UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS

FACULTAD 2

SISTEMA DE ANÁLISIS DE TRÁFICO DE LLAMADAS TELEFÓNICAS PARA LA EMPRESA DE TELECOMUNICACIONES DE CUBA S.A.

TRABAJO DE DIPLOMA PARA OPTAR POR EL TÍTULO DE INGENIERO EN CIENCIAS INFORMÁTICAS

Autores: Alexander Valdés Molina

Braiman González Sánchez

Tutores: MSc.Yasser Azán Basallo

Ing.Yaislenis Landabe Barbarú

La Habana, junio 2016

Declaración jurada de autoría

Declaro ser autor de la presente tesis y reconozco a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste firmamos a los la presente a los ____ días del mes de _____ del año _____.

Autores

_____ _____

Braiman González Sánchez Alexander Valdés Molina

Tutores

_____ _____

MSc. Yasser Azán Basallo Ing.Yaislenis Landabe Barbarú

Resumen

En las últimas décadas ha existido un amplio crecimiento de los fraudes en las telecomunicaciones a nivel mundial y principalmente en Cuba. Este escenario en el país se ha distinguido por un creciente uso de los dispositivos móviles y el acceso a Internet, como consecuencia ha aumentado el número de fraudes, los cuales comprometen la disponibilidad y calidad de los servicios que ofrece la Empresa de Telecomunicaciones de Cuba S. A. (ETECSA). El fraude en el sector es conocido con el uso ilegal de los medios de telecomunicaciones y todo acto por eludir o burlar el pago de los servicios.

La investigación presentada se titula: “Sistema de análisis de tráfico de llamadas telefónicas para la Empresa de Telecomunicaciones de Cuba S. A.”, en la misma se aborda el tema referente a los sistemas de gestión de alertas de posibles fraudes, así como sus características. Por consiguiente los tesistas proponen como objetivo general, desarrollar un sistema de análisis de las alertas en las llamadas telefónicas para el departamento de antifraude correspondiente a la Empresa de Telecomunicaciones de Cuba S. A.

Los resultados finales de la investigación elaborada se materializan en el desarrollo de un sistema de gestión de análisis de alertas (SGAT) en el departamento de antifraude de la Empresa de Telecomunicaciones de Cuba S. A.

PALABRAS CLAVES: alertas, ETECSA, fraude, gestión, SGAT

Abstract

In recent decades there has been extensive growth of fraud in telecommunications worldwide and especially in Cuba. This scenario in the country has been marked by an increasing use of mobile devices and Internet access, as a result has increased the number of scams, which compromise the availability and quality of services offered by the Telecommunications Company of Cuba S. A. (ETECSA). Fraud in the sector is known to the illegal use of telecommunications facilities and any act to evade or circumvent the payment of services.

The research presented is entitled "System traffic analysis of phone calls to the Telecommunications Company of Cuba S. A.," on the same theme concerning management systems alerts of possible fraud is addressed, as well as its characteristics. Therefore the proposed thesis student general objective to develop an analysis system alerts in telephone calls to the department corresponding to the Telecommunications Company of Cuba antifraud S. A.

The final results of research contribute elaborate management system analysis alerts (SGAT) in the anti-fraud department of the Telecommunications Company of Cuba S. A.

KEYWORDS: alerts, ETECSA, fraud management, SGAT

Índice

Introducción

Con el desarrollo de las Tecnologías de la Información y la Comunicaciones (TIC) a través de los años, la informática y las telecomunicaciones han evolucionado a una velocidad nunca antes soñada por el hombre, dándole a este un sin fin de facilidades para su quehacer cotidiano. En estos días no se concibe las telecomunicaciones sin la presencia de la informática como principal impulsor de su desarrollo progresivo, en áreas como la gestión de datos, el análisis de tráfico y la seguridad, convirtiéndose esta última en un objetivo primordial a desarrollar por las empresas distribuidoras de servicios telemáticos.

El término seguridad en las telecomunicaciones se ha visto limitado a la banca, las aplicaciones aeroespaciales o militares. Pero, debido al rápido y amplio crecimiento de las telecomunicaciones de datos (a partir del surgimiento y desarrollo de Internet), la seguridad se ha convertido en una preocupación universal, por la fuerza que ha alcanzado el fraude.

En este sentido, en el sector específico de las telecomunicaciones juega un rol esencial la seguridad de las redes y los servidores que brindan este servicio, en tanto gestionar, controlar y medir el comportamiento de los fraudes es de vital importancia para una empresa de estas características. El fraude[1] como uno de los flagelos que tienen hoy las telecomunicaciones *"(…) produce pérdidas de alrededor del 4.8% del total de ingresos económicos obtenidos por las empresas que brindan estos servicios*" (1).De igual modo el proceso de gestión de fraude en este sector es altamente priorizado por las empresas que lo integran, aunque no todas las realizan utilizando el mismo proceder.

Regularmente, estas empresas buscan implementar en sus procesos de gestión diferentes métodos para detectar conductas inusuales por parte de los suscriptores, que puedan reflejar potenciales usos indebidos de los servicios o casos de fraude, donde en un primer instante serán identificados como alertas de fraude.

Cada empresa en particular es capaz de implementar según sus recursos materiales, económicos y humanos, uno o disimiles métodos para la gestión del fraude en sus redes. Estos métodos pueden dividirse en dos grupos, un primer grupo que operan en tiempo real y un segundo que opera sobre la base de un registro del tráfico emitido por las líneas telefónicas. Sin embargo, aunque tienden a parecer muy diferentes, ambos grupos convergen en el mismo objetivo, lanzar alertas de posibles fraudes para su análisis.

[1] Fraude: Engaño, con la intención de conseguir un beneficio, y con el cual alguien queda perjudicado.

Las alertas de posibles fraudes son analizadas indistintamente en cada empresa dependiendo de cuan automatizado esté su proceso de gestión de alertas. La calidad de este proceso es sumamente importante y se encuentra dada en la gran mayoría de los casos por la variable tiempo, pues cuanto más elevada sea, mayor será el impacto negativo que tendrá el posible hecho fraudulento, tanto para la empresa como para sus clientes y su país.

La Empresa de Telecomunicaciones de Cuba S.A (ETECSA), no está exenta de este fenómeno. En la última década el escenario de las telecomunicaciones en el país se ha caracterizado por el auge del uso de los servicios celulares así como a un mayor acceso a la red mundial de redes, Internet. Esto amplía las posibilidades para acometer hechos de fraude, los que comprometen la disponibilidad y calidad de los servicios que se ofertan y por tanto, afectan la imagen de la empresa así como sus ingresos (2).

Para la detección de las diferentes modalidades de fraude ETECSA cuenta con un departamento de antifraude, en el que se ha desarrollado un sistema de gestión de fraude basado en un método que opera sobre la base de un registro del tráfico cursado por las líneas telefónicas, llamado perfilamiento de tráfico telefónico.

Este método comienza en esta empresa con la obtención de las alertas de seguridad dadas por el sistema Nikira[2], por parte de los investigadores del departamento. Luego obtienen los datos de los números que constituyeron alertas a través de un sistema gestor de base de datos llamado Discovery[3]. Finalmente son analizados estos datos en la herramienta HAT,[4] que es la encargada de realizar el análisis del tráfico de dichas alertas. Dicha herramienta está constituida por dos paquetes fundamentales, en el primero se almacena en una serie de hojas de cálculo creadas en Microsoft Excel que conforman una importante fuente de datos para la herramienta. Luego en el segundo paquete se encuentra una base de datos creada con Microsoft Access la cual se encarga de hacer todas las consultas pertinentes referentes con los números que constituyen sospechosos, la misma constituye la fuente de información de una segunda hoja de cálculo. Esta última se encuentra debidamente estructurada según las pautas establecidas por el departamento para el análisis de las alertas, mostrándose los datos resultantes de las consultas en una serie de tablas para de esta manera concluir con el proceso.

[2] Nikira: Sistema de alertas de fraude para telefonía móvil y fija de desarrollo hindú.

[3] Discovery: Sistema gestor de base de dato basado en Oracle.

[4] HAT: Herramienta de análisis de tráfico

Este procedimiento, con el desarrollo y aumento de los fraudes y las alertas en la actualidad, se vuelve engorroso para los investigadores, debido al número de análisis al unísono que deben realizar, por lo que pierden mucho tiempo en el proceso de obtener las alertas. Luego, obtener los datos y finalmente ubicarlos en el sistema HAT, donde se va a acometer el análisis. A este conjunto de acciones también se le suma la necesidad que tienen los investigadores de tener actualizada los datos que se guardan en las hojas de cálculo del primer paquete de herramienta, por lo que el jefe del departamento tiene la obligación de realizar esta tarea de actualización periódicamente, la cual es acometida actualizando una sola estación de trabajo a la vez. Por tanto, el proceso de gestión de alertas de posibles fraudes se ve obstaculizado debido a que los procedimientos que se realizan en él se han vuelto ineficientes y tediosos, lo que a su vez hace aumentar el impacto negativo en las ganancias de la empresa, del país y del servicio que la entidad brinda a la población.

Teniendo en cuenta la **situación problemática** antes expuesta se plantea como **problema a resolver** el siguiente: ¿Cómo contribuir al proceso de análisis de las alertas por fraude en las llamadas telefónicas en el departamento de antifraude de la Empresa de Telecomunicaciones de Cuba S.A.?

Partiendo del problema planteado, el **objeto de estudio** se enmarca en: el proceso de gestión de las alertas en telecomunicaciones.

El **campo de acción** se acota a: el proceso de análisis de las alertas de las llamadas telefónicas para detectar fraudes en el departamento de antifraude de la Empresa de Telecomunicaciones de Cuba S. A.

Para darle solución a la problemática descrita se define el **objetivo general:**

Desarrollar un sistema para contribuir al proceso de análisis de las alertas de las llamadas telefónicas en el departamento de antifraude perteneciente a Empresa de Telecomunicaciones de Cuba S. A.

Para dar cumplimiento al objetivo antes planteado se definieron las siguientes **preguntas de investigación:**

1. ¿Cuál es el estado actual de las herramientas y metodologías para la detención del fraude en el campo de las telecomunicaciones a nivel mundial?
2. ¿Cuál es el estado actual del fraude de las llamadas telefónicas en la región?
3. ¿Cómo evaluar el estado actual de las alertas de posibles fraudes en las llamadas telefónicas en el territorio nacional a través de la Empresa de Telecomunicaciones de Cuba S. A.?

4. ¿Cómo concebir un sistema que contribuya al proceso de análisis de las alertas de posibles fraudes en el departamento de antifraude de la Empresa de Telecomunicaciones de Cuba S. A.?
5. ¿Cómo diseñar un sistema que contribuya al proceso de análisis de las alertas de posibles fraudes en el departamento de antifraude de la Empresa de Telecomunicaciones de Cuba S. A.?
6. ¿Qué resultado se alcanzaran con el desarrollo de las pruebas de aceptación, caja negra, rendimiento y seguridad?

Para complementar las preguntas de investigación antes planteadas se definieron las siguientes **tareas de investigación**:

- Diagnóstico del estado y comportamiento de las metodologías y herramientas para la detención del fraude en el campo de las telecomunicaciones a nivel mundial.
- Determinación del estado actual del fraude de las llamadas telefónicas en la región.
- Análisis de los principales países que más inciden en alertas por fraude.
- Análisis del estado de las alertas por fraude en las llamadas telefónicas en el territorio nacional.
- Ejecución de las pruebas de aceptación, caja negra, rendimiento y seguridad.
- Validación de los resultados alcanzados con la propuesta del sistema para mejorar el proceso de análisis de las alertas de posibles fraudes de las llamadas en la Empresa de Telecomunicaciones de Cuba S. A.

Con el propósito de lograr una mayor organización en la realización de la presente investigación se recurre a los siguientes métodos científicos:

Métodos Teóricos:

Analítico–Sintético: Se utilizó durante el proceso de revisión bibliográfica para conocer el funcionamiento de la herramienta HAT, utilizada por ETECSA. Se utilizó además en el estudio y comprensión de toda la información recopilada para llegar a conclusiones válidas y necesarias para el desarrollo de la investigación.

Histórico-Lógico: Este método permite estudiar la trayectoria histórico real de la evolución y desarrollo de los sistemas de gestión del fraude.

Inducción-Deducción: Este método se utiliza con el objetivo de conocer la lógica y las características de la herramienta HAT y el sistema Nikira utilizados por ETECSA.

Métodos Empíricos:

La entrevista: A través de este método se obtuvo la información necesaria para la validación de los requisitos identificados por parte de los expertos de ETECSA.

La observación: Este método permite recopilar la información de cómo se desarrolla hoy el proceso de gestión de fraude en ETECSA.

Esta investigación se desarrolló sobre la base del actual proceso de cambios socio-económicos que vive Cuba, en el contexto de actualización del proyecto social cubano, pues tributa al objetivo No. 52 de la primera conferencia del partido en cuanto a aprovechar las ventajas de las tecnologías de las telecomunicaciones, como herramientas para el desarrollo del conocimiento(...)".

El aporte práctico de la presente investigación es un sistema para reducir el tiempo de análisis de las alertas de posibles fraudes en las llamadas telefónicas en el departamento de antifraude perteneciente a la Empresa de Telecomunicaciones de Cuba S.A.

El trabajo de diploma se divide en 3 capítulos, los cuales estarán estructurados de la siguiente forma:

Capítulo1. "**Fundamentación teórica y metodológica para el control y seguimiento a las alertas de fraude en telecomunicaciones a nivel nacional e internacional**": En este capítulo se recogen los principales sistemas para la gestión y detección del fraude en telefonía fija y móvil a nivel nacional e internacional. Así como la metodología y tecnologías de desarrollo del software a utilizar.
Capítulo 2. "**Propuesta y planificación del sistema**": Se presenta la propuesta de solución del sistema, así como las características y funcionalidades del mismo. Además, se describen las historias de usuarios de cada requisito del sistema. También, se realiza el plan de las iteraciones y el plan de entrega de las historias de usuario.
Capítulo 3. "**Implementación y Pruebas**": En este capítulo se plasma el proceso de implementación de la herramienta, garantizando así su correcto funcionamiento, mediante las pruebas realizadas. Para verificar que el producto cumple con los requisitos definidos por el cliente.

Capítulo 1: Fundamentación teórica y metodológica para el control y seguimiento de las alertas que constituyen fraude en telecomunicaciones a nivel mundial y en Cuba.

1.1 Introducción

Los fraudes que se cometen en la telefonía fija y móvil son sin duda una preocupación para el departamento de antifraude de ETECSA. En el presente capítulo se abordan las características y funcionalidades más importantes de la herramienta HAT[5], así como un análisis de distintos sistemas para la detección de los fraudes. Además, se realiza un estudio de las técnicas, tecnologías y metodologías de desarrollo del software a utilizar.

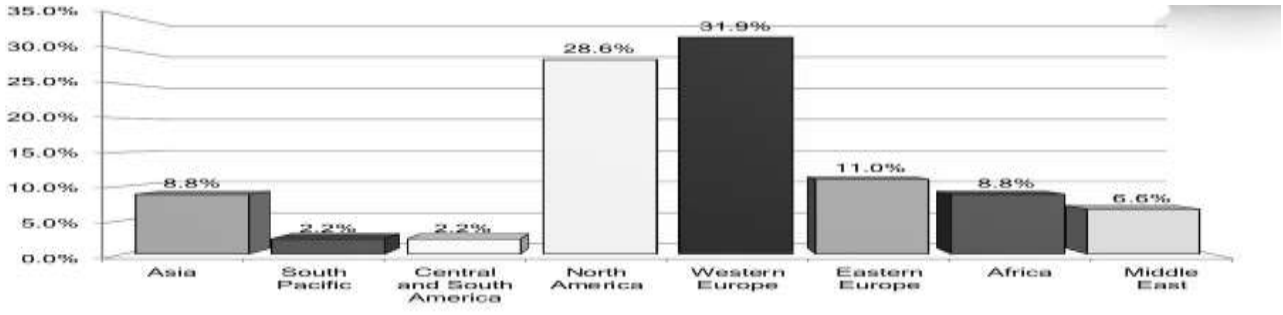
1.2 Análisis global sobre fraude en las telecomunicaciones.

A continuación se muestra un análisis global sobre el fraude en las telecomunicaciones que realizó la Communications Fraud Control Association (CFCA[6]). La CFCA comenzó en febrero de 1895 como un grupo de profesionales de la seguridad concienciado sobre el fraude en las telecomunicaciones, intentando encontrar una manera más eficiente de combatirlo. (3)

En la actualidad cuenta con más de 200 corporaciones que aúnan esfuerzos y elaboran informes acerca del estado del fraude, así como las nuevas tendencias cibercriminales en todo el mundo. Desde el 2009 y cada 2 años, elabora un estudio entre los miembros de la asociación para la elaboración de estadísticas (3).

En 2015, año de su último estudio realizado a las organizaciones y corporaciones consultadas, dentro de las que se encuentran proveedores de Wireless,[7] banda ancha, voz, datos o servicios financieros, arrojaron las estadísticas de cómo se encuentra el índice de fraude en las siguientes regiones (3).

Ilustración 1.1: Estudio del fraude a nivel mundial. (3)



[5] HAT: Herramienta de Análisis de Trafico desarrollada por ETECSA

[6] CFCA en español: Asociación de Control de Fraude en las Comunicaciones

[7] Wireless (inalámbrico o sin cables) es un término usado para describir las telecomunicaciones en las cuales las ondas electromagnéticas (en vez de cables) llevan la señal sobre parte o toda la trayectoria de la comunicación.

Según el estudio se estima que el fraude en las telecomunicaciones alrededor del mundo tiene pérdidas del 46.3 billones de dólares, incrementándose en un 15% desde su último estudio en el 2013. Debido a que cada vez es mayor la implementación de los entornos Wireless. (3)

Como se puede observar, América del norte y más específicamente Estados Unidos, presenta uno de los mayores índices de fraude a nivel mundial. Por lo que vuelve a Cuba en un punto muy vulnerable para la realización del fraude. En el caso de Cuba no es debe obviar, que cuenta con una gran comunidad de ciudadanos residentes en los Estados Unidos, hacia donde se registran más del 50 porciento de las llamadas exteriores.

