra	Amazonía	
Falla del sistema eléctrico	Frecuencia	0	7	7
	Frecuencia esperada	1.8	5.2	7.0
Corte de energía eléctrica	Frecuencia	9	14	23
	Frecuencia esperada	5.9	17.1	23.0
Corte de fibra óptica	Frecuencia	6	21	27
	Frecuencia esperada	6.9	20.1	27.0
Falla de equipos de comunicaciones	Frecuencia	3	10	13
	Frecuencia esperada	3.3	9.7	13.0
Total	Frecuencia	18	52	70
	Frecuencia esperada	18.0	52.0	70.0

Las frecuencias observadas muestran mucha cercanía con los valores esperados para los grupos de corte de fibra óptica y falla en equipos de comunicaciones, por otro lado, la frecuencia de corte de energía eléctrica en las dos regiones presenta variaciones de mayor magnitud lo que da un indicio que no existe una independencia entre las dos variables, del mismo modo en la causa falla del sistema eléctrico.

4.4.5 PRUEBA CHI-CUADRADO DE INDEPENDENCIA Y NIVEL DE ASOCIACIÓN

Se obtuvo el valor del factor chi-cuadrado mediante la fórmula:

$$\chi^2_{calc} = \sum \frac{(f_o - f_e)^2}{f_e}$$

Tabla 4-11. Valor chi-cuadrado y Cramer phi.

	Valor	df	Significación aproximada (p)
Pearson Chi-Square	4.810	3	0.186
Cramer's V	0.262		0.186

Comparando con el valor crítico se tiene:

Tabla 4-12. Valores chi-cuadrado, crítico y calculado.

χ^2_{calc}	χ^2_{crit}
4.810	7.81

Por tanto, se puede observar en la Tabla 4-12 que el valor calculado es menor que el valor crítico lo que significaría que se podría descartar la hipótesis nula y

comprobar la hipótesis alternativa, pero debido a que la significancia estadística es 0.186, lo que representaría un 18.6% que es mayor al 5% establecido, no se puede descartar la hipótesis nula.

4.4.6 PRUEBA POST HOC

Debido a que no se puede descartar la hipótesis nula porque no hay una significancia estadística debido al valor de $p > 0.05$, por el contrario, el valor de chi-cuadrado obtenido es menor al valor crítico es necesario realizar un análisis post hoc. Para poder determinar si existe una relación de asociación se analizará los valores residuales que son el residuo de entre la frecuencia observada y la esperada, esto indica que mientras mayor sea el valor residual mayor será la diferencia de las frecuencias por lo tanto esto indica una dependencia de las variables lo que comprobaría la hipótesis alternativa.

Tabla 4-13. Valores residuales.

Causa	Región	
	Sierra	Amazonia
Falla del sistema eléctrico	-1.6	1.6
Corte de energía eléctrica	1.8	-1.8
Corte de fibra óptica	-0.5	0.5
Falla de equipos de comunicaciones	-0.2	0.2

Para este caso como se puede observar los valores residuales no son valores lo suficientemente grandes para que representen una asociación entre las variables Causa y Región, por lo tanto, no se puede descartar la hipótesis nula, la variable Región y la Variable Causa son independientes.

4.4.7 REPORTE DE RESULTADOS

Se realizó una prueba de independencia de chi-cuadrado entre Región y Causa. Las frecuencias de celdas esperadas fueron mayores que cinco para un 75% del total de las muestras. No hubo una asociación estadísticamente significativa entre Región y Causa, $\chi^2 (3) = 4.81$, $p = .186$. La asociación fue pequeña [33], $V(\phi)$ de Cramer = .262.

CAPÍTULO 5: CONCLUSIONES Y RECOMENDACIONES

El diseño y construcción de la plataforma Web, se planteó con visión de escalabilidad y compatibilidad para nuevos servicios, mediante el uso de tecnologías actuales y estructuras de software propuestas apegadas a las recomendaciones dadas por los principales organismos de estandarización Web, otro factor importante que permite cumplir con la visión planteada, es que la plataforma está desarrollada en código propio es decir el código es escrito desde cero por parte de los autores y no hace ningún uso de un CMS (Content Management System), por ello la evolución del código y su mantenimiento no se encuentra limitado por dependencias de terceros.