1.2.1 Situación del fraude en América Latina

La expansión de las telecomunicaciones y el mayor uso de los teléfonos inteligentes, las amenazas contra la seguridad y el peligro de fraude se han intensificado. De acuerdo a la Unión Internacional de Telecomunicaciones (UIT), muchos operadores están sufriendo pérdidas de casi seis porciento de sus ingresos totales anuales, a causa de los fraudes. El fraude representa aproximadamente 14 porciento del total de la fuga potencial de ingresos de los operadores a nivel mundial. Esto hace necesaria una reevaluación de las estrategias destinadas a minimizar los costos operativos, maximizar la rentabilidad y hacer frente a la mayor competencia en los mercados regionales. (4)

La gestión de fraudes ayuda a los operadores a disminuir los casos de ineficiencia en sus sectores en cuanto la calidad y disponibilidad de los servicios que brinda su empresa. Esta disciplina intenta resolver el problema de la pérdida de ingresos. Durante la última década, la mayoría de los operadores contrataban empresas especializadas en la gestión de fraudes. Sin embargo, los nuevos desafíos, como por ejemplo la necesidad de contar con un enfoque en tiempo real tanto para la detección de fraudes y la capacidad de detectar y responder a nuevos patrones de fraudes, plantean cada vez más a los operadores la necesidad de encontrar una forma más alineada de manejar sus procesos de detección de fraudes.

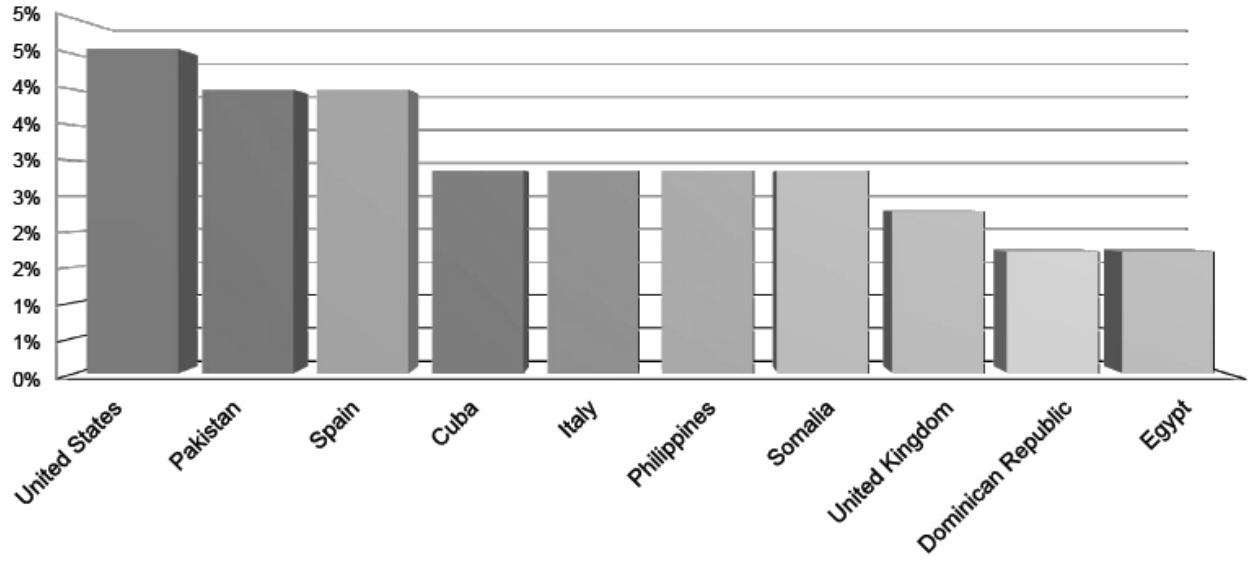
Actualmente, se considera que el mercado de las telecomunicaciones de América Latina, y en especial el sector de servicios móviles, es uno de los mercados con mayor potencial de crecimiento en el mundo. Según las cifras incluidas en el informe del Banco Mundial del año 2015, el 98% de la población latinoamericana tiene acceso a una señal de telefonía móvil y el 84% de los hogares está abonado a algún tipo de servicio móvil. Se prevé que América Latina lidere, junto con la región de Asia-Pacífico, la expansión mundial en servicios de telecomunicaciones en los próximos años. Para mantener su impresionante crecimiento y rentabilidad en el futuro, el mercado latinoamericano debe prestar especial atención a cuestiones como la gestión de fraudes. (4)

1.2.2 Análisis de los países que registran las principales acciones de fraudes:

A continuación se nuestra el último análisis a nivel global sobre el fraude en las comunicaciones que realizó la CFCA teniendo en cuenta su localización.

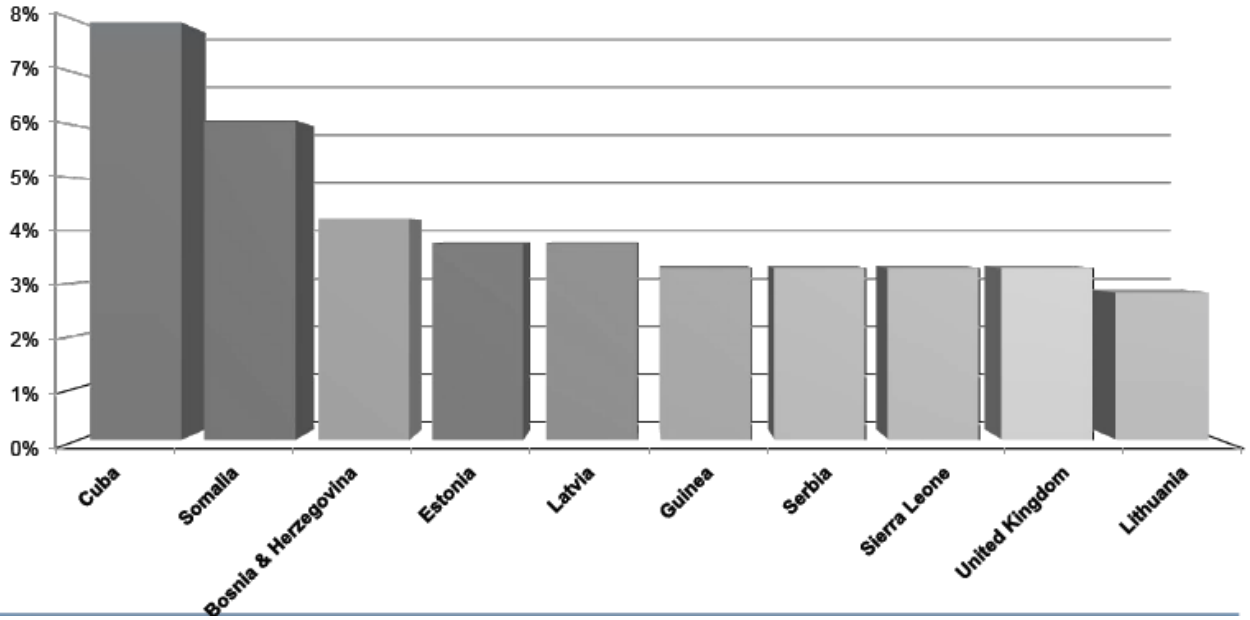
La gráfica muestra los 10 primeros países donde se origina el mayor número de llamadas que constituyen fraude.

Ilustración 1.2: Estudio del fraude a nivel mundial. (7)



La gráfica muestra los 10 primeros países que constituyen el destino que más se registra donde el fraude termina.

Ilustración 1.3: Estudio del fraude a nivel mundial. (7)



1.3 Sistemas para la detección y gestión del fraude en telefonía fija y móvil a nivel internacional:

SAP Fraud Management:

Fraud Management es un producto nuevo de SAP[8] englobado dentro de **GRC** (Gobierno, Riesgo y Cumplimiento) orientado a encontrar patrones de fraude para información SAP y no SAP, está basado en HANA[9] como BBDD[10] con un Netweaver ABAP[11] 7.4 por encima y frontend[12] en web. Tiene integración con sistemas SAP (vía SLT[13]) y no SAP (BO[14] Servicios de Datos) para llevar la información a HANA. Permite aplicar una serie de reglas (pre configuradas y permite reglas ad-hoc)[15] en tiempo real a la información para la búsqueda de posibles fraudes, a nivel de aplicación se controla todo el proceso relativo a la investigación de los posibles fraudes obtenidos a partir de las reglas anteriormente comentadas. Este conjunto de reglas constituye su mayor ventaja debido a que el investigador no tiene la necesidad de estar supervisando tan exhaustivamente el proceso de detección de fraude, solamente revisa las posibles amenazas, las clasifica y las incluye en la base de conocimiento del sistema mediante nuevas reglas. (5)

Mobileum:

Mobileum ayuda a los proveedores de servicios de telecomunicaciones a aprovechar el poder del análisis predictivo y datos en tiempo real para impulsar la transformación innovadora del negocio. En los últimos 15 años, Mobileum ha ofrecido soluciones innovadoras como Roaming[16] de Voz y Datos, Análisis de Fraude, Monetización de Datos, Análisis de la Experiencia del Abonado, y Administración y Seguridad en el Rendimiento del Servicio a 617 operadores de redes en 170 países. La plataforma Wisdom de Mobileum permite a los proveedores de servicios de telecomunicaciones entender en profundidad el comportamiento de la red y los abonados y aprovecharlo para mejorar la adopción del servicio, reducir la rotación, ofrecer una experiencia superior al cliente, predecir nuevos tipos de fraudes y obtener los beneficios de los modelos de ganancias de última generación basados en datos. Mobileum tiene sede en Silicon Valley, California, con oficinas en Bruselas, Hong Kong, Singapur, India, Dubái, Argentina, Uruguay y Brasil. (6)

Mobileum Anti-Fraud Analytics

Mobileum Anti-Fraud Analytics opera sobre su plataforma de Big Data y Análisis, cuya patente se encuentra pendiente, llamada Wisdom, que puede analizar en detalle a través de miles de millones

[8] SAP: Systems, Applications, Products in Data Processing (en español: Sistemas, Aplicaciones, Productos en el Procesamiento de datos).
[9] HANA: Hasso´s New Architecture.
[10] BBDD: Abreviatura de Bases de Datos.

[11] Plataforma de tecnología integrada para todas las aplicaciones SAP.
[12] Términos que se refieren a la separación de intereses entre una capa de presentación y una capa de acceso a datos.
[13] **SLT:** módulo de entradas/salidas remoto.

[14] BO: Business Objects.
[15] Reglas ad-hoc: conjunto de reglas específicas.

[16] Extiende la cobertura de los servicios brindados por la operadora de tu país.

de usos inesperados de transacciones en tiempo real para descubrir patrones anormales de comportamientos fraudulentos emergentes. Esto ayuda a ampliar la red para atrapar las desviaciones en los tipos conocidos de fraudes e identificar fraudes desconocidos en voz, datos y mensajes de texto, además de hacerlo en escenarios internacionales y de roaming. (6)

Recerca Anti Frau Financer Per Internet (RAFFI)

RAFFI[17] tiene por objetivo prevenir, detectar y corregir los posibles ataques a los cuales pueden estar sometidos los diferentes canales de banca electrónica (web, móvil y TDT[18]). La herramienta permite la evaluación en tiempo real del comportamiento de un acceso mediante cualquier canal y la detección del fraude en tiempo real basado en la historia del comportamiento anterior de diferentes usuarios accediendo a los diferentes canales. (7)

HAT

HAT es la herramienta para la gestión del fraude en telecomunicaciones de la telefonía móvil y fija que desarrolló ETECSA. En su momento la herramienta fue una novedosa solución para la detección del fraude en Cuba. La herramienta se encuentra dividida en dos paquetes que contienen varias hojas de cálculo de Microsoft Excel donde se cargan los datos de los números telefónicos que da como alerta el sistema Nikira. Luego, que se tiene toda la información cargada en el Excel[19], el investigador procede a detectar el fraude basándose en su experiencia y sin ninguna ayuda de la herramienta. Este proceso aumenta con el tiempo, a medida que aumentan las alertas. Además las bases de datos que se encuentran en el primer paquete tienen que ser actualizadas por el jefe del departamento periódicamente y en el orden de una estación de trabajo a la vez.

1.3.1 Resumen.

Luego del análisis realizado a los diferentes sistemas de gestión del fraude, se concluye que los mismos son sistemas informáticos propietarios, donde todo su análisis se basa en un conjunto de reglas de IA[20] predefinidas por el investigador y que además se ejecutan en tiempo real, sin utilizar como apoyo para la detección de las alertas alguna información almacenada previamente en una base de datos. De esta forma no satisfacen la necesidad de los investigadores de ETECSA de trabajar sobre los datos almacenados en el sistema Discovery y no en tiempo real. Además estos sistemas no son capaces de llevar a cabo la gestión de los casos de fraude archivados y en progreso, así como los datos de todos los dispositivos móviles que se encuentran en el país. En el caso de la herramienta HAT la misma no es capaz de realizar sus consultas directamente a la base

[17] RAFFI: en español, investigación anti fraude financiero por internet
[18] TDT: televisión digital terrestre.

[19] Excel: aplicación distribuida por la suite de oficina Microsoft Office para el trabajo con hojas de cálculo.

[20] IA: inteligencia artificial

de dato de Discovery, teniendo los investigadores que implicarse en un proceso engorroso de exportación de hojas de cálculo de tipo Excel para poder realizar el análisis de las alertas de posibles fraudes. También la herramienta no es capaz de llevar a cabo la gestión de los casos de fraude archivados y en progreso, así como los datos de todos los dispositivos móviles que se encuentran en el país. Por lo antes expuesto, se decide desarrollar un sistema informático que permita mejorar el proceso de gestión y análisis de las alertas de posibles fraudes a los números que emite el sistema Nikira.

1.4 Metodología de desarrollo de software

Posteriormente de haber abordado las principales tendencias, conceptos y métodos para la gestión del fraude en la telefonía tanto móvil como fija, se hace necesario la selección de una metodología de software adecuada que permita guiar el proceso de desarrollo de manera precisa para elaborar una solución a la problemática planteada.

Programación extrema (XP)

La programación extrema (XP) fue concebida y desarrollada para ocuparse de las necesidades específicas del desarrollo del software bajo la dirección de equipos pequeños haciendo frente a los requisitos imprecisos y cambiantes. Esta metodología ágil pone en duda muchas afirmaciones convencionales, incluyendo la vieja suposición de certidumbre que el costo de cambiar un pedazo del software necesariamente se eleva dramáticamente con el paso del tiempo. XP reconoce que los proyectos tienen que trabajar para lograr esta reducción de costos y explotar los ahorros una vez que han sido ganados. (8)

Para el desarrollo del presente trabajo de diploma se seleccionó la metodología antes expuesta teniendo en cuenta que la misma se adapta muy bien a las especificaciones y características del proyecto a desarrollar. Entre los elementos que se tuvieron en cuenta para adoptar XP como metodología de desarrollo se encuentran:

- Su capacidad para programar de forma flexible las funcionalidades de la aplicación, respondiendo a las necesidades cambiantes del negocio.
- Está diseñado para trabajar con proyectos de corta duración que pueden ser construidos por equipos de dos a diez programadores.
- El cliente es parte del equipo de desarrollo.
- Se centra en la implementación (codificación) y la poca documentación.

1.5 Tecnología de desarrollo a utilizar

Posterior al análisis de las tendencias y conceptos que sostienen el desarrollo de la aplicación, se procede a realizar una descripción de las tecnologías y herramientas a utilizar, las cuales se encuentran definidas por el centro TLM de la facultad 2 de la Universidad de las Ciencias Informáticas (UCI).

1.5.1 Python 2.7

Python es un lenguaje de programación creado por Guido van Rossum a principios de los años 90. Se trata de un lenguaje interpretado o de script, con tipado dinámico y orientado a objetos. Constituye además la base del marco de trabajo Django. Es multiplataforma, por lo que es compatible en disímiles sistemas operativos. Dentro de sus posibilidades se encuentra el desarrollo web, que es tan importante para el desarrollo de la herramienta.

El intérprete de Python incluye un modo interactivo en el cual se escriben las instrucciones. Las expresiones pueden ser introducidas una a una, pudiendo verse el resultado de su evaluación de forma inmediata, lo que posibilita probar porciones de código en el modo interactivo antes de integrarlo al programa. (9)

1.5.2 PostgreSQL 9.4

PostgreSQL es un sistema de gestión de bases de datos objeto-relacional. Utiliza un modelo cliente/servidor y usa multiprocesos para garantizar la estabilidad del sistema. Un fallo en uno de los procesos no afectará el resto y el sistema continuará funcionando.

Esta versión agrega muchas nuevas características que mejoran la flexibilidad, escalabilidad y rendimiento de PostgreSQL para diferentes tipos de usuarios de bases de datos, incluyendo mejoras al soporte para JSON, replicación y rendimiento de los índices. (10)

La versión 9.4 también introduce varias mejoras de rendimientos. Estas incluyen:

- Mejoras a los índices GIN, haciéndolos hasta 50% más pequeños y hasta 3 veces más rápidos.
- Vistas materializadas actualizables de forma concurrente, para reportes más rápidos y actualizados.
- Recarga rápida del caché de la base de datos en un reinicio usando pg_prewarm.
- Escritura paralela más rápida en el log transaccional de PostgreSQL.

PostgresSQL tiene una gran ventaja respecto al almacenamiento de información permitiendo un espacio de almacenamiento casi ilimitado en sus bases de datos, como se demuestra en la siguiente tabla.

Tabla 1: Límite de almacenamiento de PostgreSQL.

Límite	Valor
Tamaño de base de datos	Ilimitado(Depende del sistema de almacenamiento)
Tamaño de la tabla	32 TB

Tamaño de fila	1.6 TB
Tamaño de campo	1 GB
Número de filas por tabla	Ilimitado
Número de columnas por tabla	250-1600

1.5.3 Django 1.7

Django es un framework [21]de desarrollo web de código abierto, escrito en Python, que respeta el patrón de diseño conocido como Modelo-Vista-Controlador. Fue desarrollado en origen para gestionar varias páginas orientadas a noticias de la World Company de Lawrence, Kansas, y fue liberada al público bajo una Licencia BSD en julio de 2005.(11)

La meta fundamental de Django es facilitar la creación de sitios web complejos. Django pone énfasis en el re-uso, la conectividad y extensibilidad de componentes y el desarrollo. Python es usado en todas las partes del framework, incluso en configuraciones, archivos, y en los modelos de datos.

Django está fuertemente inspirado en la filosofía de desarrollo Modelo-Vista-Controlador, pero sus desarrolladores declaran que, a lo que se llamaría "controlador" en un "verdadero" framework MVC se llama en Django "vista", y lo que se llamaría "vista" se llama "plantilla", por lo que sería Modelo-Plantilla -Vista.

Django posee una alta integración con PostgreSQL. Proporciona una abstracción de la base de datos a través de su API que permite crear, recuperar, actualizar y borrar objetos. También es posible que el usuario ejecute sus propias consultas SQL directamente.

1.5.4 PyCharm 3.0.1

PyCharm es un Entorno de Desarrollo Integrado (IDE) multiplataforma utilizado para desarrollar en el lenguaje de programación Python. Proporciona análisis de código, depuración gráfica y soporte para el desarrollo web con Django, entre otras bondades. PyCharm es desarrollado por la empresa JetBrains y debido a la naturaleza de sus licencias tiene dos versiones, la Community que es gratuita y orientada a la educación y al desarrollo puro en Python y la *Professional*, que incluye más características como el soporte a desarrollo web. (12)

1.5.5 JMeter 2.10

JMeter es un proyecto de Apache que puede ser utilizado como una herramienta de prueba de carga para analizar y medir el desempeño de una variedad de servicios, con énfasis en aplicaciones web. Inicialmente fue diseñada para pruebas de estrés en aplicaciones web, hoy en día, su arquitectura ha evolucionado no sólo para llevar a cabo pruebas en componentes habilitados en Internet (HTTP), sino además en Bases de Datos, requisiciones FTP y prácticamente cualquier otro

[21] Framework, en español: marco de trabajo

medio. Además, posee la capacidad de realizar desde una solicitud sencilla hasta secuencias de requisiciones que permiten diagnosticar el comportamiento de una aplicación en condiciones de producción. En este sentido, simula todas las funcionalidades de un Navegador, o de cualquier otro cliente, siendo capaz de manipular resultados en determinada requisición y reutilizarlos para ser empleados en una nueva secuencia. (13) JMeter fue utilizado el desarrollo del proyecto para realizar las pruebas carga y estrés del sistema.

1.5.6 Acunetix Web Vulnerability Scanner 10.5

Cada año miles de hackers causan estragos a las empresas debido a la vulnerabilidad de los sitios web y sus servidores perimetrales, pudiéndose producir el robo de datos sensibles, de suma importancia para las empresas. Acunetix Vulnerability Scanner (WVS) es una herramienta de seguridad de aplicaciones Web automatizada. Acunetix WVS es capaz de escanear cualquier sitio Web o aplicación Web que es accesible a través del protocolo HTTP / HTTPS. Sin embargo, no todas las pruebas se pueden realizar de forma automática, y por lo tanto Acunetix WVS proporciona herramientas de Penetración manuales para pruebas particulares. Acunetix WVS Comprueba diferentes vulnerabilidades (por ejemplo inyección de SQL, Cross Site Scripting). Hasta la fecha Acunetix comprueba más de 500 tipos diferentes de vulnerabilidades (14). Acunetix fue utilizado para realizar las pruebas de seguridad del sistema.

1.5.7 Visual Paradigm 10.0

Visual Paradigm es una herramienta CASE: Ingeniería de Software Asistida por Computación. La misma propicia un conjunto de ayudas para el desarrollo de programas informáticos, desde la planificación, pasando por el análisis y el diseño, hasta la generación del código fuente de los programas y la documentación. Esta herramienta es utilizada para el modelado de los paquetes del patrón arquitectónico Modelo-Vista-Plantilla. (15)

1.6 Conclusiones parciales.

Con el estudio de las distintas herramientas que realizan el análisis de alertas de posibles fraudes antes expuesto se marca una gran brecha entre los sistemas internacionales investigados (SAP y Mobileum Anti-Fraud Analytics) y el sistema HAT que se encuentra en uso. Los sistemas internacionales desarrollan técnicas de inteligencia artificial, como es el aprendizaje automático y la detección del fraude basado en patrones y reglas, estos sistemas constituyen la avanzada en la gestión del fraude en telefonía tanto fija como móvil en el mundo. Sin embargo, ninguno de estos detecta el fraude basado en datos almacenados en una base de datos, sino que lo detectan en tiempo real durante su ocurrencia en la red. La herramienta HAT, la cual a pesar de que si lo realiza a través de una base de datos se demoran un tiempo considerable en el proceso de gestión de las alertas. Por tanto se llega a la conclusión de que debe desarrollarse una nueva aplicación que sustituya la herramienta HAT que utiliza ETECSA con la ayuda de las tecnologías antes expuestas.

Capítulo 2: Propuesta y planificación del sistema.

2.1 Introducción.

En el presente capítulo se describe la propuesta de solución del sistema, así como sus características y funcionalidades. Se realizará una descripción general de las historias de usuario que se proponen para dar solución a los problemas que originaron la situación problemática. Además, se realizará el plan de las iteraciones y el plan de entrega de las historias de usuario, con el objetivo de determinar el tiempo de desarrollo de la aplicación.

2.2 Propuesta del sistema.

El sistema está diseñado para gestionar todas las alertas que son arrojadas por el sistema Nikira, para su posterior análisis, estas alertas no son más que un conjunto de números telefónicos. Este debe dar la oportunidad de almacenar las alertas en una base de datos, las cuales deben estar disponibles para ser analizadas por los especialistas de ETECSA. También el sistema debe dar la oportunidad de elegir los números telefónicos que se deseen analizar, así como el rango de fecha.

Los usuarios (especialistas de ETECSA), son los encargados de analizar las alertas en cualquier momento. Los especialistas tienen la oportunidad de analizar cualquier número telefónico que se encuentre en la base de datos. También tienen la oportunidad de seleccionar un rango de fecha dentro del cual quieran analizar los números.

El sistema permite realizar un conjunto de consultas a los números que se encuentran dentro del rango de fecha especificado por los especialistas. A partir de estas consultas los especialistas serán los encargados de determinar si las alertas constituyen realmente un hecho de carácter fraudulento, basados en su conocimiento personal.

Finalmente el sistema permitirá al investigador no solo realizar el análisis a través de los grupos de consultas definidas, sino que también le da la posibilidad de exportar el análisis en el formato para la herramienta Microsoft Excel como lo requiere el investigador para un posterior análisis; así como provee al personal del departamento de antifraude de una herramienta para la consulta y gestión tanto de los expediente de los casos de fraude como para la información de los medios de telefonía móvil que circulan en Cuba.

2.2.1 Personas relacionadas con el sistema.

Administrador: Es el encargado de la gestión de los usuarios que interactúan con el sistema.

Especialistas: Son los usuarios creados por el administrador, son los que analizan y clasifican las alertas.

2.2.2 Funcionalidades del sistema.

Tabla 2: Funcionalidades del sistema.

Número	Nombre	Descripción
1	Realizar consultas generales de entradas ➢ Llamadas de servicios de especial interés. ➢ Total de llamadas de servicios de especial interés. ➢ Llamadas de origen fraudulento. ➢ Resumen general de entrada.	Permite obtener datos solicitados en las diferentes consultas generales de entrada.
2	Realizar consultas del resumen general de llamadas de entradas. ➢ Datos de los clientes. ➢ Grupos formados por los clientes. ➢ Llamadas de entrada. ➢ Total de llamadas por servicios. ➢ Resumen general de las llamadas de entrada.	Permite obtener los datos solicitados por las consultas que componen el resumen general de las llamadas de entrada.
3	Realizar consultas de solamente los llamadores de las llamadas de entrada. ➢ Solamente llamadores. ➢ Total de Llamadas por Servicio. ➢ Resumen de los servicios que no presentan relación.	Permite obtener los datos de las llamadas recibidas, de los servicios que no presentan relación con los usuarios llamados.
4	Realizar consulta de las llamadas de LDN [22]recibidas de entrada. ➢ Resumen de las llamadas LDN recibidas.	Permite obtener los datos de las llamadas de larga distancia nacional recibidas. Los datos obtenidos son de las llamadas que entran.
5	Realizar consulta de las llamadas de LDI [23]recibidas, de entrada.	Permite obtener los datos de las llamadas de larga distancia internacional

[22] LDN: larga distancia nacional

[23] LDI: larga distancia internacional

	➢ Resumen de las llamadas LDI recibidas.	recibidas, de las llamadas de entradas.
6	Realizar consulta de las llamadas con duración menor que 3 minutos. ➢ Resumen de las llamadas con duración menor que 3 minutos.	Permite obtener los datos de las llamadas recibidas con una duración menor que 3 minutos.
7	Realizar consultas que se originan de los móviles. ➢ Llamadas realizadas por los móviles. ➢ Llamadas recibidas por los móviles. ➢ Resumen de las llamadas que se originan de los móviles.	Permite obtener los datos de los móviles que llaman a diferentes usuarios y los móviles que reciben llamadas.
8	Realizar consultas para el fraude del servidor residencial. ➢ Llamadas recibidas con duración menor que 3 minutos. ➢ Llamadas con duración menor que 3 minutos por servicios. ➢ Clientes potenciales. ➢ Calculo de pérdidas. ➢ Resumen del fraude del servidor residencial.	Permite obtener los datos de los fraudes del servidor residencial donde se encuentran los clientes potenciales y se calculan las pérdidas.
9	Realizar consultas para el fraude de ByPass de salida. ➢ Resumen del fraude de ByPass de salida.	Permite obtener los datos de los fraudes de ByPass de salida, así como los datos de los clientes de ByPass.
10	Realizar consultas generales de salidas. ➢ Servicios que presentan una estrecha relación. ➢ Llamadas realizadas a los servicios de especial interés.	Permite obtener datos solicitados en las diferentes consultas generales de salida.

	➢ Total de llamadas realizadas a los servicios de especial interés. ➢ Llamadas realizadas a destinos fraudulentos. ➢ Resumen general de salida.	
11	Realizar consultas del resumen general de llamadas de salida. ➢ Datos. ➢ Grupos. ➢ Llamadas de salida. ➢ Total de llamadas por servicio. ➢ Resumen general de las llamadas de salida.	Permite obtener los datos solicitados por las consultas que componen el resumen general de las llamadas de salida.
12	Realizar consultas de los llamadores, de las llamadas de salida. ➢ Solamente los llamados. ➢ Total de llamadas por servicio. ➢ Resumen de los usuarios llamados.	Permite obtener los datos de las llamadas recibidas de los servicios que no presentan relación, con los usuarios llamados.
13	Realizar consultas a las llamadas mayores de 3 minutos. ➢ Llamadas realizadas con una duración mayor que 3 minutos a todos los grupos. ➢ Total de llamadas realizadas con una duración mayor que 3 minutos por servicio. ➢ Resumen de las llamadas con duración mayor que 3 minutos.	Permite obtener los datos de las llamadas realizadas que tengan una duración mayor que 3 minutos a los servicio de todos los grupos de suscriptores con los que no presentan relación.
14	Realizar consultas a las llamadas realizadas mediante el empleo de tarjetas propias. ➢ Llamadas realizadas utilizando tarjetas propias.	Permite obtener los datos de las llamadas realizadas usando tarjetas propias y la cantidad de llamadas de las mismas.

	➢ Cantidad de llamadas y recargas por tarjetas. ➢ Resumen de las llamadas mediante el empleo de tarjetas propias.	
15	Realizar consulta de las llamadas de LDN recibidas, de salida. ➢ Resumen de las llamadas de LDN realizadas.	Permite obtener los datos de las llamadas de larga distancia nacional recibidas, de las llamadas de salida.
16	Realizar consulta de las llamadas de LDI recibidas, de salida. ➢ Resumen de las llamadas de LDI realizadas.	Permite obtener los datos de las llamadas de larga distancia internacional recibidas, de las llamadas de salida.
17	Realizar consultas a las conexiones de ISP. ➢ Conexiones a ISP. ➢ Total de conexiones por días. ➢ Resumen de las conexiones a ISP.	Permite obtener los datos de conexiones a los (Proveedores de servicio de internet) ISP y la cantidad de veces que se conectan por día.
18	Realizar consultas que están destinadas a los móviles. ➢ Llamadas realizadas a los móviles. ➢ Llamadas recibidas por los móviles. ➢ Resumen de las llamadas destinadas a los móviles.	Permite obtener las llamadas que se le realizaron a los móviles
19	Realizar consultas específicas del fraude en el Cyber Café. ➢ Cálculos de pérdidas. ➢ Resumen del cálculo de las pérdidas.	Permite obtener los datos de los fraudes ocurridos en el Cyber Café y calcular las pérdidas.

20	Realizar consultas para el fraude de Bypass de entrada. ➢ Llamadas con duración mayor que 3 minutos. ➢ Total de llamadas con duración mayor que 3 minutos. ➢ Resumen de las llamadas con duración mayor que 3 minutos. ➢ Llamadas con duración mayor que 10 minutos. ➢ Llamadas con duración mayor que 20 minutos. ➢ Total de llamadas realizadas a cliente ByPass con duración mayor que 3 minutos. ➢ Resumen de las llamadas realizadas a cliente ByPass con duración mayor que 3 minutos. ➢ Cantidad de días trabajados. ➢ Cálculos de pérdidas. ➢ Resumen del fraude de ByPass de entrada.	Permite obtener los datos de los fraudes de Bypass de entrada, así como los datos de los clientes Bypass.
21	Gestionar los registros de medios. ➢ Insertar medio ➢ Eliminar medio ➢ Modificar medio	Permite añadir, eliminar o modificar algunas características de los teléfonos móviles, como: el IMEI, la marca y el modelo.
22	Gestionar la base de datos de GAF. ➢ Insertar caso ➢ Eliminar caso ➢ Modificar caso	Permite añadir, eliminar o modificar casos relacionados con el fraude.
23	Gestionar los números a analizar. ➢ Insertar números ➢ Eliminar números ➢ Modificar números	Permite añadir, eliminar o modificar cualquier número que se quiera analizar.

24	Gestionar el rango de la fecha a analizar. ➢ Insertar fecha inicial ➢ Insertar fecha final ➢ Eliminar fecha inicial ➢ Eliminar fecha final ➢ Modificar fecha inicial	Permite escoger la fecha inicial y la fecha final de los números que se deseen analizar.
25	Exportar a Excel los resultados de las consultas	Permite exportar los resultados de las consultas a un módulo Excel, con el fin de poder analizarlos posteriormente.

2.2.3 Características no funcionales del sistema

1. Apariencia e interfaz externa.

- **Interfaz de usuario**

La información en esta interfaz, está organizada por un menú principal, el cual tiene un conjunto de opciones a escoger, las cuales a su vez se derivan en varias consultas generales y estas en una serie de consultas específicas donde se mostrarán todos los datos solicitados para cada registro de las llamadas telefónicas en específico.

- **Interfaz de hardware**

Se necesita un puerto de red compatible con un cable RJ45.

2. Usabilidad

La estructura de botones y vínculos del menú principal debe de estar organizada según la funcionalidad que le corresponde así como a los submenús que despliegan los mismos, con el objetivo de facilitar al usuario la interacción con la herramienta. Los mensajes para interactuar con los usuarios y los de error deben ser lo suficientemente informativos, en idioma español y no deben revelar información interna.

3. Rendimiento

El sistema debe responder lo más rápido posible ante las solicitudes de información por parte de los especialistas de ETECSA. La eficiencia de la aplicación estará determinada en gran medida por la velocidad a las consultas a la base de datos y las características de la computadora donde se encuentra ejecutándose la aplicación.

4. Portabilidad

El sistema debe ser empleado en plataforma Windows y Linux. Para su correcto funcionamiento. Las diferentes plataformas donde sea instalado o las diferentes computadoras deben contar con las características necesarias para trabajar con bases de datos y el Framework de desarrollo Django

así como soportar la versión 2.7 del intérprete de python junto con todas las dependencias necesarias.

5. Seguridad

El sistema cuenta con técnicas de cifrado de contraseñas para proteger la integridad de los datos que son manejados por los usuarios que en él operen. Cada usuario del sistema tiene un sesión única para mantener los principios de confidencialidad, donde solo el podrá acceder a su sesión mediante su contraseña personal. Además el sistema se encontrará disponible en todo momento para todos los usuarios que tenga los permisos requeridos.

6. Disponibilidad

La aplicación debe estar disponible en todo momento para los usuarios autorizados que necesiten acceder y manejar la información contenida en la misma.

7. Soporte

El sistema cuenta con un manual de ayuda, con el objetivo de explicar cómo funciona la gestión y el análisis de las alertas con los que trabajan los especialistas. También explica cómo se debe proceder para la gestión y análisis de las alertas.

8. Software

Para el usuario: El sistema debe operar sobre los sistemas operativos Windows 7 o Linux Debian 6. Para la utilización del sistema se requerirá el uso de un navegador web Mozilla Firefox o el Crome.

Para el servidor: En cuanto a los requerimientos para el servidor se recomienda el gestor de base de datos PostgreSQL 9.4, el intérprete de python en su versión 2.7, el framework Django en su versión 1.7, las librerías y dependencias necesarios para su funcionamiento y los sistemas operativos Windows o Linux.

9. Hardware

Para la instalación del sistema se debe tener una computadora con las siguientes características:

- RAM: 1GB o superior.
- Disco Duro: 20GB o superior.
- CPU: Dual Core a 2.0GHz o superior.
- 1 puerto de red, compatible con cable RJ45.

2.3 Etapa de Planificación.

La planificación es la primera etapa definida en la metodología XP. En esta etapa el cliente define a grandes rasgos lo que necesita mediante las historias de usuarios, las cuales son de gran interés para la entrega del producto y posibilita a los programadores tener un tiempo estimado de cuánto va a demorar el desarrollo de la aplicación. Al mismo tiempo que el equipo de desarrollo se familiariza con las herramientas, tecnologías y prácticas que se utilizan para el desarrollo de la herramienta.

2.3.1 Historias de usuarios.

Las historias de usuario son la técnica utilizada en XP para especificar las funcionalidades del sistema, brindan detalles sobre la estimación del riesgo y cuánto tiempo será empleado en su implementación. El cliente es el encargado de asignarle una prioridad a cada HU[24] y es el equipo de desarrollo el encargado de asignarle un costo, este se traduce en las semanas que llevará el desarrollo de las mismas. Si las HU según lo planificado demoran en desarrollarse, se sugiere dividirla en HU más pequeñas. También, es importante destacar que las HU nuevas pueden describirse en cualquier momento, con esto se comprueba la flexibilidad de la metodología.

Las HU serán representadas mediante tablas divididas por las siguientes secciones:

- **Número:** número de la historia de usuario incremental en el tiempo.
- **Nombre de historia de usuario:** el nombre de la historia de usuario sería para identificarlas mejor entre los desarrolladores y el cliente.
- **Modificación de historia de usuario número:** si sufrió alguna modificación anterior.
- **Usuario:** es el usuario que está involucrado en el desarrollo de la HU.
- **Iteración asignada:** número de la iteración.
- **Prioridad en negocio:**
  - Las historias de usuarios que son de funcionalidades imprescindibles en el desarrollo del sistema tienen prioridad alta.
  - Las historias de usuarios que son de funcionalidades que debe de tener el sistema, pero que no son necesarias para su funcionamiento, tienen prioridad media.
  - Las historias de usuarios que son de funcionalidades auxiliares y que son independientes del sistema, tienen prioridad baja.
- **Riesgo en desarrollo:**
  - Las historias de usuarios que, en caso de tener algún error de implementación, puedan afectar la disponibilidad del sistema, tienen riesgo de desarrollo alto.
  - Las historias de usuarios que puedan presentar errores y retrasan la entrega de la versión, tienen riesgo de desarrollo medio.
  - Las historias de usuario que puedan presentar errores, pero estos son tratados con facilidad y no afectan en desarrollo del proyecto, tienen riesgo de desarrollo bajo.
- **Puntos estimados**: tiempo estimado que se demorará el desarrollo de la HU.
- **Puntos reales**: tiempo que se demoró en realidad el desarrollo de la HU.
- **Descripción**: breve descripción de la HU.
- **Observaciones**: señalamiento o advertencia del sistema.
- **Prototipo de interfaz**: Prototipo de interfaz si aplica (16).