La plataforma permite un seguimiento y administración de interrupciones con enfoque simplista mediante las herramientas planteadas, estas herramientas fueron diseñadas y construidas en base a los requerimientos establecidos por los funcionarios de la ARCOTEL, cada requerimiento genera una herramienta propuesta, por lo que las herramientas propuestas son susceptible a cambios o actualizaciones, por ello el enfoque de estructura de árbol de componentes desacoplados propuesto para el aplicativo se presenta una alternativa útil para este proceso.

El aplicativo se estructuró en base al procedimiento actual para el manejo de interrupciones, es decir su objetivo es ser el intermediario para el manejo y seguimiento de interrupciones entre operadores y funcionarios de la ARCOTEL, permitiendo un flujo de trabajo relativamente más sencillo para este proceso, debido a que el aplicativo se encarga de los procesos a nivel visual, organizacional y registro de información, para el seguimiento y administración de la interrupción más no toma decisiones acerca de la interrupción de manera autónoma.

En las estadísticas descriptivas se pudo observar que la causa predominante de las interrupciones no programadas fue los cortes de fibra óptica para la región Amazónica, del mismo modo para las interrupciones programadas fueron predominantes en la región Sierra con la causa de mantenimiento programado, esto da una posible relación entre la causa de la interrupción y la región en donde se produce y una tendencia a posibles interrupciones para dichas regiones.

Mediante el método chi-cuadrado no se logró establecer una relación entre la causa y la región de las interrupciones no programadas debido a que no presentó una significancia estadística, es decir dado los resultados obtenidos las variables causa y región son independientes. No se puede descartar la posibilidad de asociación entre las dos variables debido a que la cantidad de datos empleada no fue la más óptima si no que se encontraba en el límite para poder ejecutar el análisis por lo tanto en futuros análisis se podrá dar uso de la información recopilada por la plataforma de interrupciones que permitirá obtener un resultado más contundente.

Es fundamental identificar y ajustar la información para poder determinar el método que mejor se ajuste a la naturaleza de esta, para así poder obtener mejores resultados y aprovechar todos los datos disponibles de una manera más eficiente.

REFERENCIAS BIBLIOGRÁFICAS

- [1] ARCOTEL, «arcotel.gob.ec,» Diciembre 2018. [En línea]. Available: http://www.arcotel.gob.ec/wp-content/uploads/2019/01/1.1.1-Lineas-activas-por-servicio_y_Densidad_Nov-2018_C.xlsx. [Último acceso: Enero 2019].
- [2] ARCOTEL, «arcotel.gob.ec,» Noviembre 2018. [En línea]. Available: http://www.arcotel.gob.ec/wp-content/uploads/2019/01/1.2-Radiobases-por-operador-y-tecnologia-nivel-provincial_Nov-2018_R.xlsx. [Último acceso: Enero 2019].
- [3] «LEY ORGANICA DE TELECOMUNICACIONES,» REGISTRO OFICIAL ÓRGANO DEL GOBIERNO DEL ECUADOR, Quito, 2015.
- [4] SUPERTEL, «MANUAL DE USUARIO APLICATIVO PARA NOTIFICACION DE INTERRUPCIONE4S DEL SERVICIO DE TELEFONIA FIJA».
- [5] ARCOTEL, «SEÑAL MÓVIL ECUADOR,» ARCOTEL, 2015. [En línea]. Available: <http://smovilecuador.arcotel.gob.ec/SenalMovilEcuadorWeb/mapas.html>. [Último acceso: Agosto 2018].
- [6] «Infoantenas,» SECRETARIA DE ESTADO PARA LA SOCIEDAD DE LA INFORMACION Y LA AGENDA DIGITAL , [En línea]. Available: <https://geoportal.minetur.gob.es/VCTEL/vcne.do>. [Último acceso: Agosto 2018].
- [7] D. Wetcher-Hendricks, Analyzing Quantitative Data, Hoboken: John Wiley & Sons, Incorporated , 2014.