[24] HU: historia de usuario

Tabla 3: Realizar consultas generales de entrada.

Historia de Usuario	
Número: 1	**Nombre:** Realizar consultas generales de entrada.
Modificación de historia de usuario: Ninguna	
Usuario: Alexander Valdés Molina	**Iteración asignada:** 1
Prioridad en negocio: Media	**Puntos estimado:** 4/5
Riesgo en desarrollo: Medio	**Puntos reales:** 4/5
Descripción: Permite obtener datos solicitados en las diferentes consultas generales de entrada como son: • Llamadas de servicios de especial interés. • Total de llamadas de servicios de especial interés. • Llamadas de origen fraudulento. • Resumen general de entrada.	
Observaciones: El usuario debe estar correctamente autenticado en el sistema.	
Prototipo de Interfaz: 	

Tabla 4: Realizar consultas del resumen general de llamadas de entrada.

Historia de Usuario	
Número: 2	**Nombre:** Realizar consultas del resumen general de llamadas de entradas.
Modificación de historia de usuario: Ninguna	
Usuario: Alexander Valdés Molina	**Iteración asignada:** 1
Prioridad en negocio: Media	**Puntos estimado:** 1
Riesgo en desarrollo: Medio	**Puntos reales:** 1

Descripción: Permite obtener los datos solicitados por las consultas que componen el resumen general de las llamadas de entrada, las cuales son: • Datos de los clientes. • Grupos formados por los clientes. • Llamadas de entrada. • Total de llamadas por servicios. • Resumen general de las llamadas de entrada.
Observaciones: El usuario debe estar correctamente autenticado en el sistema.
Prototipo de Interfaz: 

Tabla 5: Realizar consultas de las llamadas LDN recibidas de entrada.

Historia de Usuario	
Número: 4	**Nombre:** Realizar consulta de las llamadas de LDN recibidas de entrada.
Modificación de historia de usuario: Ninguna	
Usuario: Alexander Valdés Molina	**Iteración asignada:** 1
Prioridad en negocio: Media	**Puntos estimado:** 1/5
Riesgo en desarrollo: Medio	**Puntos reales:** 1/5
Descripción: Permite obtener los datos de las llamadas de larga distancia nacional recibidas, así como un resumen de las mismas. Los datos obtenidos son de las llamadas que entran.	
Observaciones: El usuario debe estar correctamente autenticado en el sistema.	

Prototipo de Interfaz:

Tabla 6: Gestionar los números a analizar.

Historia de Usuario	
Número: 23	**Nombre:** Gestionar los números a analizar.
Modificación de historia de usuario: Ninguna	
Usuario: Alexander Valdés Molina	**Iteración asignada:** 3
Prioridad en negocio: Alta	**Puntos estimado:** 4/5
Riesgo en desarrollo: Alto	**Puntos reales:** 4/5
Descripción: Permite añadir, eliminar o modificar cualquier número que se quiera analizar.	
Observaciones: El usuario debe estar correctamente autenticado en el sistema. Los números deben estar registrados en la base de datos.	
Prototipo de Interfaz:	

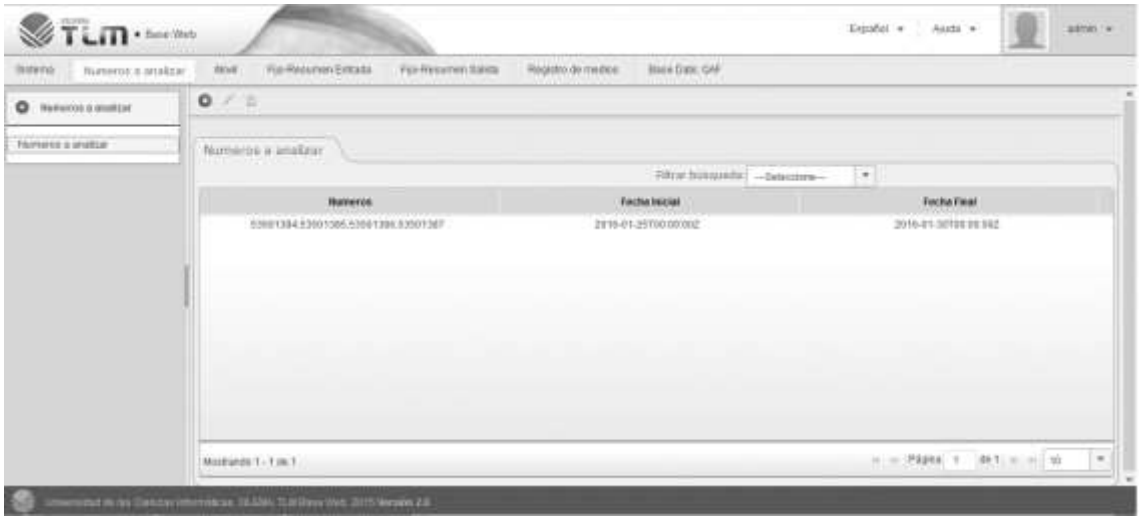
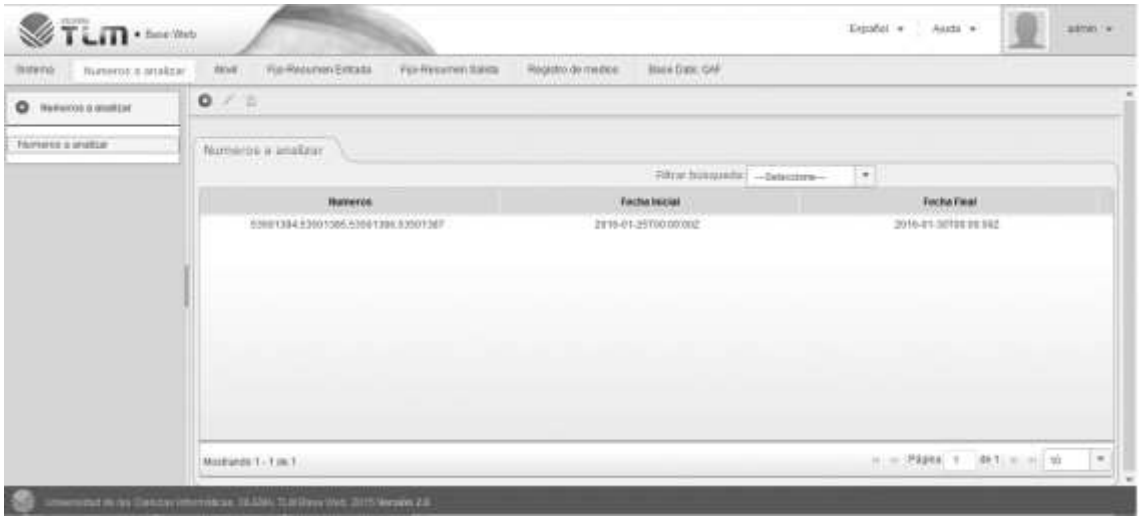


Tabla 7: Gestionar el rango de la fecha a analizar.

Historia de Usuario

Número: 24	**Nombre:** Gestionar el rango de la fecha a analizar.
Modificación de historia de usuario: Ninguna	
Usuario: Alexander Valdés Molina	**Iteración asignada:** 3
Prioridad en negocio: Alta	**Puntos estimado:** 3/5
Riesgo en desarrollo: Alto	**Puntos reales:** 3/5
Descripción: Permite escoger la fecha inicial y la fecha final de los números que se deseen analizar.	
Observaciones: El usuario debe estar correctamente autenticado en el sistema. La fecha debe tener un formato valido.	
Prototipo de Interfaz: 	

2.3.2 Plan de las Iteraciones.

Una iteración es un período de tiempo que se ejecuta dentro del proyecto, en el cual se produce una versión ejecutable de la aplicación. El plan de iteraciones incluye varias iteraciones sobre la herramienta antes de ser entregado finalmente. Todo el trabajo que se realiza en las iteraciones es descrito en las tareas de ingeniería. Luego que se analizaran todos los requisitos definidos por el cliente y se detallaran las historias de usuario, se estableció una división de cuatro iteraciones.

- **Iteración 1.**

  En la primera iteración se llevará a cabo el desarrollo de las historias de usuario desde el número 1 hasta el número 7, donde se gestionan parte de las consultas de las llamadas de entrada. Al terminar la iteración esto representará un 22 % de la implementación de la aplicación.

- **Iteración 2.**

  En la segunda iteración se llevará a cabo el desarrollo de las historias de usuario, desde el número 8 hasta el número 11, donde se gestionan parte de las consultas de las llamadas de

entrada y salida. Al terminar la iteración esto representará un 47 % de la implementación de la aplicación.

- **Iteración 3.**

  En la tercera iteración se llevará a cabo el desarrollo de las historias de usuario desde el número 12 hasta el número 19, donde se gestionan parte de las consultas de las llamadas de salida. Al terminar la iteración esto representará un 73 % de la implementación de la aplicación.

- **Iteración 4.**

  En la tercera iteración se llevará a cabo el desarrollo de las historias de usuario desde el número 20 hasta el número 25, donde se gestionarán las características relacionadas con los teléfonos móviles. Además, se gestionará la información relacionada con la base de datos de GAF y los números y el rango de la fecha a analizar

2.3.3 Plan de duración de las iteraciones.

En el plan de duración de las iteraciones se mostrará el orden en que deben implementarse las historias de usuario, en que iteración se realizara cada una y la duración de cada iteración.

Tabla 8: Plan de duración de las iteraciones.

Iteración	Orden de las Historias de Usuario a implementar	Duración total
1	➢ Realizar consultas generales de entrada. ➢ Realizar consultas del resumen general de llamadas de entradas. ➢ Realizar consultas de solamente los llamadores de las llamadas de entrada. ➢ Realizar consulta de las llamadas de LDN recibidas de entrada. ➢ Realizar consulta de las llamadas de LDI recibidas de entrada. ➢ Realizar consulta de las llamadas con duración menor que 3 minutos ➢ Realizar consultas que se originan de los móviles.	4 semanas
2	➢ Realizar consultas para el fraude de servidor residencial. ➢ Realizar consultas para el fraude de ByPass de salida. ➢ Realizar consultas generales de salidas.	4 semanas

	➢ Realizar consultas del resumen general de llamadas de salida.	
3	➢ Realizar consultas de solamente los llamadores de las llamadas de salida. ➢ Realizar consultas a las llamadas mayores de 3 minutos. ➢ Realizar consultas a las llamadas realizadas mediante el empleo de tarjetas propias. ➢ Realizar consulta de las llamadas de LDN recibidas de salida. ➢ Realizar consulta de las llamadas de LDI recibidas de salida. ➢ Realizar consultas a las conexiones de ISP ➢ Realizar consultas que están destinadas a los móviles ➢ Realizar consultas específicas del fraude en el Cyber Café.	4 semanas
4	➢ Realizar consultas para el fraude de ByPass de entrada. ➢ Gestionar los registros de medios ➢ Gestionar la base de datos de GAF ➢ Gestionar los números a analizar ➢ Gestionar el rango de la fecha a analizar ➢ Exportar a Excel los resultados de las consultas	3 semanas

2.3.4 Plan de entrega.

Luego de elaborar el plan de duración de las iteraciones, se confecciona el plan de entrega, donde se estima el tiempo de desarrollo de las historias de usuario, para así definir cuanto demorará la implementación de cada historia de usuario y estimar el día de entrega de la aplicación.

Tabla 9: Plan de entrega de las iteraciones

Herramienta	Iteración 1	Iteración 2	Iteración 3	Iteración 4
Sistema de gestión y análisis de trafico	24/02/2016	24/03/2016	19/04/2016	18/05/2016

2.5 Conclusiones parciales.

En el presente capítulo se describió la propuesta de solución del sistema a desarrollar. Se realizó una descripción general de las historias de usuarios propuestas y los requisitos, con el objetivo de definir las condiciones y capacidades que deben estar presentes en la aplicación, para así satisfacer las necesidades del cliente. A partir de las Historias de usuario descritas, se definieron 3 iteraciones, así como el tiempo estimado para la realización de cada una. Se definió además el plan de entrega de las iteraciones que permitió delimitar el ciclo de desarrollo de la aplicación.

Capítulo 3: Implementación y pruebas

3.1 Introducción.

Para el desarrollo de toda aplicación es de vital importancia la fase de implementación y prueba del sistema, para dar cumplimiento a los objetivos planteados. Es por esto, que en el presente capítulo se describen las tareas de ingeniería generadas por las historias de usuario, como base para la implementación del software. También se realizarán un conjunto de pruebas para evaluar la calidad de la aplicación.

3.2 Tareas de ingeniería.

Las historias de usuario son descompuestas en tareas de la ingeniería y asignadas a los programadores para ser implementadas durante una iteración. Las tareas de la ingeniería serán representadas mediante tablas divididas por las siguientes secciones:

- **Número tarea:** los números deben ser consecutivos.
- **Número historia de usuario:** número de la historia de usuario a la que pertenece la tarea.
- **Nombre tarea:** nombre que identifica a la tarea.
- **Tipo de tarea:** las tareas pueden ser de: Desarrollo, Corrección, Mejora, Otra.
- **Puntos estimados:** tiempo estimado en semanas que se le asignará a su desarrollo.
- **Fecha inicio:** fecha en que inicia el desarrollo de la tarea.
- **Fecha fin**: fecha en que finaliza el desarrollo de la tarea
- **Programador responsable:** nombre y apellidos del programador.
- **Descripción:** breve descripción de la tarea (16).

Tabla 10: Tarea de ingeniería #1: Llamadas de servicios de especial interés.

Tarea de ingeniería	
Número de tarea: 1	**Número de historia de usuario:** 1
Nombre de tarea: Llamadas de servicios de especial interés.	
Tipo de tarea: Desarrollo	**Puntos estimados:** 1/5
Fecha inicio: 1/02/2016	**Fecha fin:** 1/02/2016
Programador responsable: Alexander Valdés Molina y Braiman González Sánchez	
Descripción: De las consultas generales, muestra los datos de las llamadas de servicios de especial interés solicitados por el usuario.	

Tabla 11: Tarea de ingeniería #2: Total de llamadas de servicios de especial interés.

Tarea de ingeniería	
Número de tarea: 2	**Número de historia de usuario:** 1
Nombre de tarea: Total de llamadas de servicios de especial interés.	
Tipo de tarea: Desarrollo	**Puntos estimados:** 1/5

Fecha inicio: 2/02/2016	**Fecha fin:** 2/02/2016
Programador responsable: Alexander Valdés Molina y Braiman González Sánchez	
Descripción: De las consultas generales muestra el total de las llamadas de servicio de especial interés solicitadas por el usuario.	

Tabla 12: Tarea de ingeniería #3: Llamadas de origen fraudulento.

Tarea de ingeniería	
Número de tarea: 3	**Número de historia de usuario:** 1
Nombre de tarea: Llamadas de origen fraudulento y resumen de las consultas generales.	
Tipo de tarea: Desarrollo	**Puntos estimados:** 1/5
Fecha inicio: 3/02/2016	**Fecha fin:** 3/02/2016
Programador responsable: Braiman González Sánchez y Alexander Valdés Molina	
Descripción: De las consultas generales muestra todas las llamadas que se originan de los números telefónicos que han cometido fraudes.	

Tabla 13: Tarea de ingeniería #8: Llamadas de LDN recibidas.

Tarea de ingeniería	
Número de tarea: 10	**Número de historia de usuario:** 4
Nombre de tarea: Llamadas de LDN recibidas.	
Tipo de tarea: Desarrollo	**Puntos estimados:** 1/5
Fecha inicio: 16/02/2016	**Fecha fin:** 16/02/2016
Programador responsable: Alexander Valdés Molina y Braiman González Sánchez	
Descripción: Muestra los registros de las llamadas de larga distancia nacional, recibidas por los números telefónicos que han cometido fraude.	

Tabla 14 Tarea de ingeniería # 56. Gestionar los números a analizar

Tarea de ingeniería	
Número de tarea: 56	**Número de historia de usuario:** 23
• **Nombre de tarea:** Gestionar los números a analizar.	
Tipo de tarea: Desarrollo	**Puntos estimados:** 3/5
Fecha inicio: 09/05/2016	**Fecha fin:** 11/05/2016
Programador responsable: Braiman González Sánchez y Alexander Valdés Molina	
Descripción: Permite añadir, modificar o borrar un número telefónico que sea de especial interés para los investigadores. El adicionar un número, permite realizar una serie de consultas a estos números.	

Tabla 15: Tarea de ingeniería # 56. Gestionar el rango de la fecha a analizar.

Tarea de ingeniería	
Número de tarea: 57	**Número de historia de usuario:** 24
• **Nombre de tarea:** Gestionar el rango de la fecha a analizar.	
Tipo de tarea: Desarrollo	**Puntos estimados:** 2/5
Fecha inicio: 12/05/2016	**Fecha fin:** 13/05/2016
Programador responsable: Braiman González Sánchez y Alexander Valdés Molina	
Descripción: Permite añadir, modificar o borrar un rango de fecha, dentro del cual se encuentran los números que se deseen analizar.	

3.3 Arquitectura del sistema

Una arquitectura es un entramado de componentes funcionales que, aprovechando diferentes estándares, convenciones, reglas y procesos, permite integrar una amplia gama de productos y servicios informáticos, de manera que pueden ser utilizados eficazmente dentro de una organización.

La arquitectura del sistema permite la comunicación entre las partes interesadas en el desarrollo del sistema. Además, resalta las primeras decisiones que tendrán un efecto profundo en todo el proceso de desarrollo. Constituye un modelo de cómo está estructurado el sistema y la forma en que interactúan sus componentes.

En el desarrollo del sistema se utilizará una arquitectura cliente-servidor teniendo en cuenta que esta posee las características necesarias para la implementación de la aplicación. (16)

3.3.1 Arquitectura cliente-servidor

La arquitectura cliente-servidor es un modelo de aplicación distribuida para el desarrollo de sistemas informáticos, en el que las tareas se reparten entre los proveedores de recursos o servicios, llamados servidores y los demandantes llamados clientes. Los clientes y los servidores pueden estar conectados a una red local o una red amplia, como la que se puede implementar en una empresa o a una red mundial como lo es la Internet. Un cliente realiza peticiones a otro programa, el servidor. El servidor contiene la información que debe ser compartida y es el encargado de dar respuesta a las peticiones del cliente.

Ilustración 2: Arquitectura cliente-servidor

Para el desarrollo del sistema de gestión y análisis de alertas, se emplea la arquitectura cliente-servidor por lo que cada usuario podrá acceder a la aplicación y realizar peticiones al servidor mediante su pc cliente utilizando su navegador para realizar dichas peticiones. Permitiendo así el acceso a los datos a los especialistas de ETECSA. El usuario inserta los números a analizar y el rango de la fecha de esos números a analizar, enviándole así una petición al servidor. El servidor en respuesta de esto, cargar en la base de datos un registro de los números solicitados. Luego el usuario selecciona la consulta deseada, mostrándole así los registros de los números (18).

3.3.2 Patrón arquitectónico.

Un patrón arquitectónico de software representa un diseño organizativo estructural, fundamental para guiar el desarrollo del software. Proporcionando un conjunto de sub-sistemas predefinidos, especificando sus responsabilidades, reglas, directrices que determinan la organización, comunicación, interacción y relaciones entre ellos. Dentro de los patrones arquitectónicos podemos encontrar el Modelo-Vista-Controlador (MVC). En el framework Django usa una modificación llamada Model-Template-View (MTV) que sería Modelo-Plantilla-Vista empleado en la implementación de da solución. El *Modelo* en Django se le sigue llamando *Modelo*, pero a la *Vista* se le llama *Template* y el *Controlador* se le llama *View.*

En este patrón, el "Modelo" hace referencia al acceso a la capa de datos, la "Vista" se refiere a la parte del sistema que selecciona qué mostrar y cómo mostrarlo, y el "Controlador" implica la parte del sistema que decide qué vista usar, dependiendo de la entrada del usuario, accediendo al modelo si es necesario.

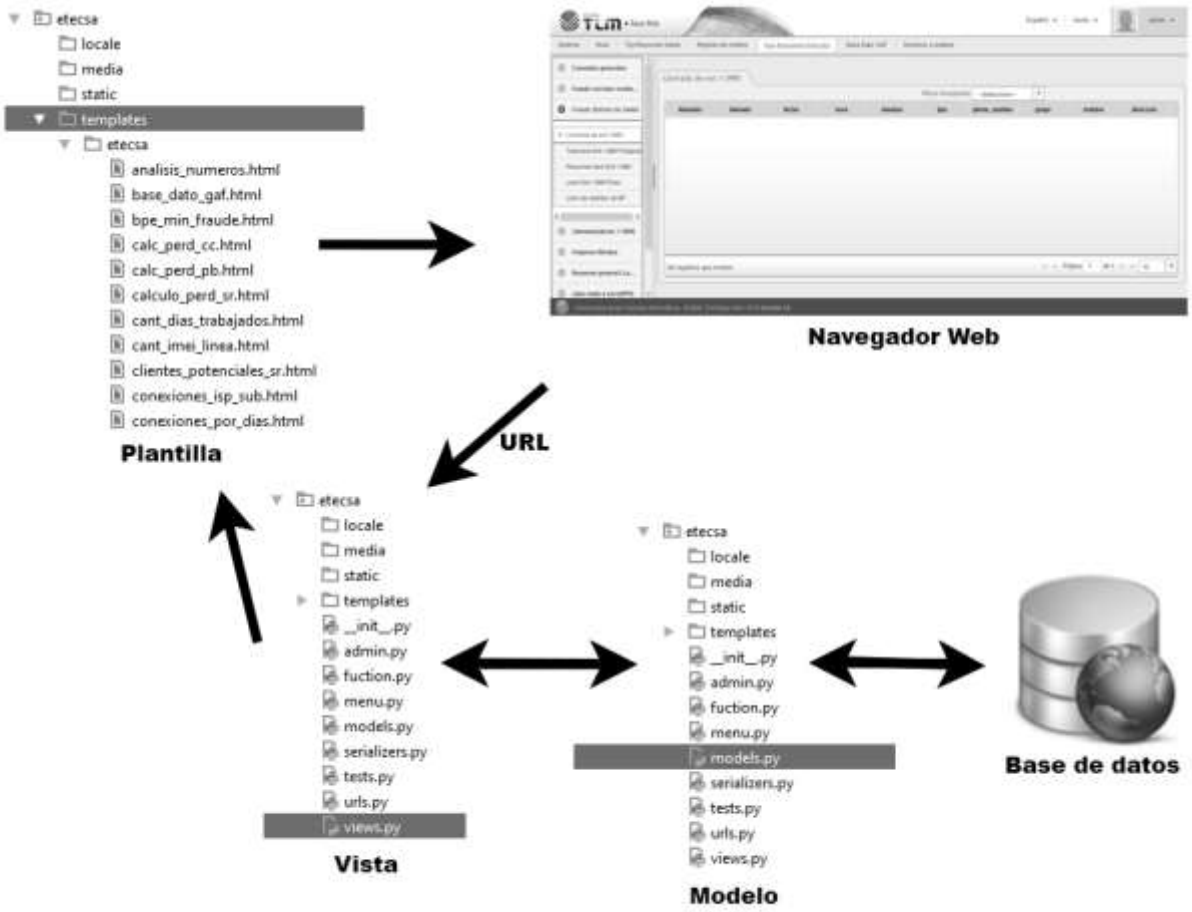
La capa modelo (models): En este patrón, el "Modelo" hace referencia al acceso a la capa de datos. Esta capa contiene toda la información sobre los datos: cómo acceder a estos, cómo validarlos, cuál es el comportamiento que tiene, y las relaciones entre los datos.

La capa plantilla (templates): Es la capa de presentación. Esta capa contiene las decisiones relacionadas a la presentación: como son mostradas algunas cosas sobre una página web.

La capa vista (views): Es la capa de la lógica de negocios. Esta capa contiene la lógica que accede al modelo y la delega a la plantilla apropiada: puedes pensar en esto como un puente entre la modelo y la plantilla.

Como parte del funcionamiento del patrón MTV, primeramente el navegador manda una solicitud a la vista (views), la urls que permite controlar el despliegue de las vistas interpreta la solicitud y luego la vista interactúa con el modelo (models) para obtener los datos, después hace una llamada a la plantilla (templates) y la plantilla se encarga de renderizar la respuesta a la solicitud del navegador. A continuación se muestra en la ilustración 3 dicha relación. (19)

Ilustración 3.1: Patrón de arquitectura Modelo-Plantilla-Vista



.

3.4 Representación de las capas de la arquitectura

A continuación se describen las clases situadas en cada una de las capas de la arquitectura del patrón MTV. Dicha descripción no es exigida por la metodología seleccionada. La misma se realiza para un mejor entendimiento del patrón arquitectónico implementado.

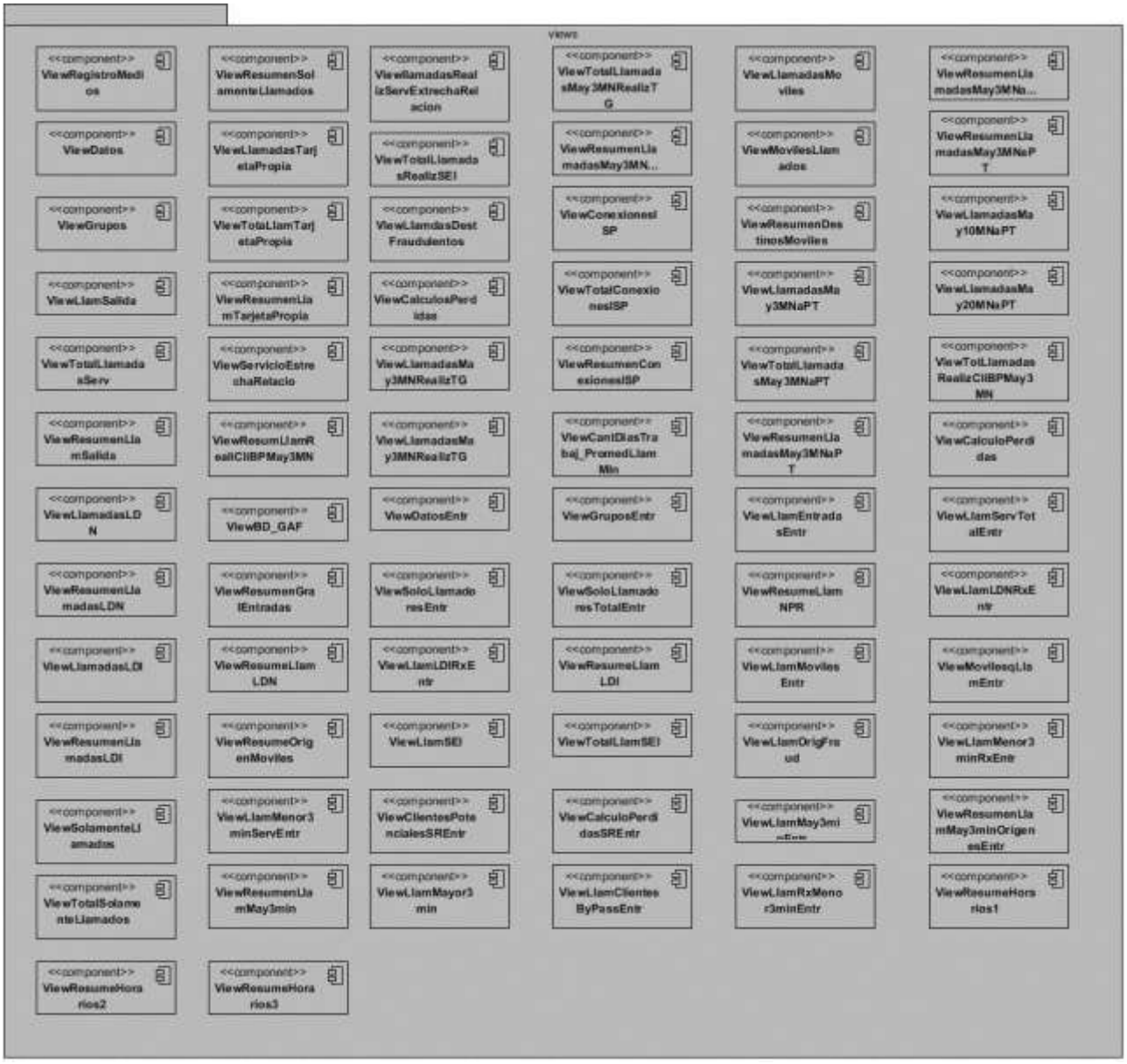
3.4.1 Capa modelo

Ilustración 4.2: Representación de la capa modelo.

Esta capa se encuentra conformada en su mayoría por los modelos que guardan los resultados de las consultas. Además de los modelos bd_gaf, cdr, clientebp, isp y subscriber que son la fuente de datos a donde se realizar mismas.

3.4.2 Capa vista

Ilustración 5.3: Representación de la capa vista.



Esta capa se encuentra conformada por los grupos de consultas para el análisis de las alertas de posibles fraudes.

3.4.3 Capa plantilla

Ilustración 6.3: Representación de la capa plantilla.

Esta capa se encuentra conformada en su mayoría por las plantillas que muestran mediante una tabla los datos de las consultas realizadas. Además se encuentran las plantillas RegistroMedios.html y BD_GAF.html que muestran mediante una tabla la relación de los medios móviles del país y de los expedientes de casos de fraude respectivamente.

3.5 Tarjetas Clase-Responsabilidad-Colaboración (CRC)

La utilización de tarjetas CRC es una técnica de diseño orientado a objetos. El objetivo de la misma es hacer, mediante tarjetas, un inventario de las clases que vamos a necesitar para implementar el sistema y la forma en que van a interactuar, de esta forma se pretende facilitar el análisis y discusión de las mismas por parte de varios actores del equipo de proyecto con el objeto de que el diseño sea lo más simple posible verificando las especificaciones del sistema (17).

Tabla 16: Tarjeta CRC ViewGrupos.

Clase: ViewGrupos	
Responsabilidad	**Colaboración**
➢ Borra todos los registros anteriores de los grupos. ➢ Carga los registros de los grupos. ➢ Muestra los registros solicitados de los grupos.	Grupos, NumerosAnalisisProcesar, CDR, GrupoSerializers.

Tabla 17: Tarjeta CRC ViewDatos.

Clase: ViewDatos	
Responsabilidad	**Colaboración**
➢ Borra todos los registros anteriores de los datos de los clientes. ➢ Carga los registros de los datos de los clientes. ➢ Muestra los registros solicitados de los datos de los clientes.	Datos, NumerosAnalisisProcesar, CDR, SUBSCRIBER, DatosSerializers.

Tabla 18: Tarjeta CRC ViewLlamSalida.

Clase: ViewLlamSalida	
Responsabilidad	**Colaboración**
➢ Borra todos los registros anteriores de las llamadas de salida. ➢ Carga los registros de los números a analizar. ➢ Muestra los registros solicitados de las llamadas de salida.	llamadasSalida, NumerosAnalisisProcesar, CDR, llamadasSalidaSerializers.

3.6 Patrones de diseño

Los patrones de diseño son principios generales de soluciones que aplican ciertos estilos que ayudan a la creación de software. Es una descripción de un problema y la solución a la que le da el nombre y que se puede aplicar en nuevos contextos. Muchos patrones ayudan a asignar responsabilidades a los objetos (23). Dentro de los patrones de diseño se encuentran los patrones

Generales de Software para Asignación de Responsabilidades (GRASP[25] por sus siglas en ingles) y los patrones Banda de los Cuatro (GOF[26] por sus siglas en ingles).

3.6.1 Patrones GRASP

Los patrones GRASP describen los principios fundamentales de la asignación de responsabilidades a objetos, expresados en formas de patrones. El nombre se eligió para indicar la importancia de captar estos principios, si se quiere diseñar eficazmente el software orientado a objetos.

Los patrones GRASP, son parejas de problema-solución con un nombre, que codifican buenos principios y sugerencias relacionados frecuentemente con la asignación de responsabilidades. (20)

3.6.1.1 Experto

El Patrón Experto consiste en asignar una responsabilidad al experto en información: la clase que cuenta con la información necesaria para cumplir la responsabilidad.

Este patrón se ve reflejado en la aplicación, mediante la declaración de los atributos y funciones destinadas al trabajo de las clases. Este indica que la responsabilidad de la creación de las tablas de la base de datos, debe recaer sobre la clase que conoce toda la información necesaria para crearlo.

Ejemplo: Las clases *BD_GAF*, *SUBSCRIBER* y *CDR* cuentan con la información necesaria para dar cumplimiento a cada una de las responsabilidades que les corresponden, las mismas serán las que manejen la información de los registros de las llamadas telefónicas como se muestra a continuación en la Ilustración 3.5.

[25] GRASP: Patrones de Software para la asignación General de Responsabilidad, General Risponsibility Asignment Software Patterns.

[26] GOF: Gand-of-Four (“Pandilla de los cuatro”).

Ilustración 3.5: Ejemplo de la clase BD_GAF.

```
class BD_GAF(models.Model):
    caso = models.CharField(editable=True,max_length=256)
    expediente = models.CharField(editable=True,max_length=256)
    tipo_de_fraude = models.CharField(editable=True,max_length=256)
    rol = models.CharField(editable=True,max_length=256)
    fecha_de_inicio = models.DateField(editable=True)
    fecha_de_apertura = models.DateField(editable=True)
    fecha_de_cierre = models.DateField(editable=True)
    DT_o_provincia = models.CharField(editable=True,max_length=256)
    investigador = models.CharField(editable=True,max_length=256)
    servicios = models.FloatField(null=True)
    fecha_de_corte = models.DateField(editable=True)
    posible_corte = models.DateField(editable=True)
    nota1 = models.CharField(editable=True,max_length=256)
    nota2 = models.CharField(editable=True,max_length=256)

    def __str__(self):
        return unicode(self.anno),unicode(self.caso),unicode(self.expediente),\
               unicode(self.tipo_de_fraude),unicode(self.rol),unicode(self.fecha_de_inicio),\
               unicode(self.fecha_de_apertura),unicode(self.fecha_de_cierre),\
               unicode(self.DT_o_provincia),unicode(self.investigador),\
               unicode(self.servicios),unicode(self.imei),unicode(self.min_total_sms),\
               unicode(self.min_por_dias),unicode(self.perd_al_cierre_cuc),unicode(self.indemnizar)

```

3.5.1.2 Creador

El patrón Creador guía la asignación de responsabilidades relacionadas con la creación de objetos, tarea muy frecuente en los sistemas orientados a objetos.

Problema: ¿Quién debería ser responsable de crear una nueva instancia de alguna clase?

Solución: Crear una nueva instancia por la clase que tiene la información necesaria para realizar la creación del objeto y usar directamente la instancia creada por el objeto.

Ejemplo: En las clases *ViewGrupos* y *ViewDatos* se crea una instancia de la clase *function*, las mismas se usarán algunas funcionalidades de esta clase como *list_number*, la cual devolverá una lista de números a analizar como se muestra a continuación en la Ilustración 3.6.

Ilustración 3.6: Ejemplo de la clase ViewDatos.

```
class ViewDatos(MyOwnListCreateApiView):
    model = Datos
    queryset = Datos.objects.filter()
    serializer_class = DatosSerializers

    Datos.objects.all().delete()

    if NumerosAnalisisProcesar is not None:
        list = list_number(NumerosAnalisisProcesar)
        print(list)
        if list is not None:
            if CDR is not None:
                for t in list:
                    obj1 = SUBSCRIBER.objects.filter(PHONE_NUMBER = t)
                    obj = obj1.filter(PHONE_NUMBER = t).distinct("PHONE_NUMBER")
```

3.5.1.3 Bajo Acoplamiento

El acoplamiento es una medida de la fuerza con que una clase está conectada a otras clases, con qué medida las conoce y con qué medida recurre a ellas. Acoplamiento bajo significa que una clase

no depende de muchas clases. Acoplamiento alto significa que una clase recurre a muchas otras clases.

Problema: ¿Cómo dar soporte a una dependencia escasa y a un aumento de la reutilización?

Solución: Diseñar con el objetivo de tener las clases lo menos ligadas entre sí. De tal forma, qué en caso de producirse una modificación en alguna de ellas, se tenga la mínima repercusión posible en el resto de clases, potenciando la reutilización, y disminuyendo la dependencia entre las clases.

El patrón propone el diseño de clases más independientes, lo que reduce el impacto del cambio y facilita la reutilización en otros sistemas. Permite definir las funcionalidades de cada clase sin que estas presenten tanta dependencia entre ellas, así los cambios realizados en una, no afecta el funcionamiento de la otra. Dado que el framework Django introduce la utilización de vistas genéricas, brinda la posibilidad de reutilizar las funciones definidas en otras operaciones del sistema.