- [8] E. Christmann, *Beyond the Numbers*, Arlington: National Science Teachers Association, 2011.
- [9] Impactum, «Impactum,» Impactum, [En línea]. Available: <https://impactum.mx/diferencia-pagina-web-plataforma-web-apps/>. [Último acceso: Agosto 2018].
- [10] WorldWideWebSize.com, «WorldWideWebSize.com,» WorldWideWebSize.com, [En línea]. Available: <https://www.worldwidewebsite.com/>. [Último acceso: Enero 2019].
- [11] D. E. Comer, *COMPUTER NETWORKS AND INTERNETS*, New Jersey: Pearson, 2014.
- [12] T. Zhiyentayev, «Hackernoon,» Hackernoon, [En línea]. Available: <https://hackernoon.com/server-side-vs-client-side-rendering-in-react-apps-443efd6f2e87>. [Último acceso: Julio 2018].
- [13] J. Eguíluz Pérez, *Introducción a JavaScript*, uniwebsidad , 2008.
- [14] J. Eguíluz Pérez, *Introducción a AJAX*, uniwebsidad , 2008.
- [15] C. Lindley, *DOM Enlightenment*, Sebastopol: O'Reilly Media, Inc., 2013.

- [16] react.org, «react.org,» Facebook Inc., [En línea]. Available: <https://reactjs.org/>. [Último acceso: Agosto 2018].
- [17] A. Banks y E. Porcello, Learning React, Functional Web Development with React and Redux, Sebastopol: O'Reilly Media, Inc., 2017.
- [18] BBVA, «BBVA,» BBVA, 23 Marzo 2016. [En línea]. Available: <https://bbvaopen4u.com/es/actualidad/postcss-que-es-y-por-que-es-tan-popular>. [Último acceso: Agosto 2018].
- [19] T. Fredrich, RESTful Service Best Practices, Pearson eCollege, 2013.
- [20] alexserver, «GitHubGist,» 2016. [En línea]. Available: <https://gist.github.com/alexserver/2fcc26f7e1ebcfc9f6d8>. [Último acceso: Agosto 2018].
- [21] Postgres, «PostgreSQL Tutorial,» PostgreSQL, [En línea]. Available: <http://www.postgresqltutorial.com/what-is-postgresql/>. [Último acceso: Septiembre 2018].
- [22] Asamblea Constituyente, «CONSTITUCIÓN DEL ECUADOR,» 2008.
- [23] SENPLADES, «www.buenvivir.gob.ec,» SENPLADES, [En línea]. Available: http://www.buenvivir.gob.ec/image/image_gallery?img_id=11945&t=1374351407495. [Último acceso: Noviembre 2018].

- [24] ARCOTEL, «CONTRATO DE CONCESIÓN PARA LA PRESTACIÓN DEL SERVICIO MÓVIL AVANZADO».
- [25] R. Kumar, «FreeCodeCamp,» FreeCodeCamp, 02 Abril 2018. [En línea]. Available: <https://medium.freecodecamp.org/server-side-rendering-your-react-app-in-three-simple-steps-7a82b95db82e>. [Último acceso: Agosto 2018].
- [26] A. A. Vega, «Medium,» Medium, 17 Diciembre 2017. [En línea]. Available: <https://medium.com/@alonsus91/tipos-de-componentes-en-reactjs-f387a6f8e2b7>. [Último acceso: Septiembre 2018].
- [27] S. Xalambri, «Medium,» Medium, 13 Marzo 2016. [En línea]. Available: <https://medium.com/react-redux/introducci%C3%B3n-a-redux-js-8bdf4fe0751e>. [Último acceso: Septiembre 2018].
- [28] P. Patel, «FreeCodeCamp,» FreeCodeCamp, 18 Abril 2018. [En línea]. Available: <https://medium.freecodecamp.org/what-exactly-is-node-js-ae36e97449f5>. [Último acceso: Agosto 2018].
- [29] T. Piros, «FULLSTACK DEVELOPER ACADEMY,» FULLSTACK DEVELOPER ACADEMY, 17 Julio 2018. [En línea]. Available: <https://fullstack-developer.academy/concurrent-http-connections-in-node-js/>. [Último acceso: Septiembre 2018].
- [30] F. Inguanez, Entity Relationship Modelling, 2012.
- [31] MDN web docs, «developer.mozilla.org,» developer.mozilla.org, [En línea]. Available: https://developer.mozilla.org/en-US/docs/Web/API/IndexedDB_API/Basic_Concepts_Behind_IndexedDB. [Último acceso: Diciembre 2018].