Ejemplo: A las clases *BD_GAF*, *RegitroMedios* y *Grupos* se le asignan responsabilidades de forma tal que *solo se comuniquen* con las clases que facilitan el proceso de dependencia de cada una de ellas, manteniendo así una mayor reutilización como se muestra a continuación en la Ilustración 3.7.

Ilustración 3.7: Ejemplo de la clase Grupos.

```
class Grupos(models.Model):
    Llamador = models.IntegerField(null=True)
    Grupo = models.CharField(null=True, editable=True, max_length=32)
    Dest_dif = models.IntegerField(null=True)
    Llamadas = models.IntegerField(null=True)
    Minutos = models.FloatField(null=True)

    def __str__(self):
        return unicode(self.Llamador), unicode(self.Grupo), unicode(self.Dest_dif),
        unicode(self.Llamadas), unicode(self.Minutos)
```

3.5.1.4 Alta Cohesión.

La cohesión es una medida de cuán relacionadas y enfocadas están las responsabilidades de una clase. Una alta cohesión caracteriza a las clases con responsabilidades estrechamente relacionadas que no realicen un trabajo enorme. Una baja cohesión hace muchas cosas no afines o realiza trabajo excesivo, por lo que se hacen difíciles de comprender, de reutilizar, de conservar y le afectan constantemente los cambios.

Problema: ¿Cómo mantener la complejidad dentro de límites manejables?

Solución: Asignar responsabilidades de manera que la información que almacena una clase sea coherente y esté relacionada con la clase.

Ejemplo: Las clases *LlamadasSalida* y *LlamadasEntrada* se le asignan responsabilidades para que trabajen solo en realizar consultas a la base de datos como se muestra a continuación en la Ilustración 3.8.

Ilustración 3.8: Ejemplo de la clase LlamadasSalida

```
class LlamadasSalida(MyOwnListCreateApiView):
    model = ResumenGraldeLlamadasSalida,
    queryset = ResumenGraldeLlamadasSalida.objects.filter(),
    serializer_class = ResumenGraldeLlamadasSalidaSerializer

    ResumenGraldeLlamadasSalida.objects.all().delete()


    if NumerosAnalisisProcesar is not None:
        list = list_number(NumerosAnalisisProcesar)
        if list is not None:
            if CDR is not None:
                for t in list:
                    obj = Grupos.objects.filter(Llamador = t).distinct("Llamador")

                    for j in obj:
                        llamador = j.Llamador
                        llamadas = Grupos.objects.filter(Llamador = j.Llamador).aggregate(Sum('Llamadas')).values().pop()
                        minutos = Grupos.objects.filter(Llamador = j.Llamador).aggregate(Sum('Minutos')).values().pop()
                        GruposLlamados = Grupos.objects.filter(Llamador = j.Llamador).__len__()
                        DestDif = Grupos.objects.filter(Llamador = j.Llamador).aggregate(Sum('Dest_dif')).values().pop()

                        ResumenGraldeLlamadasSalida(Llamador = llamador,DestDif=DestDif,Llamadas=llamadas,
                            Minutos = minutos,GruposLlamados=GruposLlamados).save()
```

3.6.2 Patrones GOF

Los patrones GOF describen las formas comunes en que diferentes tipos de objetos pueden ser organizados para trabajar unos con otros. Tratan la relación entre clases, la combinación de clases y la formación de estructuras de mayor complejidad. Permiten crear grupos de objetos que ayudan a realizar tareas complejas. Facilitan el aprendizaje y la comunicación entre programadores y diseñadores. Estos patrones se clasifican en tres tipos: creacionales, estructurales y de comportamiento. (20)

3.6.2.1 Estructurales.

Son los patrones de diseño de software que solucionan problemas de composición (agregación) de las clases y objetos. Estos patrones usan herencia para componer interfaces. Definen maneras de componer un objeto para obtener nuevas funcionalidades.

- **Decorador:** La función de este patrón es añadir responsabilidades adicionales a un objeto dinámicamente, proporcionando una alternativa flexible a la especialización mediante herencia, cuando se trata de añadir funcionalidades. Es aplicado a la generación de vistas, la solución que ofrece dicho patrón es la de añadir funcionalidad adicional a las plantillas. Este patrón se ve evidenciado en la creación de las vistas, teniendo en cuenta que el código HTML es el mismo para todas las vistas de la herramienta como se muestra a continuación en la Ilustración 3.9.

Ilustración 3.9: Ejemplo de utilización del patrón Decorador

```
{% extends 'jqgridGeneric.html' %}

{% block extrajs %}
    <script data-main="{{ STATIC_URL }}apps/etecsa/js/llam_clientesByPassEntr.js" src="{{ STATIC_URL }}js/libs/require.js"></
        script>
{% endblock %}
```

3.7 Pruebas de Software

Las aplicaciones por lo general son propensas a tener fallos. A veces, pueden contribuir al fracaso de cualquier proyecto de software, e impactar de forma negativa para la empresa que se cree. Surgen, por tanto, la pruebas de software con la necesidad de asegurar en lo posible, la calidad del producto.

Las pruebas de software son las investigaciones empíricas y técnicas cuyo objetivo es proporcionar información objetiva e independiente sobre la calidad del producto a la parte interesada. Las pruebas son básicamente un conjunto de actividades dentro del desarrollo de software. Dependiendo del tipo de pruebas, estas actividades podrán ser implementadas en cualquier momento de dicho proceso de desarrollo. Existen distintos modelos de desarrollo de software, así como modelos de pruebas. A cada uno corresponde un nivel distinto de involucramiento en las actividades de desarrollo (17).

3.7.1 Pruebas de Caja Negra

Las pruebas de caja negra se centran en los requisitos funcionalidades del sistema. Tiene como objetivo verificar que las entradas sean correctas, y que se muestren los resultados esperados. Son pruebas funcionales sin acceso al código fuente de las aplicaciones, y se llevan a cabo sin tener conocimiento de la estructura y funcionamiento interno del sistema.

La prueba de caja negra intenta encontrar las siguientes categorías de errores:

- Funciones incorrectas o ausentes.
- Errores de Interfaz.
- Errores en estructura de datos o en acceso a base de datos externos.
- Errores de rendimiento.
- Errores de Inicialización y de terminación (17).

3.7.1.1 Pruebas de Aceptación.

El uso de cualquier producto de software tiene que estar justificado por las ventajas que ofrece. Sin embargo, antes de empezar a usarlo es muy difícil determinar si sus ventajas realmente justifican su uso. El mejor instrumento para esta determinación es la llamada 'prueba de aceptación'. En esta prueba se evalúa el grado de calidad del software con relación a todos los aspectos relevantes para que el uso del producto se justifique.

Las pruebas de aceptación son pruebas de caja negra definidas por el cliente creadas a partir de las Historias de Usuario. Una Historia de Usuario no se puede considerar terminada hasta tanto pase correctamente todas las pruebas de aceptación. El objetivo del cliente es verificar que el software esté listo y que puede ser usado por los usuarios finales para ejecutar aquellas funciones y tareas para las cuales fue construido.

A continuación, se muestra el caso de prueba correspondiente a la historia de usuario “Realizar consultas generales de entrada” El resto de los casos de prueba de aceptación se detallan en el Anexo 4.

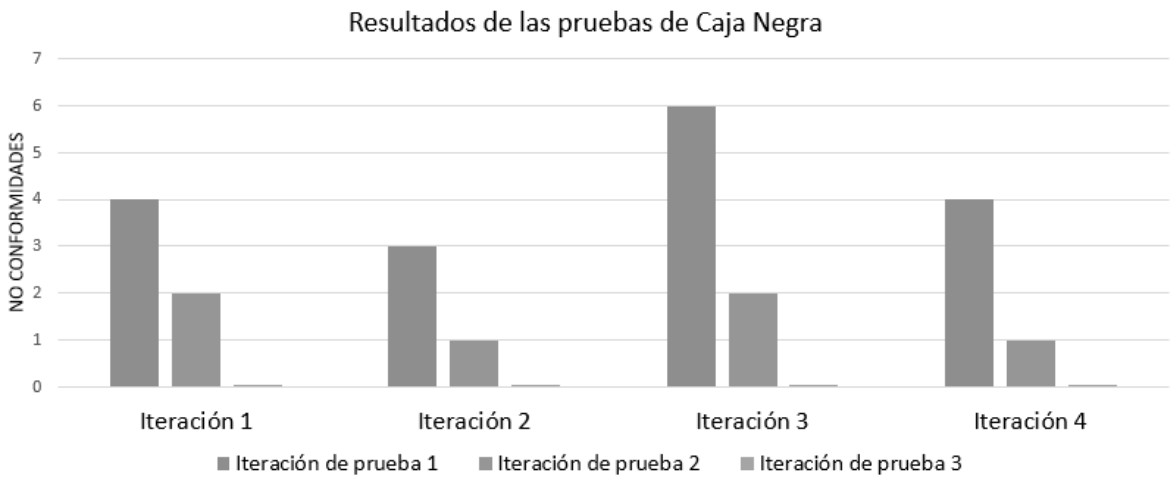
Tabla 19: Prueba de aceptación #1

Caso de Prueba de Aceptación	
Código: 1	**HU:** 1. Realizar consultas generales de entrada.
Responsable: Braiman González Sánchez y Alexander Valdés Molina	
Descripción: Prueba de funcionalidad para verificar que permita realizar todas las consultas generales.	
Condiciones de ejecución: El usuario debe dar clic en el menú “Consultas generales”	
Entrada/Pasos de ejecución: • Verificar que se desplegaron todas las consultas generales. • Verificar que cada consulta muestre correctamente los datos solicitados.	
Resultados esperados: Se debe mostrar todos los datos de los registros de las llamadas telefónicas solicitadas por el usuario al realizar la consulta.	
Evaluación de la prueba: Prueba realizada satisfactoriamente.	

3.6.1.2 Resultados de las pruebas.

La ilustración #7 muestra los resultados obtenidos de las pruebas de aceptación a los 25 casos de pruebas definidos por el cliente durante las 3 iteraciones de codificaciones planificadas. La primera iteración arrojó un total de 4 no conformidades (NC), la segunda 8 NC y la tercera 2 NC, lo que equivale a un total de 14 NC.

Ilustración 7: No conformidades detectadas durante las iteraciones.



A las historias de usuario correspondientes en cada iteración se le realizaron 3 iteraciones de pruebas. Estos errores fueron corregidos durante las iteraciones, cumpliéndose así todos los requisitos solicitados por el cliente.

3.6.1.3 Entorno de ejecución de las pruebas.

Las pruebas se realizaron en diferentes entornos, para verificar su correcto funcionamiento en las diferentes plataformas y verificar su funcionamiento en computadoras de disímiles características.

Sistema Operativo Windows 7 con las siguientes características:

- Procesador: Intel(R) Celeron(R) CPU 847 – 1.10 GHz
- Memoria RAM: 2.0 GB

Sistema Operativo Linux Debian 6 con las siguientes características:

- Procesador: Core TM i5-4210U 1.7 GHz
- Memoria RAM: 4GB

3.7.2 Pruebas de rendimiento

En la ingeniería del software, las pruebas de rendimiento son las pruebas que se realizan, desde una perspectiva, para determinar lo rápido que realiza una tarea un sistema en condiciones particulares de trabajo. También puede servir para validar y verificar otros atributos de la calidad del sistema, tales como la escalabilidad, fiabilidad y uso de los recursos. Las pruebas de rendimiento se esfuerza por mejorar el rendimiento, englobándose en el diseño y la arquitectura de un sistema, antes incluso del esfuerzo inicial de la codificación (17).

Para una mayor comprensión de los resultados de las pruebas, se tienen en cuenta las siguientes medidas:

- **Mínimo:** Mínimo tiempo (mili segundos) de conexión entre todas las solicitudes realizadas.

- **Máximo:** Máximo tiempo (mili segundos) de conexión entre todas las solicitudes realizadas.
- **Rendimiento:** número de peticiones procesadas en una unidad de tiempo, que puede ser segundos, minutos y horas.
- **Error:** Porciento de error respecto al número total de peticiones.

3.6.2.1 Pruebas de carga

Este es el tipo más sencillo de pruebas de rendimiento. Una prueba de carga se realiza generalmente para observar el comportamiento de una aplicación bajo una cantidad de peticiones esperada. Esta carga puede ser el número esperado de usuarios concurrentes utilizando la aplicación y que realizan un número específico de transacciones durante el tiempo que dura la carga. Esta prueba puede mostrar los tiempos de respuesta de todas las transacciones importantes de la aplicación (17).

A continuación se establecen los elementos a tener en cuenta para la realización de la prueba.

- Se analizará el comportamiento del sistema cuando 70 usuarios intentan conectarse concurrentemente al servidor en 1 segundo.
- Recursos necesarios:

Tabla 20: Recursos necesarios para la prueba de carga.

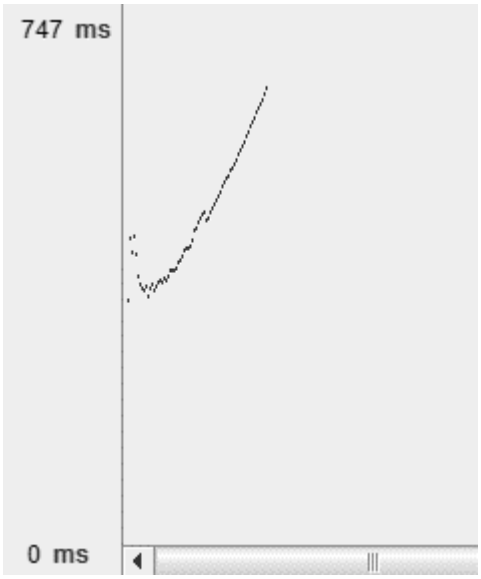
Tipo de prueba	Software	Hardware
Carga	JMeter 2.10	➢ Procesador: Core TM i5-4210U ➢ Motherboard: Acer Aspire E5-571 ➢ RAM: 1GB ➢ Disco duro: 20GB ➢ CPU: 1.7 GHz ➢ Red: 100 Mbps

Resultados de las pruebas:

- **Mínimo:** 112
- **Máximo:** 1499
- **Rendimiento:** 30 peticiones/segundos
- **Errores:** 0**%**

Gráfico del resultado:

Ilustración 3.11: Gráfico del resultado de la prueba de carga.



La gráfica muestra el comportamiento del tiempo de respuesta del servidor cuando se conectar 70 usuarios concurrentemente.

3.6.2.2 Pruebas de estrés

Esta prueba se utiliza normalmente para colapsar la aplicación. Se va doblando el número de usuarios que se agregan a la aplicación y se ejecuta una prueba de carga hasta que se rompe. Este tipo de prueba se realiza para determinar la solidez de la aplicación en los momentos de carga extrema y ayuda a los administradores a determinar si la aplicación rendirá lo suficiente en caso de que la carga real supere a la carga esperada. Permite además, determinar el límite real de la aplicación en cuanto a número de usuarios concurrentes y el número de transacciones por segundo (17).

Prueba #1: Esta prueba consiste en analizar el sistema teniendo en cuenta la conexión de un número de aplicaciones clientes muy superior a la esperada.

A continuación se establecen los elementos a tener en cuenta para la realización de la prueba.

- Se analizará el comportamiento del sistema cuando 90 usuarios intentan conectarse concurrentemente al servidor en 1 segundo.
- Recursos necesarios:

Tabla 21: Recursos necesarios para la prueba #1 de estrés.

Tipo de prueba	Software	Hardware
Carga	JMeter 2.10	➢ Procesador: Core TM i5-4210U ➢ Motherboard: Acer Aspire E5-571 ➢ RAM: 4GB ➢ Disco duro: 500GB ➢ CPU: 1.7 GHz

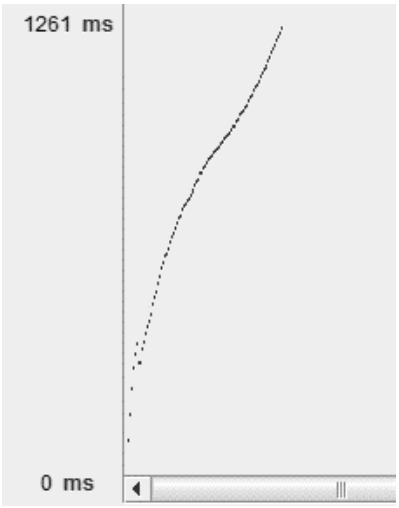
		➢ Red: 100 Mbps

Resultados de las pruebas:

- **Mínimo:**192
- **Máximo:** 2339
- **Rendimiento:** 33,6 peticiones/segundos
- **Error:** 0%

Gráfico del resultado:

Ilustración 3.12: Gráfico del resultado de la prueba #1 de estrés.



La gráfica muestra el comportamiento del tiempo de respuesta del servidor cuando se conectar 90 usuarios concurrentemente.

Prueba #2: Esta prueba consiste en analizar el sistema teniendo en cuenta la conexión de un número de aplicaciones clientes a las que el sistema no puede dar respuesta en su totalidad. A continuación se establecen los elementos a tener en cuenta para la realización de la prueba:

- Se analizará el comportamiento del sistema cuando 100 usuarios intentan conectarse concurrentemente al servidor en 1 segundo.
- Recursos necesarios:

Tabla 22: Recursos necesarios para la prueba #2 de estrés.

Tipo de prueba	Software	Hardware
Carga	JMeter 2.10	➢ Procesador: Core TM i5-4210U ➢ Motherboard: Acer Aspire E5-571 ➢ RAM: 4GB ➢ Disco duro: 500GB ➢ CPU: 1.7 GHz

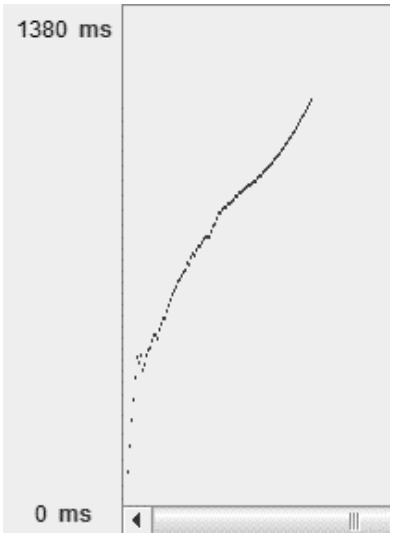
		➢ Red: 100 Mbps

Resultados de las pruebas:

- **Mínimo:** 120
- **Máximo:** 2543
- **Rendimiento:** 35,5 peticiones/segundos
- **Errores:** 0,13%

Gráfico del resultado:

Ilustración 3.13: Gráfico del resultado de la prueba #2 de estrés.



La gráfica muestra el comportamiento del tiempo de respuesta del servidor cuando se conectar 100 usuarios concurrentemente.

3.7.3 Pruebas de seguridad

Las pruebas de seguridad consisten en revisar las aplicaciones en búsqueda de vulnerabilidades y huecos de seguridad en las aplicaciones web, que pueden derivar en violaciones de los mecanismos de seguridad o en un mal funcionamiento del sistema. Este proceso se realiza bajo diferentes fases tales como el diseño de amenazas de acuerdo al tipo de aplicación, la ejecución de las pruebas, el análisis de los resultados y posteriormente la entrega de los hallazgos encontrados junto con el impacto generado por cada vulnerabilidad (17).

Para el desarrollo de este conjunto de pruebas se utilizó la aplicación Acunetix Web Vulnerability Scanner[27] mediante la cual se detectaron un conjunto de vulnerabilidades. La aplicación es capaz

[27] Herramienta heurística diseñado para replicar la metodología de un hacker para encontrar vulnerabilidades

de otorgar una calificación teniendo en cuenta la cantidad de vulnerabilidades encontradas. Estas calificaciones se dividen en 4 niveles:

- Nivel de alerta 3: Riesgo Alto – Son las vulnerabilidades clasificadas como las más peligrosas, estas ponen a un sitio en riesgo máximo para la piratería y el robo de datos.
- Nivel de alerta 3: Riesgo Medio – Son las vulnerabilidades causadas por una mala configuración del servidor y los defectos del sitio de codificación, facilitan la interrupción del servidor y la intrusión.
- Nivel de alerta 1: Riesgo Bajo – Son las vulnerabilidades derivadas de la falta de encriptación del tráfico de datos, o revelaciones de vías de directorio.
- Alerta Informativa - Estos son los elementos que se han descubierto durante una exploración y que se consideran de interés, por ejemplo, la posible revelación de una dirección IP interna o la dirección de correo electrónico, o la búsqueda de una cadena que se encuentre en la base de datos de Google Hacking.

Se realizaron dos iteraciones de prueba con el objetivo de eliminar todas las vulnerabilidades detectadas por la aplicación, detectándose en la primera iteración los siguientes resultados:

- Cantidad de alertas de nivel 3: 0
- Cantidad de alertas de nivel 2: 65
- Cantidad de alertas de nivel 1: 9
- Cantidad de alertas informativas: 7

De esta forma la aplicación se le otorga la calificación de riesgo medio al sistema, como se muestra a continuación en la siguiente imagen.

Ilustración 3.13: Resultado de la iteración #1 de la prueba de seguridad.

Luego de corregidos las vulnerabilidades del sistema detectadas por la aplicación, se realizó una segunda iteración donde se obtuvieron los siguientes resultados:

- Cantidad de alertas de nivel 3: 0

- Cantidad de alertas de nivel 2: 0
- Cantidad de alertas de nivel 1: 0
- Cantidad de alertas informativas: 0

De esta forma al sistema se le otorga la calificación de seguro como se muestra a continuación en la siguiente imagen.

Ilustración 3.14: Resultado de la iteración #2 de la prueba de seguridad.

3.8 Conclusiones parciales

En el presente capítulo se abordaron temas relacionados con el diseño e implementación del sistema. Se describieron las tareas de ingeniería realizadas a partir de las historias de usuario para obtener un nivel más detallado de la implementación de la aplicación. Además, se escogió la arquitectura Cliente/Servidor, el estilo arquitectónico MTV (Modelo-Template-View) y algunos de los patrones de diseño GRASP y GOF permitiendo organizar el diseño lógico de la aplicación. Se realizaron también pruebas de aceptación las cuales arrojaron un conjunto de no conformidades corrigieron todas las no conformidades en la última iteración realizada. Se realizaron las pruebas carga y estrés con la herramienta JMeter las cueles dieron como resultado que el sistema instalado en una computadora con 4GB de RAM, 500 de disco duro, 1.7GHz de CPU y una velocidad de red de 100,0 Mbps, admite 100 aplicaciones conectadas concurrentemente. Con las pruebas de seguridad se eliminaron las vulnerabilidades detectadas por la herramienta Acunetix Web Vulnerability Scanner.

Conclusiones generales

Una vez finalizada la fundamentación teórica que sustentó la presente investigación, definidas las características de la propuesta de solución y efectuado su desarrollo y validación, se obtuvieron las siguientes conclusiones:

- El estudio de las diferentes herramientas y sistemas de gestión del fraude, tanto a nivel nacional como internacional demostró que estas no cumplen con las características necesarias para la optimización del proceso de análisis de las alertas de posibles fraudes de ETECSA.
- Se determinaron las herramientas, tecnologías y la metodología de desarrollo de software que permitió el desarrollo de la aplicación web.
- Se logró mejorar el proceso de gestión y análisis de las alertas de posibles fraudes, eliminando las tareas de actualización manual de la base de datos por parte del jefe del departamento.

  Sustituyendo la herramienta HAT por un sistema que implementa una arquitectura cliente-servidor y que por consiguiente tiene la habilidad de ejecutar las consultas directamente a la base de datos de Discovery sin tener que realizar el proceso de exportar los datos de las alertas en hojas de cálculo, para luego cargarlos en el HAT y finalmente comenzar el análisis.
- Las pruebas realizadas arrojaron resultados satisfactorios, pues se identificó un grupo de no conformidades que fueron corregidas, lo que posibilitó la verificación y validación de las funcionalidades del sistema.

Referencia bibliográfica

1. Orquera E.; Herrera C.; Estévez L. (2015). Análisis De Protocolos De Señalización Para La Detección De Comportamientos Irregulares En Líneas De Telefonía Fija, Utilizando Sondas De Señalización. *Politécnica, (*35*), 1-11.*

2. Fraudes telefónicos (I) - Informática - Suplementos - Juventud Rebelde - Diario de la juventud cubana. [Online]. [Accessed 5 November 2015]. Available from: http://www.juventudrebelde.cu/suplementos/informatica/2014-05-28/fraudes-telefonicos-i.

3. Análisis global sobre fraude en las telecomunicaciones | Secure&IT. [Online]. [Accessed 2 November 2015]. Available from: https://www.secureit.es/analisis-global-sobre-fraude-en-las-telecomunicaciones.

4. América Latina busca seguridad en las cifras - Latin AmericaLatin America. [online]. [Accessed 8 June 2016]. Available from: http://www.gsma.com/latinamerica/es/america-latina-busca-seguridad-en-las-cifras.

5. Fraud Management & Detection Software | GRC | Analytics | SAP HANA | SAP. [Online]. [Accessed 24 November 2015]. Available from: http://www.sap.com/pc/analytics/governance-risk-compliance/software/fraud-management/index.html.

6. COMUNICADO: Mobileum presenta su nueva solución de análisis anti-fraude. *lainformacion.com* [online]. 3 March 2015. [Accessed 24 November 2015]. Available from: http://noticias.lainformacion.com/policia-y-justicia/fraude/comunicado-mobileum-presenta-su-nueva-solucion-de-analisis-anti-fraude_bxKtNaWeyATVn2GPVyEbt4/BARCELONA, March 3, 2015 /PRNewswire.

8. Kent Beck. (1999).*Extreme programing explained. [s.n.].*

9. The Python Tutorial — Python 2.7.11rc1 documentation. [Online]. [Accessed 23 November 2015]. Available from: https://docs.python.org/2.7/tutorial/index.html.

10. PostgreSQL: Comunicado de Prensa para PostgreSQL 9.4. [online]. [Accessed 23 November 2015]. Available from: http://www.postgresql.org/about/press/presskit94/es.

11. QuerySet API reference | Django documentation | Django. [Online]. [Accessed 26 May 2016]. Available from: https://docs.djangoproject.com.

12. PyCharm. *JetBrains* [online]. [Accessed 8 June 2016]. Available from: https://www.jetbrains.com/pycharm/Intelligent Python IDE with refactorings, debugger, code completion, on-the-fly code analysis and coding productivity orientation.

13. Apache JMeter - Apache JMeter™. [Online]. [Accessed 25 May 2016]. Available from: http://jmeter.apache.org.

14. Web application security with Acunetix. [Online]. [Accessed 25 May 2016]. Available from: http://www.acunetix.com/10. Apache JMeter - Apache JMeter™. [Online]. [Accessed 25 May 2016]. Available from: http://jmeter.apache.org.

15. Visual Paradigm - EcuRed. [online]. [Accessed 8 June 2016]. Available from: http://www.ecured.cu/Visual_Paradigm.

16. Autores, Colectivo de**.** *Metodologías Ágiles en el Desarrollo de Software.* [ed.] Emilio A. Sánchez, López Patricio Letelier Torres. Alicante, España : s.n., 12 de 11 de 2003.

17. Roger S. Pressman (2010). *Ingeniería de software: Un enfoque práctico*. (7ma.ed.).México: McGRAW-HILL INTERAMERICANA EDITORES, S.A. DE C.V.

18. Entorno cliente/servidor. [online]. [Accessed 8 June 2016]. Available from: http://es.ccm.net/contents/148-entorno-cliente-servidor.

19. Saul Garcia M (2015). La guía definitiva de django: Desarrolla aplicaciones Web de forma rápida y sencilla. [ed] Jeremy Dunck. Mountain View, CA. USA.

20. Larman, C. (1999). *UML y Patrones: Introducción al Análisis y Diseño orientado a Objetos.* (P. E. Vázquez, Ed., & L. M. Rodríguez, Trad.) México: Prentice Hall.Inc.

Bibliografía

1. Orquera E.; Herrera C.; Estévez L. (2015). Análisis De Protocolos De Señalización Para La Detección De Comportamientos Irregulares En Líneas De Telefonía Fija, Utilizando Sondas De Señalización. *Politécnica, (*35*), 1-11.*

2. Fraudes telefónicos (I) - Informática - Suplementos - Juventud Rebelde - Diario de la juventud cubana. [Online]. [Accessed 5 November 2015]. Available from: http://www.juventudrebelde.cu/suplementos/informatica/2014-05-28/fraudes-telefonicos-i.

3. Análisis global sobre fraude en las telecomunicaciones | Secure&IT. [Online]. [Accessed 2 November 2015]. Available from: https://www.secureit.es/analisis-global-sobre-fraude-en-las-telecomunicaciones/.

4. América Latina busca seguridad en las cifras - Latin AmericaLatin America. [online]. [Accessed 8 June 2016]. Available from: http://www.gsma.com/latinamerica/es/america-latina-busca-seguridad-en-las-cifras.

5. Fraud Management & Detection Software | GRC | Analytics | SAP HANA | SAP. [Online]. [Accessed 24 November 2015]. Available from: http://www.sap.com/pc/analytics/governance-risk-compliance/software/fraud-management/index.html.

6. COMUNICADO: Mobileum presenta su nueva solución de análisis anti-fraude. *lainformacion.com* [online]. 3 March 2015. [Accessed 24 November 2015]. Available from: http://noticias.lainformacion.com/policia-y-justicia/fraude/comunicado-mobileum-presenta-su-nueva-solucion-de-analisis-anti-fraude_bxKtNaWeyATVn2GPVyEbt4/BARCELONA, March 3, 2015 /PRNewswire.

8. Kent Beck. (1999).*Extreme programing explained. [s.n.].*

9. The Python Tutorial — Python 2.7.11rc1 documentation. [Online]. [Accessed 23 November 2015]. Available from: https://docs.python.org/2.7/tutorial/index.html.

10. PostgreSQL: Comunicado de Prensa para PostgreSQL 9.4. [online]. [Accessed 23 November 2015]. Available from: http://www.postgresql.org/about/press/presskit94/es/.

11. QuerySet API reference | Django documentation | Django. [Online]. [Accessed 26 May 2016]. Available from: https://docs.djangoproject.com.

12. PyCharm. *JetBrains* [online]. [Accessed 8 June 2016]. Available from: https://www.jetbrains.com/pycharm/Intelligent Python IDE with refactorings, debugger, code completion, on-the-fly code analysis and coding productivity orientation.

13. Apache JMeter - Apache JMeter™. [Online]. [Accessed 25 May 2016]. Available from: http://jmeter.apache.org.

14. Web application security with Acunetix. [Online]. [Accessed 25 May 2016]. Available from: http://www.acunetix.com/10. Apache JMeter - Apache JMeter™. [Online]. [Accessed 25 May 2016]. Available from: http://jmeter.apache.org/.

15. Visual Paradigm - EcuRed. [online]. [Accessed 8 June 2016]. Available from: http://www.ecured.cu/Visual_Paradigm.

16. Autores, Colectivo de**.** *Metodologías Ágiles en el Desarrollo de Software.* [ed.] Emilio A. Sánchez López Patricio Letelier Torres. Alicante, España : s.n., 12 de 11 de 2003.

17. Roger S. Pressman (2010). *Ingeniería de software: Un enfoque práctico*. (7ma.ed.).México: McGRAW-HILL INTERAMERICANA EDITORES, S.A. DE C.V.

18. Entorno cliente/servidor. [online]. [Accessed 8 June 2016]. Available from: http://es.ccm.net/contents/148-entorno-cliente-servidor.

19. Saul Garcia M (2015). La guía definitiva de django: Desarrolla aplicaciones Web de forma rápida y sencilla. [ed] Jeremy Dunck. Mountain View, CA. USA.

20. Larman, C. (1999). *UML y Patrones: Introducción al Análisis y Diseño orientado a Objetos.* (P. E. Vázquez, Ed., & L. M. Rodríguez, Trad.) México: Prentice Hall.Inc.

21. Visual Paradigm Product Overview. [Online]. [Accessed 23 November 2015]. Available from: http://www.visual-paradigm.com/support/documents/vpuserguide/12/13/5963_visualparadi.html.

22. Slideshare. [En línea] [Citado el: 1 de 4 de 2013.] http://www.slideshare.net/lilyPacheco7/arquitecturade-software-13925226.

23. Gabriel Maciá-Fernández. (2008). El fraude en roaming: estrategias de ataque y de defensa. *Taller IIRSA / CITEL "Servicios de roaming internacional". (pp. 4-7).* Granada, España.

24. CHAGOYA, Ena Ramos. Métodos y técnicas de investigación • GestioPolis. *GestioPolis - Conocimiento en Negocios* [online]. 1 July 2008. [Accessed 25 May 2016]. Available from: http://www.gestiopolis.com/metodos-y-tecnicas-de-investigacion.

14. VI Congreso del Partido Comunista de Cuba. (2011) .*Información sobre el resultado del Debate de los Lineamientos de la Política Económica y Social del Partido y la Revolución*. La Habana.

15. VI Congreso del Partido Comunista de Cuba. (2011) .*Resolución sobre los lineamientos de la política económica y social del partido y la revolución.* La Habana.

25. GUILLERMO, Publicado por: Pruebas de carga de un sitio web con JMeter. *Desarrollo Web Tutoriales para tu web* [online]. [Accessed 26 May 2016]. Available from: http://desarrollowebtutorial.com/pruebas-de-carga-de-un-sitio-web-con-jmeter.

26. GALÁN, Publicado por Ultiminio Ramos. Ingeniería de Software: Cómo hacer una prueba de estrés con JMeter. [Online]. [Accessed 26 May 2016]. Available from: http://php aplicado.blogspot.com/2013/06/como-hacer-una-prueba-de-estres-con-jmeter-apache.html
Guía para diseñar, ejecutar y verificar una prueba de estrés para una aplicación basada en Web, usando JMeter de Apache.

27. Carlos Sabino. (1994).Como hacer una tesis. (2da.ed). Caracas: Panapo.

28. Rolando A, Sayda C. (2011). *El proceso de investigación científica.* La Habana: Editorial Universitaria.

29. BEJARANO, Katherine Landsdorp; GUEVARA, Diana Nuñez; MONTIEL, Gerardo Alberto Castang. Heimdal: Sistema de alertas a través de un aplicativo para telefonía móvil. *Vínculos*, 2013, vol. 6, no 2, p. 34-47.

30. ROSERO VILLAVICENCIO, Karla Paulina; OCAMPO GAVILÁNES, Iván Vinicio. Artículo científico-Aplicación de monitoreo por centrales de emergencia país en la gestión antidelincuencial y en la aplicación táctica operativa. 2013.

31. VALENZUELA GARZÓN, Gabriel Santiago. Identificación y valoración de políticas de acción mediante sistemas de gestión de seguridad informática para evitar riesgos por fraude telefónico en las empresas que utilizan el servicio de telefonía IP. 2015.

32. Vistas genéricas basadas en clase. [Online]. [Accessed 26 May 2016]. Available from: https://pythonbc.com/blog/vistas-genericas-basadas-en-clase.

33. FRAUDE ETECSA. *Cubadebate* [online]. [Accessed 26 May 2016]. Available from: http://www.cubadebate.cu/?s=FRAUDE+ETECSA.

34. The Python Tutorial — Python 2.7.11rc1 documentation. [Online]. [Accessed 23 November 2015]. Available from: https://docs.python.org/2.7/tutorial/index.html.

35. COMMUNICATIONS FRAUD CONTROL ASSOCIATION, et al. World-wide telecom fraud survey 2006. *From CFCA website: http://www. cfca. org.*

36. BROOKS, Graham; BUTTON, Mark; GEE, Jim. The scale of health-care fraud: A global evaluation. *Security Journal*, 2012, vol. 25, no 1, p. 76-87.

37. VERSCHOOR, Curtis C. New evidence of benefits from effective ethics systems.(Ethics). *Strategic Finance*, 2003, vol. 84, no 11, p. 20-22.

38. Response Time Limits: Article by Jakob Nielsen. [Online]. [Accessed 26 May 2016]. Available from: https://www.nngroup.com/articles/response-times-3-important-limits.

Anexos

Anexo 1

Tabla 23. Realizar consultas de solamente los llamadores de las llamadas de entrada.

Historia de Usuario	
Número: 3	**Nombre:** Realizar consultas de solamente los llamadores de las llamadas de entrada.
Modificación de historia de usuario: Ninguna	
Usuario: Alexander Valdés Molina	**Iteración asignada:** 1
Prioridad en negocio: Media	**Puntos estimado:** 3/5
Riesgo en desarrollo: Medio	**Puntos reales:** 3/5
Descripción: Permite obtener los datos de las llamadas recibidas, de los servicios que no presentan relación con los usuarios llamados, estos datos son: • Solamente llamadores. • Total de Llamadas por Servicio. • Resumen de los servicios que no presentan relación.	
Observaciones: El usuario debe estar correctamente autenticado en el sistema.	
Prototipo de Interfaz: 	

Tabla 24: Realizar consulta de las llamadas de LDI recibidas de entrada.