[32] Docker, «Docker,» Docker, [En línea]. Available: <https://www.docker.com/resources/what-container>. [Último acceso: Diciembre 2018].

[33] J. Cohen, Statistical Power Analysis for the Behavioral Sciences, New York: Lawrence Erlbaum, 1988.

[34] P. Patel, «FreeCodeCamp,» FreeCodeCamp, 18 Abril 2018. [En línea]. Available: <https://medium.freecodecamp.org/what-exactly-is-node-js-ae36e97449f5>. [Último acceso: Agosto 2018].

APÉNDICES

APÉNDICE A: ESTRUCTURA DE BASE DE DATOS

APÉNDICE B: DIAGRAMA DE ÁRBOL DE COMPONENTES

SMA_APP

APÉNDICE C: DIAGRAMA DE PROCESOS PARA MANEJO DE

APLICATIVO

APÉNDICE D: ESTRUCTURA DE ARCHIVOS SMA_API

APÉNDICE E: ÁRBOL DE ESTADOS GLOBAL

ANEXOS

ANEXO 1: VALORES CRÍTICOS DE χ^2

Valores críticos de distribución chi-cuadrado. [7]

<i>df/p</i>	.750	.500	.250	.100	.050	.025	.010	.005
1	0.10153	0.45494	1.32330	2.70554	3.84146	5.02389	6.63490	7.87944
2	0.57536	1.38629	2.77259	4.60517	5.99146	7.37776	9.21034	10.59663
3	1.21253	2.36597	4.10834	6.25139	7.81473	9.34840	11.34487	12.83816
4	1.92256	3.35669	5.38527	7.77944	9.48773	11.14329	13.27670	14.86026
5	2.67460	4.35146	6.62568	9.23636	11.07050	12.83250	15.08627	16.74960
6	3.45460	5.34812	7.84080	10.64464	12.59159	14.44938	16.81189	18.54758
7	4.25485	6.34581	9.03715	12.01704	14.06714	16.01276	18.47531	20.27774
8	5.07064	7.34412	10.21885	13.36157	15.50731	17.53455	20.09024	21.95495
9	5.89883	8.34283	11.38875	14.68366	16.91898	19.02277	21.66599	23.58935
10	6.73720	9.34182	12.54886	15.98718	18.30704	20.48318	23.20925	25.18818
11	7.58414	10.34100	13.70069	17.27501	19.67514	21.92005	24.72497	26.75685
12	8.43842	11.34032	14.84540	18.54935	21.02607	23.33666	26.21697	28.29952
13	9.29907	12.33976	15.98391	19.81193	22.36203	24.73560	27.68825	29.81947
14	10.16531	13.33927	17.11693	21.06414	23.68479	26.11895	29.14124	31.31935
15	11.03654	14.33886	18.24509	22.30713	24.99579	27.48839	30.57791	32.80132
16	11.91222	15.33850	19.36886	23.54183	26.29623	28.84535	31.99993	34.26719
17	12.79193	16.33818	20.48868	24.76904	27.58711	30.19101	33.40866	35.71847
18	13.67529	17.33790	21.60489	25.98942	28.86930	31.52638	34.80531	37.15645
19	14.56200	18.33765	22.71781	27.20357	30.14353	32.85233	36.19087	38.58226
20	15.45177	19.33743	23.82769	28.41198	31.41043	34.16961	37.56623	39.99685