Historia de Usuario	
Número: 5	**Nombre:** Realizar consulta de las llamadas de LDI recibidas, de entrada.
Modificación de historia de usuario: Ninguna	
Usuario: Alexander Valdés Molina	**Iteración asignada:** 1
Prioridad en negocio: Media	**Puntos estimado:** 1/5

Riesgo en desarrollo: Medio	**Puntos reales:** 1/5
Descripción: Permite obtener los datos de las llamadas de larga distancia internacional recibidas, así como un resumen de las llamadas de entradas.	
Observaciones: El usuario debe estar correctamente autenticado en el sistema.	
Prototipo de Interfaz: 	

Tabla 25: Realizar consulta de las llamadas con duración menor que 3 minutos.

Historia de Usuario	
Número: 6	**Nombre:** Realizar consulta de las llamadas con duración menor que 3 minutos.
Modificación de historia de usuario: Ninguna	
Usuario: Alexander Valdés Molina	**Iteración asignada:** 1
Prioridad en negocio: Media	**Puntos estimado:** 1/5
Riesgo en desarrollo: Medio	**Puntos reales:** 1/5
Descripción: Permite obtener los datos de las llamadas recibidas con una duración menor que 3 minutos, además de un resumen de las mismas.	
Observaciones: El usuario debe estar correctamente autenticado en el sistema. Solo devuelve las llamadas que no sobrepasen los 3 minutos.	

Prototipo de Interfaz:

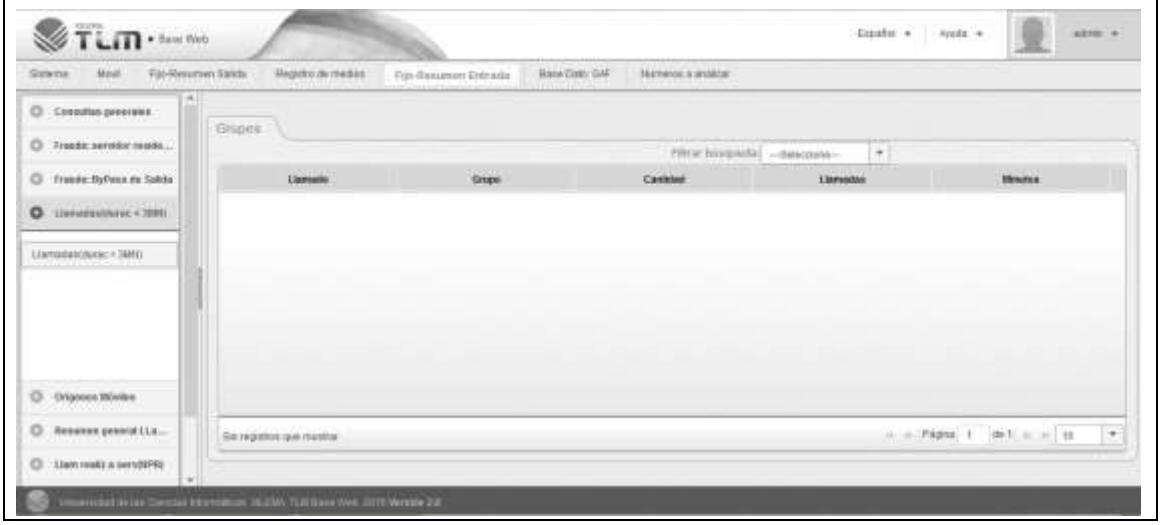


Tabla 26: Realizar consultas que se originan de los móviles.

Historia de Usuario	
Número: 7	**Nombre:** Realizar consultas que se originan de los móviles.
Modificación de historia de usuario: Ninguna	
Usuario: Alexander Valdés Molina	**Iteración asignada:** 1
Prioridad en negocio: Media	**Puntos estimado:** 3/5
Riesgo en desarrollo: Medio	**Puntos reales:** 3/5
Descripción: Permite obtener los datos de los móviles que llaman a diferentes usuarios y los móviles que reciben llamadas. • Llamadas realizadas por los móviles. • Llamadas recibidas por los móviles. • Resumen de las llamadas que se originan de los móviles.	
Observaciones: El usuario debe estar correctamente autenticado en el sistema.	

Prototipo de Interfaz:

Tabla 27: Realizar consultas para el fraude del servidor residencial.

Historia de Usuario	
Número: 8	**Nombre:** Realizar consultas para el fraude del servidor residencial.
Modificación de historia de usuario: Ninguna	
Usuario: Alexander Valdés Molina	**Iteración asignada:** 1
Prioridad en negocio: Media	**Puntos estimado:** 1
Riesgo en desarrollo: Medio	**Puntos reales:** 1
Descripción: Permite obtener los datos de los fraudes del servidor residencial donde se encuentran los clientes potenciales y se calculan las perdidas. • Llamadas recibidas con duración menor que 3 minutos. • Llamadas con duración menor que 3 minutos por servicios. • Clientes potenciales. • Cálculos de pérdidas. • Resumen del fraude del servidor residencial.	
Observaciones: El usuario debe estar correctamente autenticado en el sistema.	

Prototipo de Interfaz:

Tabla 28: Realizar consultas para el fraude de ByPass de salida.

Historia de Usuario	
Número: 9	**Nombre:** Realizar consultas para el fraude de ByPass de salida.
Modificación de historia de usuario: Ninguna	
Usuario: Alexander Valdés Molina	**Iteración asignada:** 1
Prioridad en negocio: Media	**Puntos estimado:** 6/10
Riesgo en desarrollo: Medio	**Puntos reales:** 6/10
Descripción: Permite obtener los datos de los fraudes de ByPass de salida, así como los datos de los clientes de ByPass. • Llamadas de entrada con duración mayor que 3 minutos. • Total de llamadas de entrada con duración mayor que 3 minutos por orígenes. • Resumen de llamadas de entrada con duración mayor que 3 minutos. • Llamadas de entrada con duración mayor que 3 minutos por días. • Llamadas de clientes ByPass. • Resumen del fraude de ByPass de salida.	
Observaciones: El usuario debe estar correctamente autenticado en el sistema.	

Prototipo de Interfaz:

Tabla 29: Realizar consultas generales de salidas.

Historia de Usuario	
Número: 10	**Nombre:** Realizar consultas generales de salidas.
Modificación de historia de usuario: Ninguna	
Usuario: Braiman González Sánchez	**Iteración asignada:** 2
Prioridad en negocio: Media	**Puntos estimado:** 1
Riesgo en desarrollo: Medio	**Puntos reales:** 1
Descripción: Permite obtener datos solicitados en las diferentes consultas generales de salida. • Servicios que presentan una estrecha relación. • Llamadas realizadas a los servicios de especial interés. • Total de llamadas realizadas a los servicios de especial interés. • Llamadas realizadas a destinos fraudulentos. • Resumen general de salida.	
Observaciones: El usuario debe estar correctamente autenticado en el sistema.	

Prototipo de Interfaz:

Tabla 30: Realizar consultas del resumen general de llamadas de salida.

Historia de Usuario	
Número: 11	**Nombre:** Realizar consultas del resumen general de llamadas de salida.
Modificación de historia de usuario: Ninguna	
Usuario: Braiman González Sánchez	**Iteración asignada:** 2
Prioridad en negocio: Media	**Puntos estimado:** 1
Riesgo en desarrollo: Medio	**Puntos reales:** 1
Descripción: Permite obtener los datos solicitados por las consultas que componen el resumen general de las llamadas de salida. • Datos. • Grupos. • Llamadas de salida. • Total de llamadas por servicio. • Resumen general de las llamadas de salida.	
Observaciones: El usuario debe estar correctamente autenticado en el sistema.	

Prototipo de Interfaz:

Tabla 31: Realizar consultas de los llamadores de las llamadas de salida.

Historia de Usuario	
Número: 12	**Nombre:** Realizar consultas de los llamadores, de las llamadas de salida.
Modificación de historia de usuario: Ninguna	
Usuario: Braiman González Sánchez	**Iteración asignada:** 2
Prioridad en negocio: Media	**Puntos estimado:** 3/5
Riesgo en desarrollo: Medio	**Puntos reales:** 3/5
Descripción: Permite obtener los datos de las llamadas recibidas de los servicios que no presentan relación, con los usuarios llamados. Estos datos son: • Solamente los llamados. • Total de llamadas por servicio. • Resumen de los usuarios llamados.	
Observaciones: El usuario debe estar correctamente autenticado en el sistema.	

Prototipo de Interfaz:

Tabla 32: Realizar consultas a las llamadas mayores de 3 minutos.

Historia de Usuario	
Número: 13	**Nombre:** Realizar consultas a las llamadas mayores de 3 minutos.
Modificación de historia de usuario: Ninguna	
Usuario: Braiman González Sánchez	**Iteración asignada:** 2
Prioridad en negocio: Media	**Puntos estimado:** 3/5
Riesgo en desarrollo: Medio	**Puntos reales:** 3/5
Descripción: Permite obtener los datos de las llamadas realizadas que tengan una duración mayor que 3 minutos a los servicio de todos los grupos de suscriptores con los que no presentan relación. • Llamadas realizadas con una duración mayor que 3 minutos a todos los grupos. • Total de llamadas realizadas con una duración mayor que 3 minutos por servicio. • Resumen de las llamadas con duración mayor que 3 minutos.	
Observaciones: El usuario debe estar correctamente autenticado en el sistema.	

Prototipo de Interfaz:

Tabla 33: Realizar consultas a las llamadas realizadas mediante el empleo de tarjetas propias.

Historia de Usuario	
Número: 14	**Nombre:** Realizar consultas a las llamadas realizadas mediante el empleo de tarjetas propias.
Modificación de historia de usuario: Ninguna	
Usuario: Braiman González Sánchez	**Iteración asignada:** 2
Prioridad en negocio: Media	**Puntos estimado:** 3/5
Riesgo en desarrollo: Medio	**Puntos reales:** 3/5
Descripción: Permite obtener los datos de las llamadas realizadas usando tarjetas propias y la cantidad de llamadas de las mismas. • Llamadas realizadas utilizando tarjetas propias. • Cantidad de llamadas y recargas por tarjetas. • Resumen de las llamadas mediante el empleo de tarjetas propias.	
Observaciones: El usuario debe estar correctamente autenticado en el sistema.	

Prototipo de Interfaz:

Tabla 34: Realizar consulta de las llamadas de LDN recibidas de salida.

Historia de Usuario	
Número: 15	**Nombre:** Realizar consulta de las llamadas de LDN recibidas, de salida.
Modificación de historia de usuario: Ninguna	
Usuario: Braiman González Sánchez	**Iteración asignada:** 2
Prioridad en negocio: Media	**Puntos estimado:** 1/5
Riesgo en desarrollo: Medio	**Puntos reales:** 1/5
Descripción: Permite obtener los datos de las llamadas de larga distancia nacional recibidas, así como un resumen de las llamadas de salida.	
Observaciones: El usuario debe estar correctamente autenticado en el sistema.	
Prototipo de Interfaz:	

Tabla 35: Realizar consulta de las llamadas de LDI recibidas, de salida.

Historia de Usuario	
Número: 16	**Nombre:** Realizar consulta de las llamadas de LDI recibidas, de salida.
Modificación de historia de usuario: Ninguna	
Usuario: Braiman González Sánchez	**Iteración asignada:** 2
Prioridad en negocio: Media	**Puntos estimado:** 1/5
Riesgo en desarrollo: Medio	**Puntos reales:** 1/5
Descripción: Permite obtener los datos de las llamadas de larga distancia internacional recibidas, así como un resumen de las llamadas de salida.	
Observaciones: El usuario debe estar correctamente autenticado en el sistema.	
Prototipo de Interfaz: 	

Tabla 36: Realizar consultas a las conexiones de ISP.

Historia de Usuario	
Número: 17	**Nombre:** Realizar consultas a las conexiones de ISP
Modificación de historia de usuario: Ninguna	
Usuario: Braiman González Sánchez	**Iteración asignada:** 2
Prioridad en negocio: Media	**Puntos estimado:** 3/5
Riesgo en desarrollo: Medio	**Puntos reales:** 3/5
Descripción: Permite obtener los datos de conexiones a los (Proveedores de servicio de internet) ISP y la cantidad de veces que se conectan por día. • Conexiones a ISP. • Total de conexiones por días. • Resumen de las conexiones a ISP.	

Observaciones: El usuario debe estar correctamente autenticado en el sistema.
Prototipo de Interfaz: 

Tabla 37: Realizar consultas que están destinadas a los móviles.

Historia de Usuario	
Número: 18	**Nombre:** Realizar consultas que están destinadas a los móviles.
Modificación de historia de usuario: Ninguna	
Usuario: Braiman González Sánchez	**Iteración asignada:** 2
Prioridad en negocio: Media	**Puntos estimado:** 3/5
Riesgo en desarrollo: Medio	**Puntos reales:** 3/5
Descripción: Permite obtener las llamadas que se le realizaron a los móviles. • Llamadas realizadas a los móviles. • Llamadas recibidas por los móviles. • Resumen de las llamadas destinadas a los móviles.	
Observaciones: El usuario debe estar correctamente autenticado en el sistema.	

Prototipo de Interfaz:

Tabla 38: Realizar consultas específicas del fraude en el Cyber Café.

Historia de Usuario	
Número: 19	**Nombre:** Realizar consultas específicas del fraude en el Cyber Café.
Modificación de historia de usuario: Ninguna	
Usuario: Braiman González Sánchez	**Iteración asignada:** 2
Prioridad en negocio: Media	**Puntos estimado:** 2/5
Riesgo en desarrollo: Medio	**Puntos reales:** 2/5
Descripción: Permite obtener los datos de los fraudes ocurridos en el Cyber Café y calcular las pérdidas. • Cálculos de pérdidas. • Resumen del cálculo de las pérdidas.	
Observaciones: El usuario debe estar correctamente autenticado en el sistema.	

Prototipo de Interfaz:

Tabla 39: Realizar consultas para el fraude de Bypass de entrada.

Historia de Usuario	
Número: 20	**Nombre:** Realizar consultas para el fraude de Bypass de entrada.
Modificación de historia de usuario: Ninguna	
Usuario: Braiman González Sánchez	**Iteración asignada:** 2
Prioridad en negocio: Media	**Puntos estimado:** 2
Riesgo en desarrollo: Medio	**Puntos reales:** 2

Descripción: Permite obtener los datos de los fraudes de Bypass de entrada, así como los datos de los clientes Bypass.

- Llamadas con duración mayor que 3 minutos.
- Total de llamadas con duración mayor que 3 minutos.
- Resumen de las llamadas con duración mayor que 3 minutos.
- Llamadas con duración mayor que 10 minutos.
- Llamadas con duración mayor que 20 minutos.
- Total de llamadas realizadas a cliente ByPass con duración mayor que 3 minutos.
- Resumen de las llamadas realizadas a cliente ByPass con duración mayor que 3 minutos.
- Cantidad de días trabajados.
- Cálculos de pérdidas.
- Resumen del fraude de ByPass de entrada.

Observaciones: El usuario debe estar correctamente autenticado en el sistema.

Prototipo de Interfaz:

Tabla 40: Gestionar los registros de medios.

Historia de Usuario	
Número: 21	**Nombre:** Gestionar los registros de medios.
Modificación de historia de usuario: Ninguna	
Usuario: Alexander Valdés Molina	**Iteración asignada:** 3
Prioridad en negocio: Media	**Puntos estimado:** 2/5
Riesgo en desarrollo: Medio	**Puntos reales:** 2/5
Descripción: Permite añadir, eliminar o modificar algunas características de los teléfonos móviles, como: el IMEI, la marca y el modelo.	
Observaciones: El usuario debe estar correctamente autenticado en el sistema.	
Prototipo de Interfaz:	

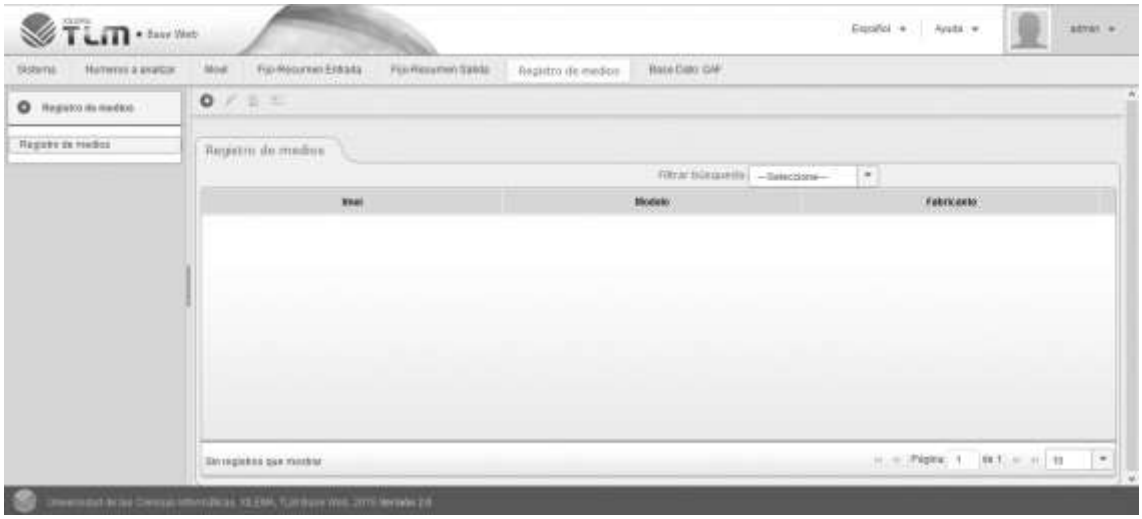


Tabla 41: Gestionar la base de datos de GAF.

Historia de Usuario

Número: 22	**Nombre:** Gestionar la base de datos de GAF.
Modificación de historia de usuario: Ninguna	
Usuario: Alexander Valdés Molina	**Iteración asignada:** 3
Prioridad en negocio: Media	**Puntos estimado:** 2/5
Riesgo en desarrollo: Medio	**Puntos reales:** 2/5
Descripción: Permite añadir, eliminar o modificar casos relacionados con el fraude.	
Observaciones: El usuario debe estar correctamente autenticado en el sistema.	
Prototipo de Interfaz: 	

Tabla 42: Exportar a Excel los resultados de las consultas.

Historia de Usuario	
Número: 25	**Nombre:** Exportar a Excel los resultados de las consultas.
Modificación de historia de usuario: Ninguna	
Usuario: Alexander Valdés Molina	**Iteración asignada:** 3
Prioridad en negocio: Media	**Puntos estimado:** 2/5
Riesgo en desarrollo: Medio	**Puntos reales:** 2/5
Descripción: Permite exportar los resultados de las consultas a un módulo Excel, con el fin de poder analizarlos posteriormente.	
Observaciones: El usuario debe estar correctamente autenticado en el sistema. Deben estar los datos cargados en la tabla para su correcta exportación.	

Prototipo de Interfaz: