UNIVERSIDAD DE CHILE
FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

SISTEMA DE RENOVACION DE CREDENCIALES EN TELEFONICA CHILE

TESIS PARA OPTAR AL GRADO DE MAGISTER EN TECNOLOGÍAS DE LA INFORMACION

LUISA JASMIN ZURITA HENRIQUEZ

PROFESOR GUIA:
ALEJANDRO HEVIA ANGULO

MIEMBROS DE LA COMISION:
NELSON BALOIAN TATARYAN
JOSE A. PINO URTUBIA
XAVIER BONNAIRE FAVRE

SANTIAGO DE CHILE

2018

Resumen

Telefónica Chile, como parte de sus servicios por el área de empresas tiene el deber de mantener, monitorear y mejorar el servicio de red hacia las empresas clientes de la organización. Entre las empresas clientes se encuentran grandes empresas financieras y de retail chilenas.

Una de las principales problemáticas que presenta la organización es la asignación de credenciales de acceso al personal para que estos puedan operar sobre dispositivos como router y firewalls. Hoy no existe control sobre quienes operan estos dispositivos y por ende cualquier persona con los conocimientos técnicos necesarios puede accederlos. Ante esto, la organización tomó la determinación de iniciar un proyecto denominado GUIA orientado a solucionar esta problemática. Sin embargo aún después de este proyecto persisten algunas dificultades, entre las que se cuentan: 1) Las credenciales de acceso de los dispositivos de red no son secretas, puesto que hoy en día son genéricas y su valor puede ser encontrado navegando en internet. 2) La operación de quienes se encargan de mantener y monitorear los servicios es poco flexible: sólo operan mediante planillas Excel con la información de las empresas clientes. 3) No existe registro alguno de las actividades realizadas dentro de los dispositivos de red del cliente.

Luego de una revisión y posterior análisis de la situación actual, se determinó que muchas de las caídas de los servicios hacia las empresas clientes se deben a accesos no autorizados y manipulación maliciosa de la configuración de los propios dispositivos. Es aquí en donde se presenta la oportunidad de mejorar el sistema GUIA.

La solución propuesta consiste en crear una herramienta de visualización de los dispositivos de red y de su estado de renovación de credenciales. El sistema identifica qué dispositivos corresponden a cada cliente, permite renovar las credenciales de los dispositivos en forma automática y periódica, dando la opción de generar una renovación a petición, en caso de falla de la renovación automática. Los beneficios asociados a esta solución son variados: Las credenciales asociadas a los dispositivos de red ya no serán de dominio público, ya que cambiarán continuamente de acuerdo a las políticas preestablecidas por la organización. Adicionalmente se mantendrá registro de los accesos a dichos dispositivos, por lo que se conocerá quienes han operado sobre estos.

En nuestra opinión, la incorporación de estas mejoras conlleva un aporte significativo en términos de seguridad de las redes de los clientes de Telefónica.

Agradecimientos

A mi familia…

Tabla de Contenido

Índice de Figuras

Índice de Tablas

Introducción

Antecedentes

Oracle Identity Manager (OIM) [1] se concibió con el objetivo de administrar todo el ciclo de vida de las identidades y controlar el acceso a los diferentes recursos corporativos TI. Su funcionamiento se enfoca principalmente en la protección del otorgamiento de accesos a sistemas, reducción del trabajo operativo del personal soporte y a la automatización de procesos de negocio mediante reglas de aprobación y políticas de otorgamiento de acceso [2]. Un ejemplo de esto, es la desvinculación de una persona de la organización. Este sistema detecta esto desde las bases de datos de recursos humanos y elimina automáticamente todos sus accesos. De la misma manera ocurre cuando una persona es contratada, la plataforma OIM [1] detecta la incorporación y según su cargo, unidad organizacional y lugar de trabajo al que pertenece le otorga las cuentas de accesos a los sistemas que le corresponde según su perfil. Así, uno de sus objetivos es centralizar los mecanismos de acceso y disminuir el trabajo operativo de los profesionales que mantienen la plataforma.

Motivados por la necesidad de implementar estos mecanismos, Telefónica ve la necesidad de iniciar el proyecto GUIA aplicando las definiciones de OIM y a su vez aplicando las políticas internas de la organización [3]. Si bien con el proyecto GUIA[2] se cubría una gran necesidad dentro de la organización, esta no quedaba exenta de algunas falencias no contempladas en el proyecto original. En adelante se detallan estas falencias y la forma en cual se enfrenta su posterior desarrollo.

[1] OIM: Oracle Identity Manager.

[2] GUIA: Gestión Unificada e Integrada de acceso.

Descripción general del problema

Telefónica cuenta actualmente con un gran número de dispositivos de red, denominado "el parque", entre los que se incluyen módems, routers y firewalls. Los dispositivos de red se encuentran distribuidos entre sus clientes segmentados en dos grandes grupos: empresas y particulares. Para el grupo empresas, se creó una red de monitoreo denominada HGM (Herramienta de Gestión y Monitoreo) mediante la cual se hace chequeo de los servicios que la organización le presta a los clientes. Este servicio es llevado a cabo por las denominadas mesas tecnológicas, quienes reciben las problemáticas y realizan íntegramente la labor de monitoreo (ver Figura 1). Hoy se cuenta con tres mesas tecnológicas y entre estas se tienen distribuidos los distintos grupos de clientes.

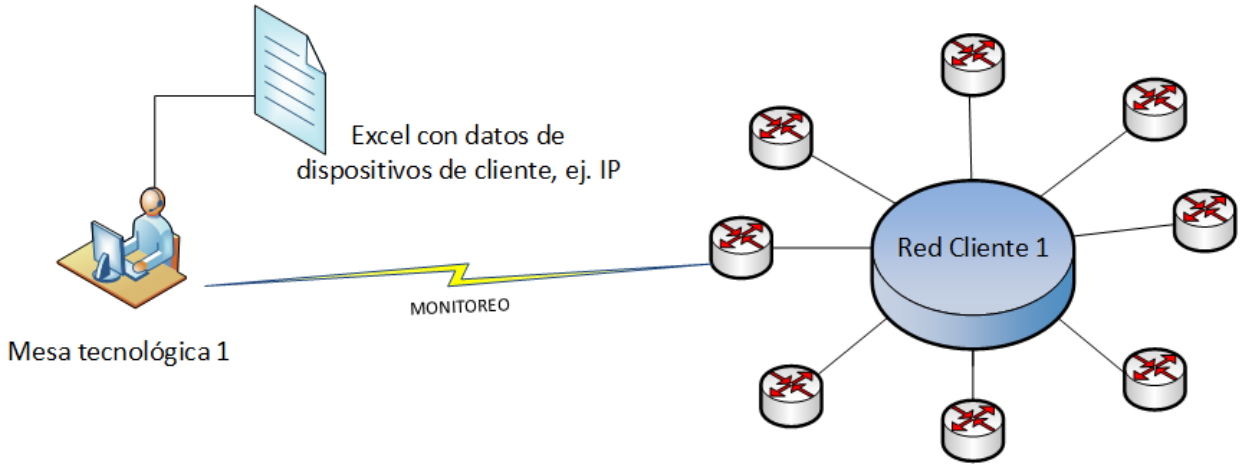

FIGURA 1: ESQUEMA GENERAL

El servicio prestado por las mesas tecnológicas consiste en: monitoreo de red LAN/WAN, respaldo de configuración de los dispositivos de red, gestión de fallas del equipamiento y generación de reportes.

Cada empresa cliente posee internamente una red propia y esta puede o no ser administrada de forma íntegra a través de la red HGM[3], esto conforma el servicio prestado por Telefónica. La Figura 2, muestra los dispositivos de red asociados al cliente empresa "X" y que además pertenecen a Telefónica SA. En este caso, se habla de clientes en banda.

[3] HGM: Herramienta de gestión y Monitoreo.

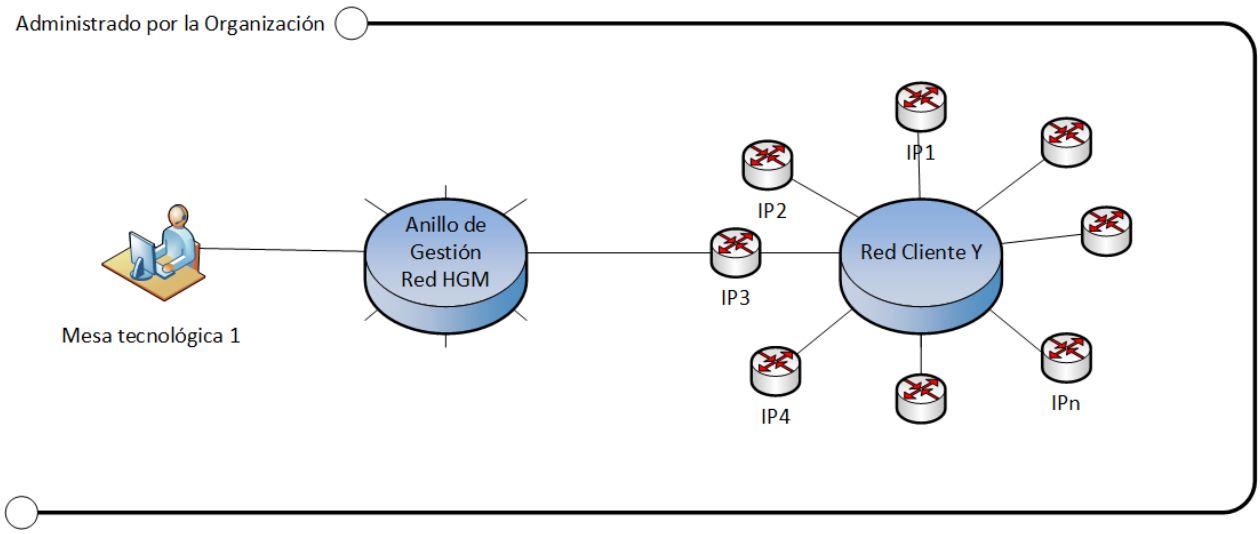

FIGURA 2: DISPOSITIVOS DE RED ASOCIADOS A CLIENTE EMPRESA X.

LA

Figura 3, muestra los dispositivos de red que son propios de la empresa cliente y aun así son monitoreados por una mesa tecnológica. En este caso se salta desde uno de los dispositivos de la red HGM hacia la red del cliente mediante elementos dispuestos para estos saltos. En este caso se habla de clientes fuera de banda.

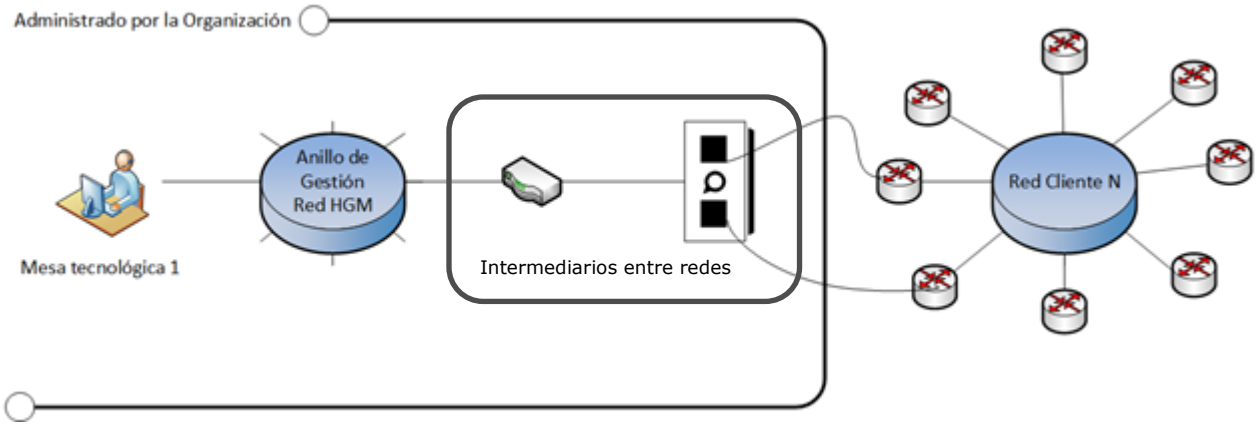

FIGURA 3: RED DE CLIENTE FUERA DE BANDA.

En ambos casos el o los servicios prestados se hacen a través de las mesas tecnológicas. Con esto varía sólo la responsabilidad de la organización sobre los dispositivos de red mantenidos en el cliente. Las mesas tecnológicas al momento de realizar el monitoreo necesitan toda la información asociada a los dispositivos de red, incluidas las credenciales de acceso de los mismos para poder hacer una evaluación del dispositivo en caso de presentarse algún problema. Aquí es donde se presenta la principal problemática a resolver (ver Figura 4), dado que se necesita acceder a los dispositivos de red mediante las credenciales propias de éste y hoy estas credenciales son de dominio público. De hecho, dependiendo del modelo del dispositivo, estas credenciales se pueden encontrar navegando por internet en sitios del mismo vendedor, foros de usuarios u otros lugares similares, lo que hace que cualquier red o servicio que se preste a los clientes sea vulnerable y propenso a ser accedido en forma no autorizada por personas externas a la organización.

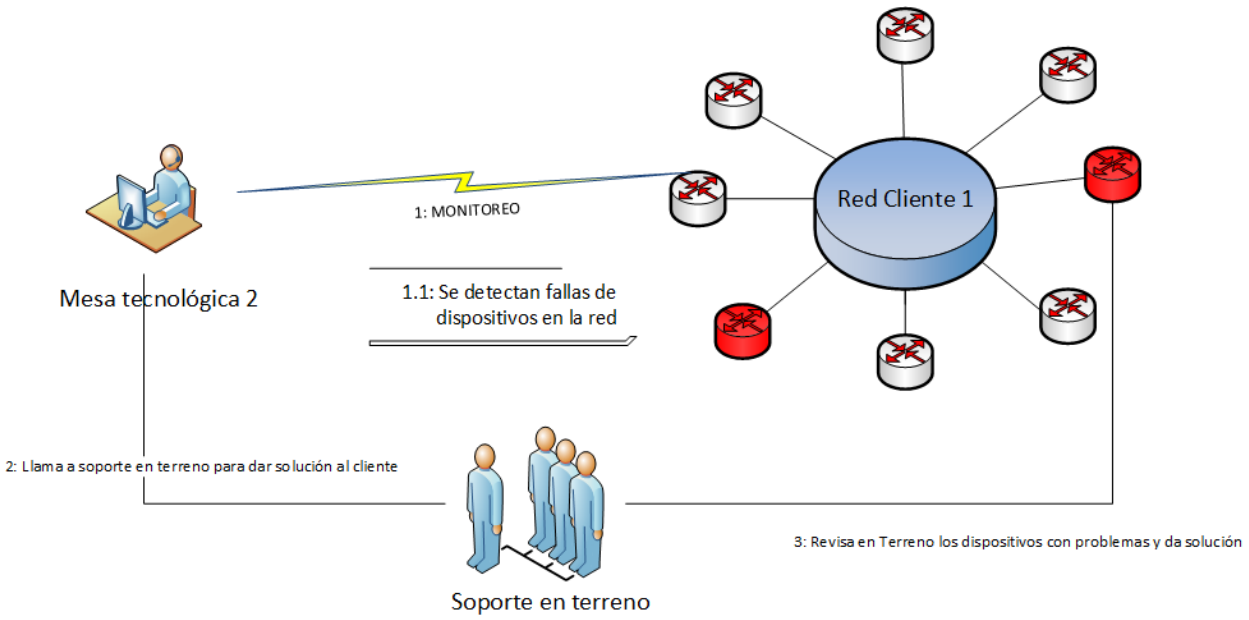

FIGURA 4: FLUJO ASOCIADO AL MONITOREO DE RED.

Derivado de lo anterior se presenta un conjunto de problemáticas que requieren de solución a corto plazo, como por ejemplo:

1. Las credenciales locales de los dispositivos de red son genéricas y se encuentran publicadas en internet.
2. Las empresas clientes generan reclamos asociados a la generalidad de las credenciales que se utilizan. Por ejemplo las empresas advierten de:
    a. Riesgo de accesos no autorizados a los dispositivos de los clientes por parte de personal desvinculado de la organización.
    b. Carencia de un registro centralizado de las actividades realizadas en los dispositivos de red del cliente.
    c. Carencia de un catastro de los dispositivos de red instalados por clientes donde se puede identificar la marca y el tipo de dispositivo.
    d. Carencia de mecanismos más robustos e integrados de manejo de credenciales de red, todo el operar de las mesas tecnológicas se basa en planillas Excel para acceder a dichas credenciales.

Solución Propuesta

Dado lo anterior y la preocupación de los clientes, expuesta con anterioridad, es que la organización ha decidido llevar a cabo el desarrollo de un sistema que se integrará con GUIA, que permita abordar la problemática. Las personas que hoy operan sobre los dispositivos de red lo realizan de forma directa sobre el equipo al momento de generarse una contingencia, ejemplo de esto es el cambio de configuración en un router en terreno. Para este caso, las personas por defecto ya conocen las credenciales de los dispositivos, dado que son generales a todos estos y además se pueden encontrar en internet.

La solución propuesta consiste en proveer de estos accesos pero de forma restringida; sólo tendrán acceso quienes previamente estén autorizados para ello y las credenciales de los dispositivos dependerán directamente del nuevo sistema. En este caso, el sistema de renovación de credenciales, SRC[4], se cohesionará con los otros subsistemas de GUIA y además hará uso de algunos servicios expuestos. De esta forma, por ejemplo, sólo los usuarios autorizados mediante el subsistema de perfilamiento podrán operar sobre el

[4] SRC: Sistema de renovación de Credenciales.

SRC y así se podrá mantener un control centralizado en relación a los accesos sobre los dispositivos de red.

Las credenciales de cada dispositivo de red creadas vía SRC sólo serán de utilidad para el caso en que no haya comunicación entre el ACS[5] y los dispositivos de red, es decir ante una contingencia. En este contexto, podemos mencionar que en el procedimiento normal de uso los usuarios que tengan los privilegios de acceso sobre los dispositivos podrán accederlos con sus credenciales de red. Todo el personal de la organización posee credenciales de red, que le permite acceder a diversos sistemas dentro de la compañía, entre los que se cuentan intranet y correo. En este caso, el nuevo sistema ante una contingencia demuestra su ventaja con el sistema actual al mantener registradas y actualizadas las credenciales de cada uno de los dispositivos de red. Estas credenciales serán renovadas periódicamente, es decir se generarán configuraciones que permitirán agrupar los dispositivos de red y cada configuración podrá tener una periodicidad de renovación.

El sistema SRC tiene como requisitos lo siguiente:

- Tener una renovación periódica de las credenciales de los equipos de red.
- Permitir configurar la periodicidad de renovación.
- Mostrar gráficamente el estado del proceso de renovación para todos los dispositivos de red, en este caso si fue o no renovado.
- Registrar todo evento asociado a la renovación.
- Crear nuevas cuentas locales dentro de los dispositivos de red para poder accederlos en caso de contingencia. Por ejemplo, si esta cuenta no existe debe ser creada vía MTO[6] y asignadas vía SRC.
- Se deben crear diferentes configuraciones de renovación sobre el parque de dispositivos de red.

[5] ACS: Access Control Server.
[6] MTO: Make To Order.

Metodología y desarrollo de la solución

Para llevar a cabo esta solución, se utilizó el modelo de desarrollo iterativo-incremental [4], dada la experiencia y los buenos resultados obtenidos en proyectos anteriores, en donde en el primer incremento se consideraron las necesidades del módulo de renovación, mientras que en el segundo incremento se enfocó en la parte web y su integración con GUIA. Esta metodología se adopta en pro de continuar con la ya existente en el sistema GUIA y con quien finalmente se debe integrar.

Así, en la primera fase de este trabajo se identificaron los tipos de marcas de dispositivos a tratar en donde se incluyen pruebas de conectividad, esto definido previamente con el cliente interesado en el proceso de renovación. A continuación, se identificaron los dispositivos a renovar y los atributos de evaluación a la hora de automatizar el proceso. En esta fase, se desarrolló el diseño de un motor de renovación y validación de características propias de los tipos de dispositivos considerados. Esto se realizó para poder estandarizar la forma en la cual se realiza la renovación de credenciales en cada uno de los dispositivos. El producto final correspondió al módulo de Renovación (Middleware) de credenciales.

En la segunda etapa, los esfuerzos se centraron en analizar la integración con GUIA [5]. Para esto, se determinó cuáles eran los tipos de dispositivos a renovar junto con las operaciones de administración de acceso para estos dispositivos (inserción, eliminación y actualización de una Configuración de renovación). Luego, se diseñó y desarrolló la aplicación web y un Middleware de integración el cual expone una interfaz genérica de administración [3], la que es responsable de procesar todas las operaciones de administración de configuraciones de renovación. El producto final corresponde a una aplicación Web llamada Sistema de renovación de Credenciales.

Como productos finales se obtuvo una aplicación Web interoperable e integrable con otras plataformas. Esto es importante, dado que en un futuro si se desea migrar de plataforma (incluso para otros fabricantes de este tipo de plataformas) o incorporar nuevos tipos de dispositivos que requieran de renovación de credenciales, los esfuerzos se centrarán en la configuración interna del producto y en menor medida en aspectos técnicos de integración con otros sistemas ya existentes en la compañía.

Objetivos

Objetivo General

Esta tesis tiene por objetivo general proporcionar a Telefónica Chile un sistema que provea un mecanismo estándar de Configuración y Renovación de credenciales para cada uno de los dispositivos de red, aún en caso de contingencia. Con esto, se espera aumentar la seguridad en los accesos hacia los dispositivos y a la vez reducir los ataques sufridos en redes de clientes de Telefónica.

Objetivos Específicos

Dentro de las características específicas que debe tener este trabajo, se considera lo siguiente:

- Diseño e implementación de un mecanismo de configuración de periodicidad de la renovación de credenciales.
- Diseño e implementación de un mecanismo de renovación automática de credenciales.
- Diseño e implementación de una funcionalidad de representación gráfica del estado del proceso de renovación para una selección de dispositivos.
- Diseño e implementación de un mecanismo de logging de los eventos asociado a la renovación.
- Diseño, implementación e integración de los distintos mecanismos en el módulo de manejo de credenciales a fin de centralizar la operación.

Alcances y Limitaciones

En el desarrollo de esta tesis, sólo se abarcó el sistema SRC compuesto por un módulo web y un módulo que permite el desarrollo de tareas programadas, además de la integración en la plataforma GUIA. Los componentes de software propios de GUIA no son abarcados dentro de este trabajo, sólo son mencionados y explicados para dar contexto al trabajo. Los temas propios de GUIA fueron tomados por el resto del equipo de trabajo,

así este trabajo sólo se enfocó en la creación del sistema de renovación de credenciales.

Algunas consideraciones que se debieron considerar para el desarrollo del SRC:

- El SRC sólo considerará dentro de su proceso de renovación a aquellos dispositivos que cumplen con los requisitos previos, es decir están conectados vía AAA[7] con el ACS [6] mediante protocolo TACACS [7].
- Aquellos dispositivos que no cumplen con los requisitos de renovación, seguirán siendo tratados de la misma forma que hasta ahora se hace, por ende no tendrán renovación de credenciales. Esto debido a que no todos los dispositivos usados, cuentan con interfaces que permitan tratar sus configuraciones de forma automática mediante programas creados para ello.

Es primordial controlar los accesos de forma centralizada, mantener registros de las renovaciones realizadas y aumentar los niveles de seguridad. Los servicios de red considerados fueron:

- Acceso a diferentes dispositivos de la infraestructura de red.
- Acceso a Sistemas de Negocio Web.

Adicionalmente, los diferentes servicios de red requeridos por el SRC, debieron suscribirse a los procedimientos provistos por GUIA. Ejemplo de esto es el proceso de autenticación llamado Single Sign-On (SSO) [8] implementado mediante un Active Directory (AD)[9], que viene a ser la misma fuente autoritativa de usuarios con la cuenta GUIA. Asimismo, la arquitectura del SRC debe seguir los lineamientos entregados por la plataforma GUIA para futuras integraciones y/o extensiones.

[7] AAA: Authentication, Authorization and Accounting Protocol.
[8] SSO: Es un procedimiento de autenticación que habilita al usuario para acceder a varios sistemas con una sola cuenta de acceso.
[9] AD: Término que utiliza Microsoft para referirse a su implementación de servicio de directorio en una red distribuida de computadores. Utiliza distintos protocolos, principalmente LDAP, DNS, DHCP y Kerberos.

La Plataforma GUIA como antesala

Antecedentes

Tal como se nombró en la introducción de este documento, la adquisición de la plataforma OIM se concibió con el objetivo de administrar todo el ciclo de vida de las cuentas de usuarios además de controlar el acceso a los diferentes recursos corporativos. Así, su funcionamiento se enfoca principalmente en la protección en el otorgamiento de accesos a sistemas, reducción del trabajo operativo del personal soporte y a la automatización de procesos de negocio mediante reglas de aprobación y políticas de otorgamiento de acceso [8]. Un ejemplo de esto es cuando una persona es desvinculada de la organización; OIM lo detecta desde las bases de datos de recursos humanos y elimina automáticamente todos sus accesos. De la misma manera ocurre cuando una persona es contratada, OIM detecta la incorporación y según su cargo, unidad organizacional, lugar de trabajo que pertenece le otorga las cuentas de accesos a los sistemas que le corresponde según su perfil. Así, uno de sus objetivos es centralizar los mecanismos de acceso y disminuir el trabajo operativo de los profesionales que mantienen la plataforma.

Si bien OIM como plataforma resuelve una gran problemática, persistían necesidades importantes dentro del área, necesidades que OIM por sí solo no era capaz de resolver. Por ejemplo la administración de dispositivos de red, incorporación de personal temporal (Consultores) que requiere trabajar sobre las redes y/o dispositivos de red entre otras. Ante esto nace GUIA, plataforma de gran envergadura que envuelve la plataforma de OIM y da solución a las necesidades antes expuestas.

Sistemas Involucrados en GUIA

La Plataforma GUIA toma como base el objetivo de OIM que corresponde a mantener y controlar el acceso de los usuarios hacia los recursos de la organización además de mantener un control centralizado del personal de la compañía. Pero dado que debe responder a necesidades que OIM no cumple, es que se crearon subsistemas que en conjunto y de forma integrada conforman GUIA. Estos son brevemente explicados a continuación:

Sistema MTO

Permite la administración de dispositivos de red y de todos los elementos necesarios que complementan a los dispositivos. Este a su vez se comunica con el sistema persistente de control de acceso a los dispositivos, en adelante ACS[10], para depositar la información asociada. Finalmente es el ACS quien tiene la comunicación directa con el dispositivo de red y quien da la autorización de operación sobre estos mismos.

Sistema NDU

Permite homogeneizar los datos de usuarios obtenidos desde la distintas fuentes de datos, esto se da ante el hecho que existen fuentes divergentes entre si y las cuales se encuentran distribuidas en toda la organización enfocadas a responder a las necesidades del negocio de los departamentos a los cuales pertenece. El objetivo principal del NDU[11] es centralizar y estandarizar la información de los usuarios de la compañía y de esta forma facilitar el acceso de los mismos al sistema GUIA.

Sistema ACU

Permite aprovisionar cuentas de usuarios para obtener acceso al sistema. En este caso mediante una solicitud, se validará que el usuario a quien se le está solicitando acceso existe dentro de la fuente autoritativa, en este caso el Active Directory de la organización, y de ser así mientras esté autorizado podrá operar en el sistema GUIA.

Sistema Perfilamiento

Tal y como lo indica su nombre permite perfilar las opciones dentro del sistema GUIA, restringiendo las vistas que podrán tener los distintos perfiles, además de poder administrar las asignaciones.

GUIA en Telefónica LATAM

La Dirección de Seguridad de la Información es la encargada de velar por la seguridad tecnológica integral de todas las empresas del Grupo Telefónica. Esta unidad organizacional nace en el año 1984 y desde hace más de 20 años se encuentra entregando soluciones innovadoras y servicios especializados en TI. Actualmente, tiene presencia directa en Chile, Perú,

[10] Access Control Server.

[11] Normalizador de Datos de Usuario.

España, Argentina, Brasil y México y tiene planes de crecimiento en otros países de Latino América.

En el año 2008, el departamento de seguridad de la información inicia su participación en la iniciativa global de Gestión de Identidades dentro del Holding Telefónica. En este sentido Chile, Perú, México y Colombia han sido pioneros en implementar localmente la solución OIM, pero Chile es pionero en tener una solución integral de gestión unificada de accesos y que además tiene la versatilidad de integrar nuevos sistemas a su operatoria diaria sin afectar lo existente. En este sentido no solo se habla de dar acceso a sistemas independientes sino a la capacidad de GUIA de integrar nuevos sistemas como parte propia. Los antecedentes en cuanto a cantidad de usuarios y sistemas integrados de forma independiente en cada país se presenta en **Tabla 1** solo con OIM[12].

COMO SE PUEDE APRECIAR EN

Tabla 1, Chile lidera en cuanto a la cantidad de usuarios que se encuentran registrados en sus fuentes autoritativas de datos seguidos por Colombia y Perú. Por otro lado, en cuanto a la integración de aplicaciones, la división de Telefónica que tiene la mayor experiencia es Colombia, seguido por Chile y México. Es importante reiterar que esta solución tuvo como principal objetivo poder dar una mejor administración a las cuentas de accesos de usuarios y a su provisión dentro de la compañía.

Nro.	País	División	N° de Usuarios	Aplicaciones
1	Chile	Telefónica Chile	60.000	6
2	Perú	Telefónica del Perú	12.000	2
3	México	Telefónica México	11.000	5
4	Colombia	Movistar Colombia	18.000	37

TABLA 1: OIM 9 IMPLANTADO EN TELEFÓNICA LATAM

[12] Oracle Identity Manager.

Arquitectura de GUIA

El sistema GUIA es la plataforma responsable de lo siguiente [9]:

- Gestionar de manera eficiente y unificada las identidades de los profesionales que trabajan en la unidad organizativa de redes, empresas, proveedores, mesas tecnológicas, clientes. Esto en detalle, involucra:
    - Sincronización con procesos de la línea operativa.
    - Acceso Lógico a los Sistemas y Recursos de Red
    - Acceso Físico a los Sitios Críticos definidos por la Dirección de Red.
- Gestionar y controlar los procesos de Incorporación y desvinculación de personas. Esto en mayor detalle, involucra:
    - Automatización de procesos mediante reglas de negocio.
    - Disminución de los aprovisionamientos de cuentas a los diferentes recursos de red.

La arquitectura que en parte soporta lo anteriormente especificado se muestra en la Figura 5

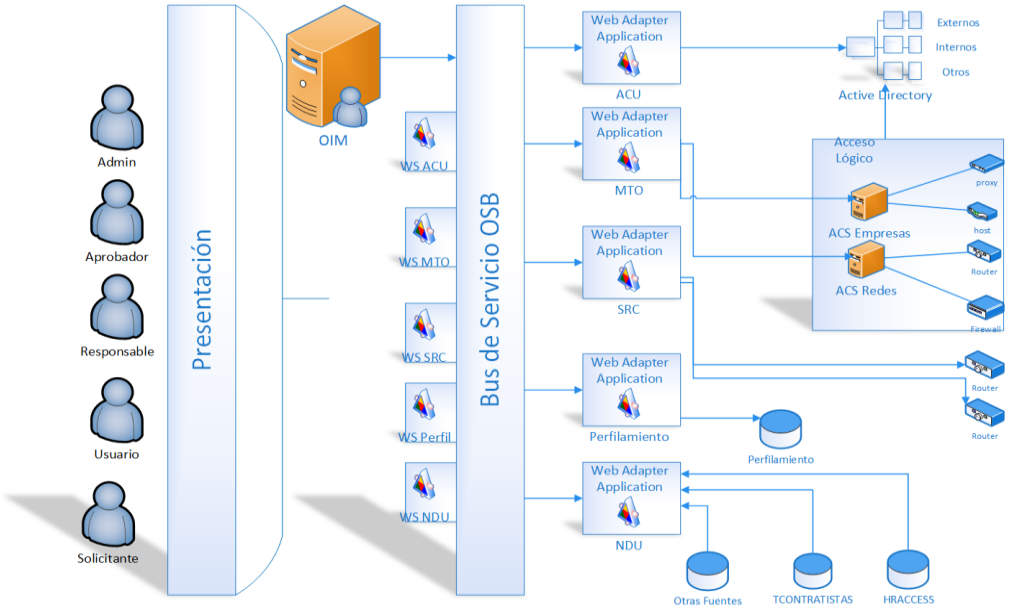

FIGURA 5: ARQUITECTURA GUIA

Revisión del Mercado

En la actualidad, para las grandes empresas, el tema de gestión cobra relevancia debido a que se están produciendo pérdidas de grandes cantidades de dólares cada año por este concepto, debido a la deficiente administración que se tiene sobre las cuentas de usuario tanto a nivel aplicación como a nivel de unidades, como lo son los dispositivos de red. Para afrontar este problema, las compañías han estado invirtiendo cada vez más en soluciones TI que permitan cubrir las principales necesidades, y en algunos casos deficiencias existentes, esperando disminuir los costos de esta actividad y garantizar la seguridad de acceso sobre los activos en toda la organización.

Adicionalmente, y revisando algunos sistemas del mercado, se ha podido detectar que en parte cubren algunas necesidades pero no todas. Un ejemplo de esto es la creación de scripts que permiten cambiar la configuración de router y switch dentro de una red. Algunos de los incidentes asociados a estas creaciones son:

- Fraudes
- Obtención de información confidencial
- Sabotaje

Para poder tener un control efectivo, dentro de cada organización, se requiere de políticas, procedimientos y tecnología que permita mantener un control de usuarios y sus respectivos privilegios, cosa en donde muchos fallan. Las principales fallas se asocian a:

- Administración de contraseñas
- Control de acceso
- Identificación de accesos

Las principales marcas del mercado muchas veces fallan en más de una categoría, de las antes presentadas.

A continuación en Tabla 2 se especifican algunos de estos productos o sistemas.

Productos o Sistemas		Proveedor
RFP	Control de acceso a Elementos de Red [10]	Ericsson
NAK	Network Access Management Solution of Nakina Systems [11]	Nokia Siemens
PAM	Administración y Control de acceso de cuentas privilegiadas [12]	DELL
NAG	Nokia Access Guard [13]	Nokia
CTX	Citrix [14]	CITRIX

TABLA 2: SISTEMAS DE LA COMPETENCIA

De los productos/sistemas antes mencionados es posible mostrar un cuadro comparativo el cual identifica algunas funcionalidades respecto de sus competencias, ver Tabla 3.

Función	RFP	NAK	PAM	NAG	CTX	Comentarios
Acceso Remoto	OK	OK	OK	OK	OK	
Registro de accesos vía Logs u otro	OK	OK	OK	OK	X	
Administración de passwords	OK	OK	OK	OK	X	Si bien la mayoría ofrece administrar password estas según la propia descripción del producto es a demanda. Es decir si el administrador lo solicita esta se renovará.
Administración de comandos para variedad de dispositivos	X[13]	X	OK	OK	X	Los sistemas acá referencian la administración de comandos como aquellos que se pueden o no ejecutar sobre los dispositivos de red. No tenemos evidencia de que podamos Ingresar una nuevo

[13] Sin evidencia

						tipo de dispositivos con una nueva línea de comandos y esta sea posible ejecutarla.
Integración flexible	X	OK	X	X	X	Nakina ofrece flexibilidad con los módulos NI-Guardian y NE-Security [11], provisión de cuentas y administración de credenciales de dispositivos respectivamente.
Cambio automático de password	X	X	X	X	X	Sin antecedentes declarados.
Administrar agrupaciones para cambio de passwords	X	X	X	X	X	Sin antecedentes declarados.

TABLA 3: CUADRO COMPARATIVO SISTEMAS DE LA COMPETENCIA.

Estudios realizados por compañías como Ericsson, hacen énfasis en la automatización de las diversas configuraciones que involucra el armado de una red con cada uno de sus componentes. Estos esfuerzos van enfocados a proveer aplicaciones web que permitan de forma fácil y rápida además de seguras y confiables el poder configurar una o varias redes manteniendo la data asociada a la red de forma centralizada.

Este tipo de soluciones hacen alusión a que actualmente las configuraciones de los router/switch de una red se lleva a cabo uno por uno teniendo que inicializarlos, mantenerlos y darlos de baja en caso de requerirse. Esto conlleva altas probabilidades de errores que repercuten en el "performance" de la red. Este tipo de soluciones tienen como principal componente un entorno web que permite la modificación en línea de las configuraciones de red [27].

De acuerdo a lo expuesto con anterioridad, podemos mencionar que si bien en el mercado hay una variedad muy amplia de sistemas / aplicaciones que cumplen en parte con algunas de las necesidades dadas para nuestro sistema SRC, podemos indicar que el sistema SRC tiene características propias que la evaluación de mercado no las provee. Una de sus mayores ventajas es la administración de credenciales que serán usadas sólo en caso de falta de conectividad, falla en ACS o similar, evento por el cual a los

usuarios se les imposibilitará el uso de sus credenciales de red y solo se podrán usar las creadas y administradas bajo el sistema SRC, además su creación bajo licencia tipo "open source" da ventajas por sobre las aplicaciones o servicios entregados por las entidades privadas antes mencionadas.

Desarrollo de la solución: Construcción de un Sistema de Renovación de Credenciales

En este capítulo, primeramente se describen y detallan las principales áreas involucradas en el proceso de renovación de credenciales que involucran a la compañía Telefónica Chile, junto con considerar sus restricciones, características de acceso y mecanismo de funcionamiento. Así mismo, se determinan los atributos a considerar dentro del proceso de renovación. Adicionalmente se explica el diseño e implementación del sistema SRC para cada una de sus capas.

Antecedentes

Administración de Configuraciones de Renovación

La administración de configuraciones de renovación vista como un proceso, se puede apreciar en la Figura 6.

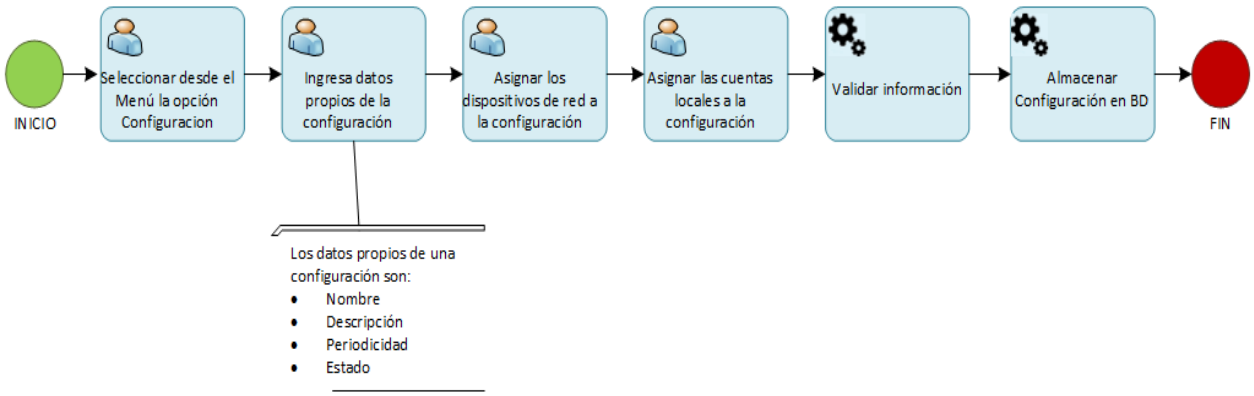

FIGURA 6: ADMINISTRACIÓN DE CONFIGURACIONES DE RENOVACIÓN.

A continuación se explican los pasos del procedimiento de administración de configuraciones:

- El proceso comienza cuando el usuario accede a la aplicación SRC y se dirige a la sección de administración.
- En la sección de administración, el usuario ingresará la información requerida.
- El sistema validará y almacenará la información previamente ingresada.

Procesos de Renovación de credenciales

El proceso de renovación de credenciales de los dispositivos de red del sistema SRC, se puede apreciar en la Figura 7.

A continuación se explican los pasos del procedimiento de renovación de credenciales:

- El proceso comienza cuando la aplicación SRC obtiene las configuraciones de renovación existentes.
- Se procesa cada configuración de renovación extrayendo los dispositivos de red asociados y se aplica un conjunto de validaciones, entre las que cuentan: usuarios conectados al dispositivo y la marca. Adicionalmente da de baja las cuentas locales de cada uno de los dispositivos de red.
- Posteriormente se envían las nuevas cuentas locales hacia el dispositivo, se evalúa la respuesta y se almacena la información en el repositorio local.
- Se mantiene historia de cada renovación ejecutada.

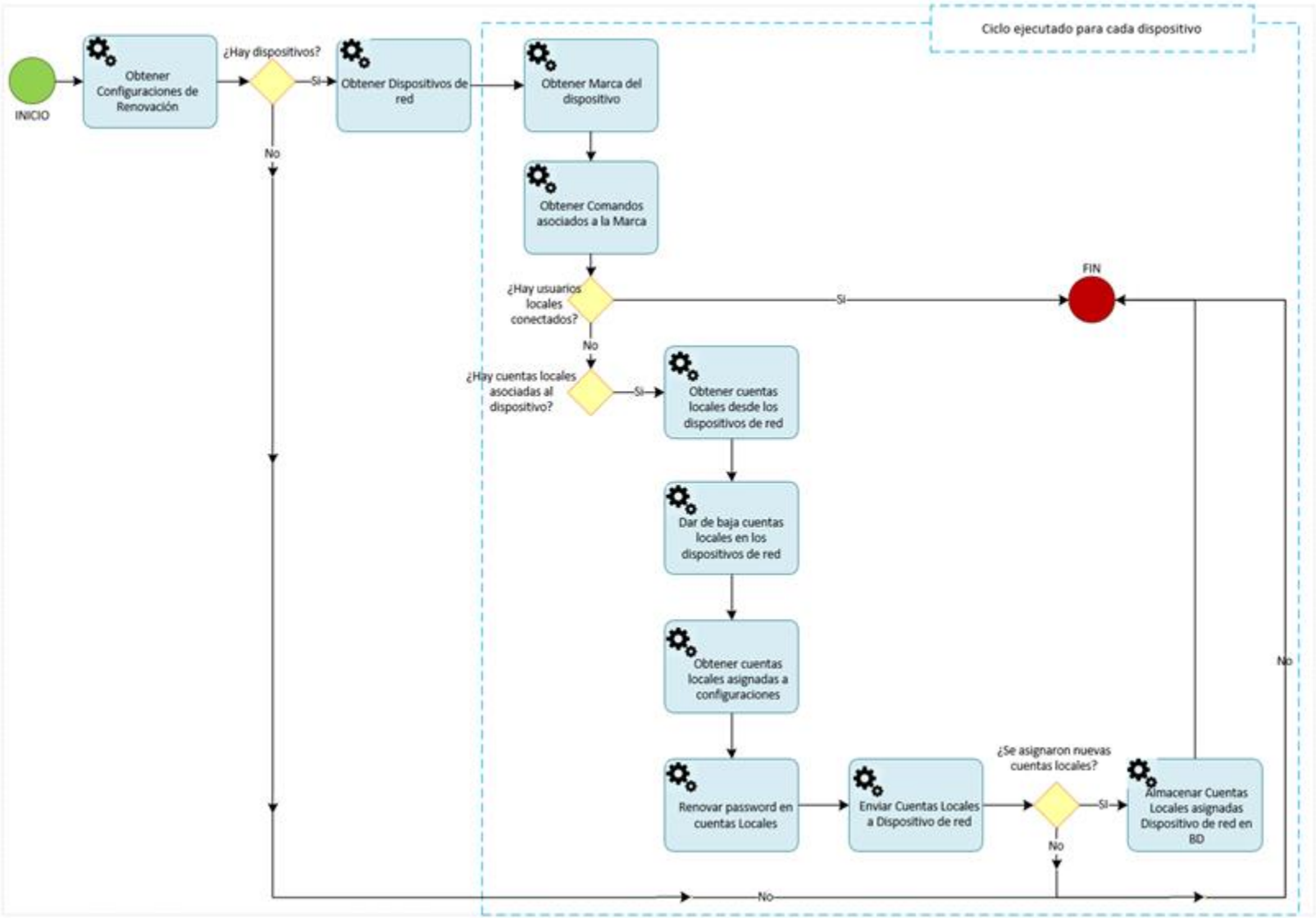

FIGURA 7: PROCESO DE RENOVACIÓN DE CREDENCIALES.

Arquitectura de SRC

El sistema SRC es una aplicación empresarial JEE[14] [15] responsable de administrar las configuraciones de renovación y ejecutar los procesos de renovación de forma automática y autónoma. Este sistema se compone de tres capas, como lo muestra la Figura 8, entre las que cuentan:

- **Capa de Presentación**: Compuesta por todos los elementos relacionados con la interfaz de usuario, tales como botones, ventanas, campos de textos, etiquetas, etc. Ésta es la capa de interfaz de usuario, el rostro visible y, en cierta forma, estético del sistema. Sin lugar a dudas, la parte más importante para el usuario final del sistema.
- **Capa de Servicios**: Es el punto de entrada a la lógica de negocios, la cual se encuentra encapsulada a través de un conjunto de componentes de servicio. Esta capa no almacena información de clientes particulares, de esta forma se mantiene completamente desacoplados de éstos, además de que una instancia puede atender múltiples peticiones y así es posible brindar un mayor desempeño.
- **Capa de Persistencia**: Al hablar de persistencia hablamos de almacenamiento de información en un medio no volátil (bases de datos, archivos de texto plano, etc). Los elementos de esta capa típicamente funcionan como objetos de acceso a datos o DAOs[15], proveyendo métodos para leer y almacenar los datos de los objetos de negocio.

[14] Aplicación JEE: Es una aplicación empresarial desarrollada en el lenguaje de programación Java.
[15] Data Access Objects.

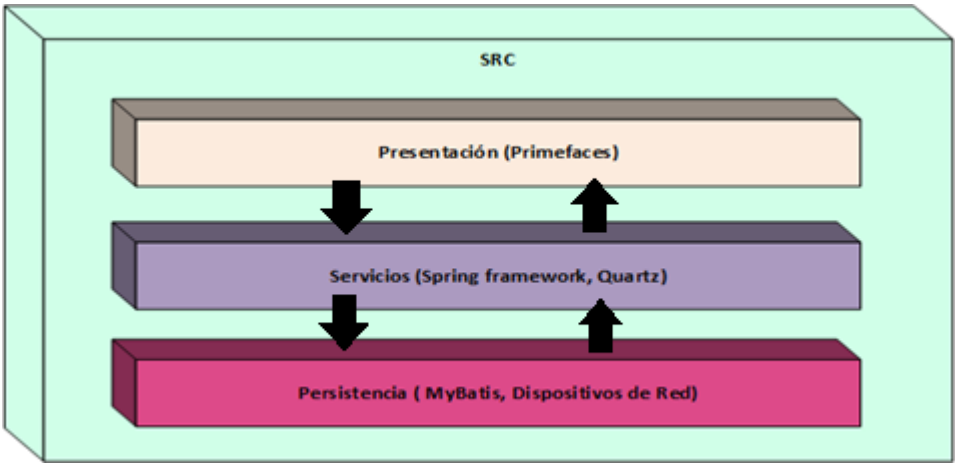

FIGURA 8: ESQUEMA GENERAL DE COMPONENTES SRC.

Tomando en cuenta las preocupaciones de mantenimiento y de interoperabilidad, este sistema fue diseñado en una arquitectura de 3 capas e implementado en la tecnología JEE, acorde a las normativas de construcción de software vigente en la compañía.

Arquitectura por Capas

Diseño de la Capa de Presentación

La capa de presentación corresponde a un módulo Web en el cual se visualizan por un lado la administración de las configuraciones de renovación y por otro lado una búsqueda, la cual dependiendo del perfil que la acceda podrán visualizarse las credenciales de cada uno de los dispositivos de red asociados.

- Acceso al sistema: esta sección es provista por GUIA, permitiéndonos acceder al sistema mediante un nombre de usuario y una password, estas últimas contenidas en la entidad persistente AD[16]. La Figura 9 muestra la interfaz de acceso.

[16] Active Directory.

FIGURA 9: ACCESO AL SISTEMA.

➢ Administración de Configuraciones: En este módulo se muestran las configuraciones de renovación existentes, además de las opciones de trabajo sobre las mismas (inserciones, actualizaciones y eliminaciones), ver Figura 10. Al seleccionar alguna de la tabla de resultados se puede visualizar tanto los dispositivos de red asignados previamente como las cuentas locales que permitirán llevar a cabo la renovación de las mismas en cada uno de los dispositivos pertenecientes a la configuración seleccionada.

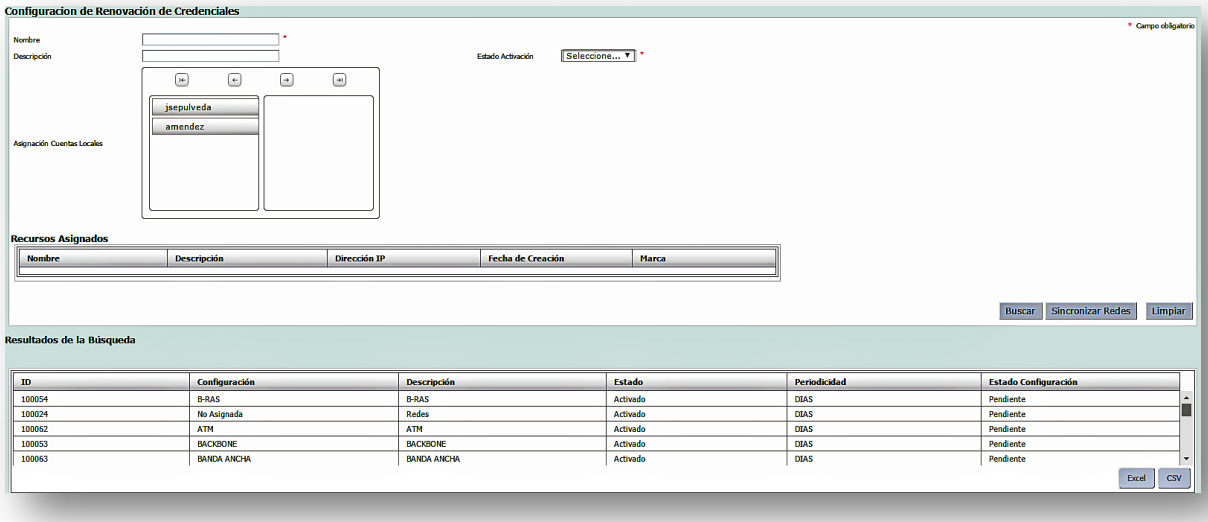

FIGURA 10: ADMINISTRACIÓN DE CONFIGURACIONES.

➢ Búsqueda de Configuraciones: En este módulo se permite realizar búsqueda de configuraciones y al seleccionar alguno de los registros de la tabla resultante podemos obtener la lista de dispositivos de red que se encuentran asociados además de las cuentas locales asociadas a estos últimos post renovación de credenciales, la Figura 11 muestra la interfaz asociada.

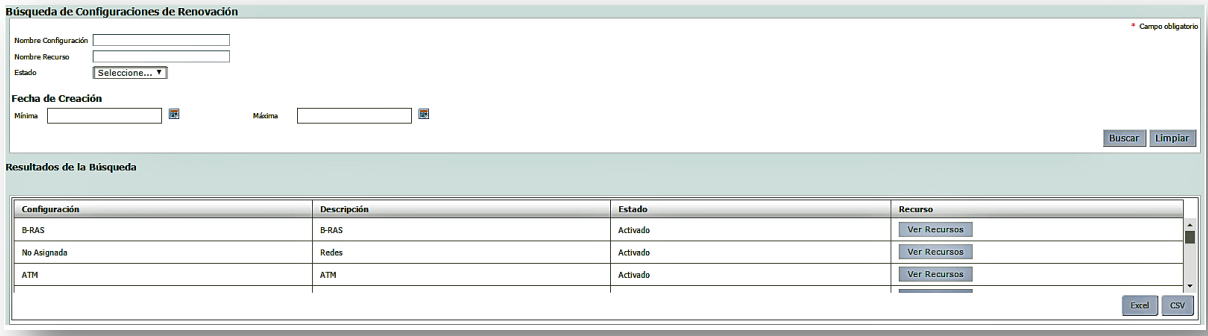

FIGURA 11: BÚSQUEDA DE CONFIGURACIONES.

Complementariamente, el sistema MTO proveerá de la información básica para operar con las configuraciones. Entre los elementos que este sistema nos provee, tenemos lo siguiente:

➢ Administración de Cuentas Locales: Este módulo nos permite insertar, actualizar y eliminar cuentas locales. Estas se componen principalmente de un nombre de usuario y una contraseña, esta última es variada por el proceso de renovación a la hora de asignar la cuenta local a una configuración de renovación. La Figura 12 muestra la interfaz asociada.

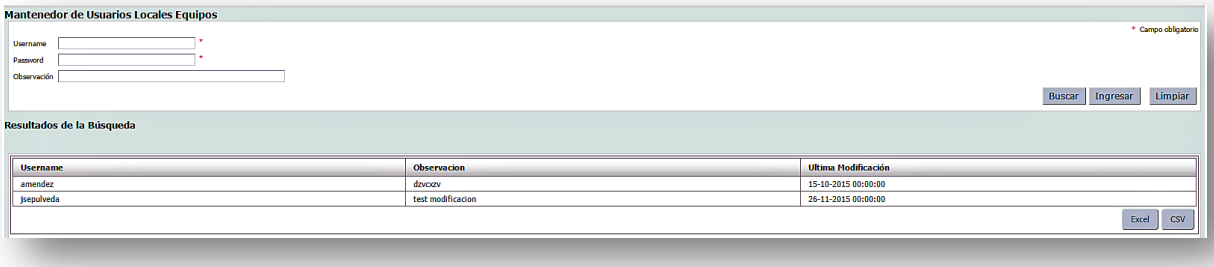

FIGURA 12: ADMINISTRACIÓN DE CUENTAS LOCALES.

➢ Administración de Dispositivos de Red: Este módulo nos permite insertar, actualizar y eliminar dispositivos de red, la Figura 13 muestra esto. Los dispositivos de red son enviados hacia el componente ACS que forma parte de GUIA y adicionalmente se almacenan de forma local. Es importante mencionar que es el ACS quien toma el rol de autorizador a la hora de querer acceder directamente los dispositivos de red.

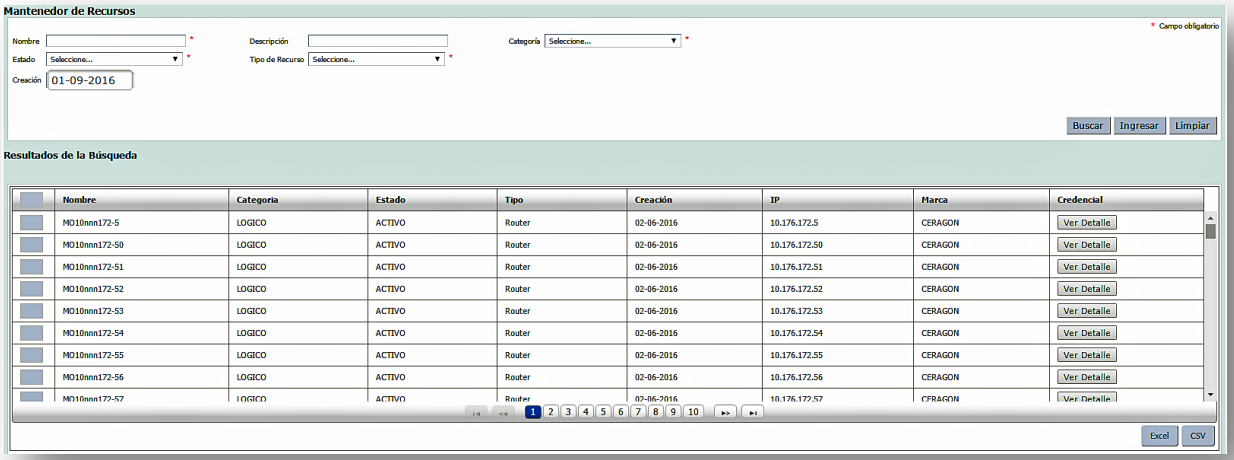

FIGURA 13: ADMINISTRACIÓN DE DISPOSITIVOS DE RED.

Implementación de la Capa de Presentación

Para la implementación de la capa de presentación, se utilizó el framework PrimeFaces V5.0 [16] basada en la especificación JSR 344 [17]. Se optó por utilizar una tecnología opensource dada su alta usabilidad, interacción y mejoramiento de la experiencia del usuario, la que hoy es ampliamente aceptada en la industria según el Ranking del portal DevRates[17] [18]. Otros beneficios que destaca particularmente el fabricante acerca de este producto son: flexibilidad, rendimiento, portabilidad e integración con otros marcos de trabajo, tomando como referencia la tecnología Java [16] Cabe señalar que este marco de trabajo utiliza como patrón arquitectónico el Modelo-Vista-Controlador (MVC):

- La vista está compuesta por páginas XHTML con componentes visuales que embeben comunicación Ajax mediante JQuery. Esto favorece el rendimiento de la interfaz web, dado a que solamente se actualizan componentes visuales requeridos y no toda la página web.
- El modelo está compuesto por clases Java llamadas ManageBean. Estas entidades son las encargadas de gestionar el envío y recepción de información hacía la vista. Para realizar el intercambio de información entre la vista y los ManageBean, se utilizan artefactos especiales llamados Data Tranfer Object[18].
- El controlador de este framework trabaja mediante un archivo XML, el cual especifica los flujos de navegación que tiene la aplicación. Adicionalmente este componente se puede representar mediante anotaciones especiales, las que son interpretadas por el componente Servlet Controller Faces.

A continuación en la Figura 14 se muestra el esquema general del funcionamiento de este modelo.

[17] DevRates: Sitio Web que valora las tecnologías según las experiencias de los desarrolladores.
[18] DTO: Entidad que encapsula los datos de negocio para optimizar la comunicación entre capas.

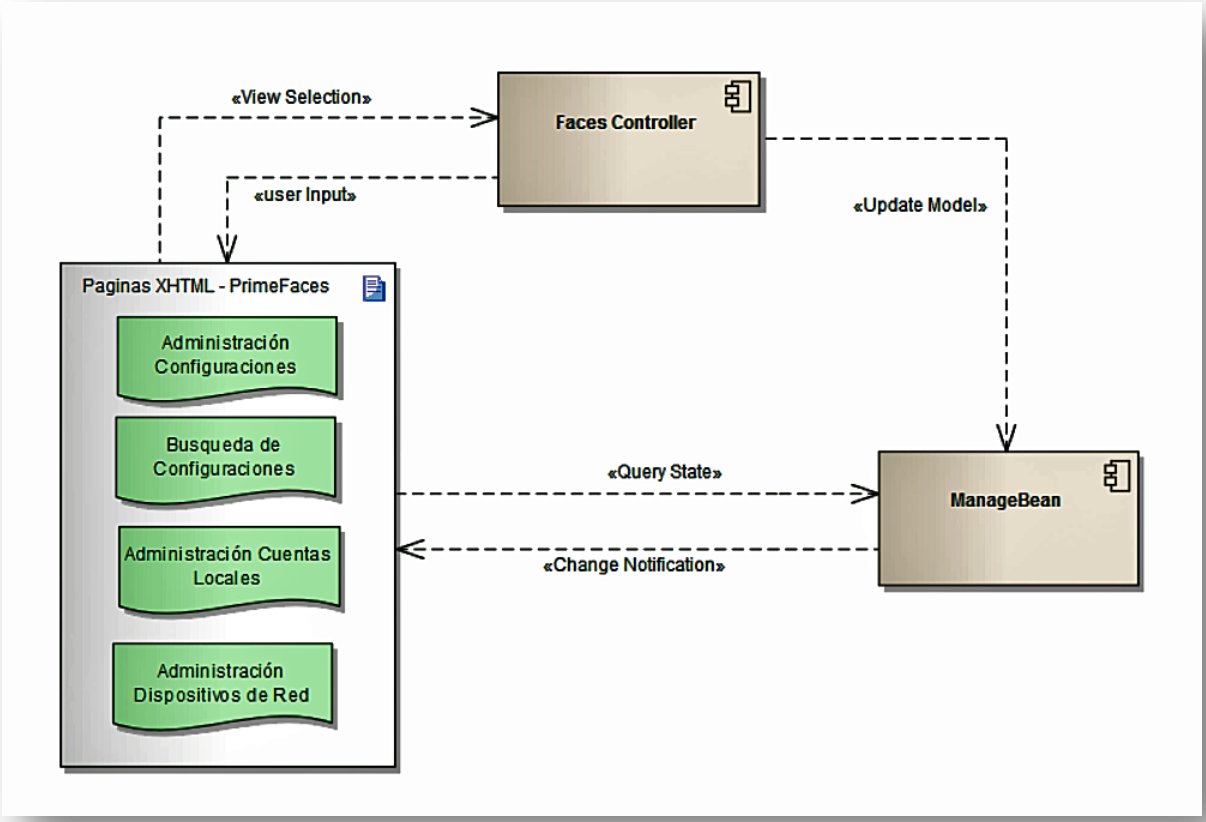

FIGURA 14: FUNCIONAMIENTO DEL FRAMEWORK PRIMEFACES.

Diseño de la Capa de Servicios

La capa de servicios especifica la lógica y reglas de negocio que contiene las principales funciones de la aplicación. En el caso del middleware del SRC, este es el encargado de ejecutar las siguientes actividades:

- Controlar y coordinar las tareas de renovación de credenciales.
- Presentar los resultados del proceso de renovación mediante servicios de consultas.
- Controla y coordina las tareas de administración de configuraciones.
- Presenta la información necesaria para administrar las configuraciones de renovación.

Los servicios participantes en esta capa son:

- Servicio de Renovación de credenciales (RenovationSchedulerService).
- Servicio de Administración (ProcessDataService).

A continuación se describen las responsabilidades de los servicios antes mencionados, que forman parte de esta capa.

Renovation Scheduler Service

Es el servicio encargado de iniciar periódica y automáticamente el proceso de renovación de credenciales. Cabe señalar que la periodicidad y horario de ejecución para las tareas programadas de este módulo son parametrizables mediante un archivo de propiedades. La Figura 15, presenta todos los elementos del diseño de este módulo.

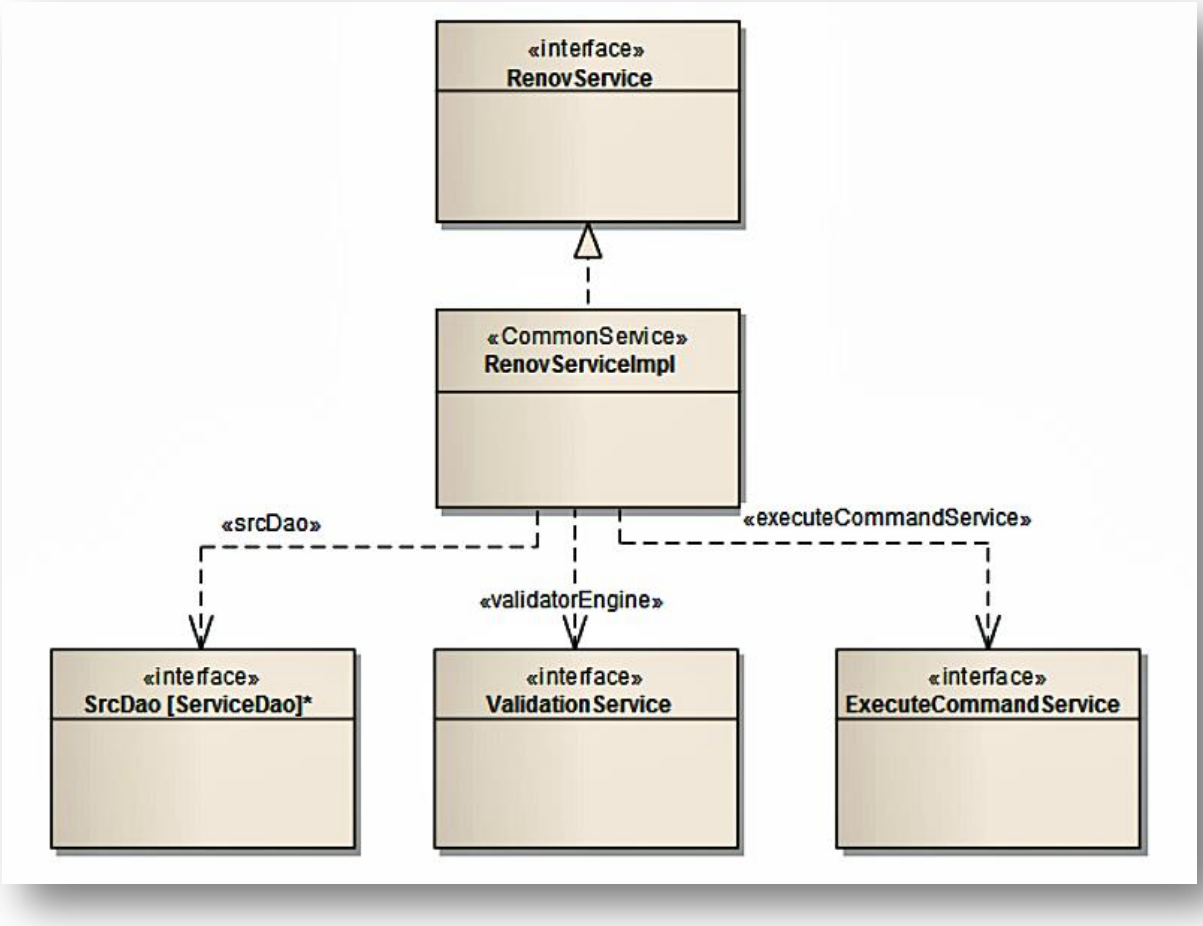

FIGURA 15: IMPLEMENTACIÓN DEL MÓDULO DE RENOVACIÓN DE CREDENCIALES.

A continuación se detallan cada uno de los componentes que conforman este servicio.

Validation Service

Este componente representa el motor de validaciones a la hora de llevar a cabo el proceso de renovación. Este servicio tiene varios objetivos dentro de sus labores de validación, entre las que cuentan:

- La versión del firmware del dispositivo
- La marca del dispositivo.
- Si hay usuarios conectados.
- Si el dispositivo de red posee cuentas locales
- Las respuestas que nos entrega el dispositivo para cada uno de los comandos ejecutados.

La interfaz de este componente recibe los dispositivos de red y entrega un conjunto de respuestas asociadas a cada ítem de validación. En el caso que no se encuentren errores, se llama al componente de ejecución de comandos. La Figura 16, presenta el diagrama de clase del componente de validación. En este caso la interfaz definida es única y sólo varía su implementación dependiendo de la marca del dispositivo considerada dentro de la validación.

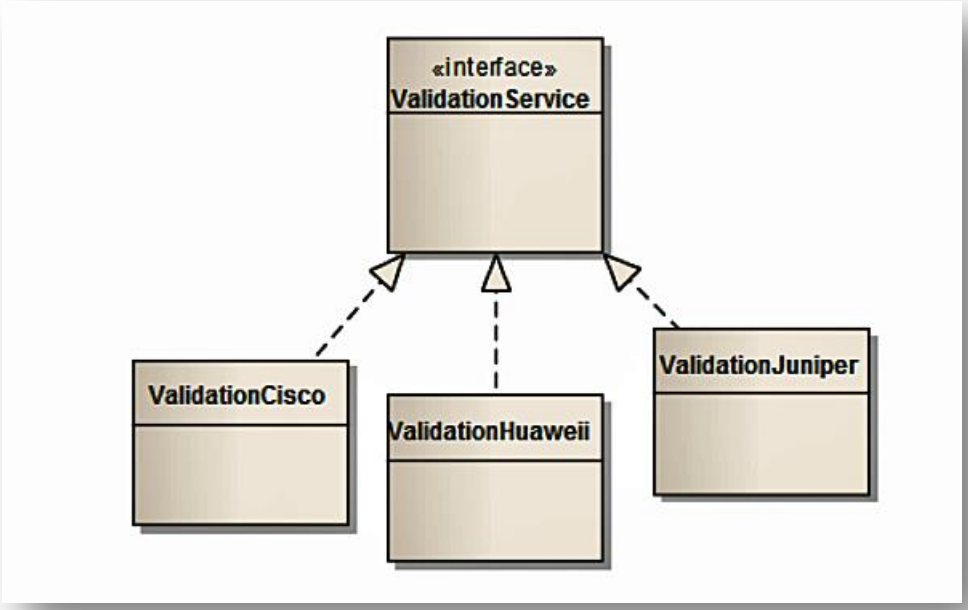

FIGURA 16: IMPLEMENTACIÓN DEL MÓDULO DE VALIDACIÓN DE DISPOSITIVOS DE RED.

a) ExecuteCommandService

Este componente es el motor de la ejecución de comandos a la hora de llevar a cabo el proceso de renovación. Este servicio tiene como principales tareas, lo siguiente:

- Obtener todos los dispositivos por configuración
- Convoca al servicio de validación
- Da de baja las cuentas locales existentes
- Obtiene las cuentas locales de los dispositivos a renovar y les asigna una nueva password.
- Da de alta a cada una de las cuentas locales asignadas a los dispositivos de red.
- Analiza respuestas entregadas por los dispositivos
- Genera histórico de renovación.

De forma similar, para el componente de validación se definió una única interfaz y sólo se varió la implementación de ésta, asociada a cada una de las marcas consideradas en el proceso. Esto deja la ventana abierta para futuras marcas en caso de requerir incorporar una nueva. En la Figura 17, se presenta el diagrama de clase del componente de ejecución de comandos de renovación.

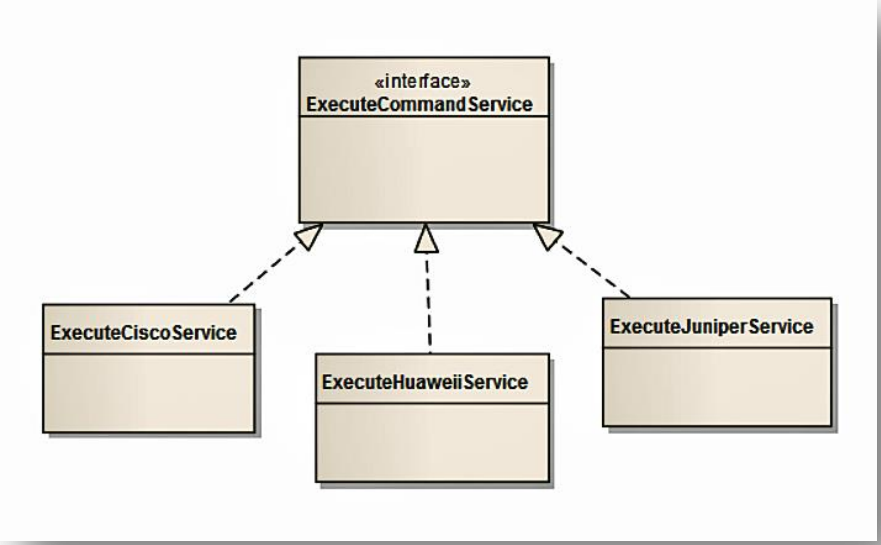

FIGURA 17: IMPLEMENTACIÓN DEL MÓDULO DE EJECUCIÓN DE COMANDOS.

b) SrcDao

Este componente es el encargado de orquestar las interacciones con la base de datos, por ende si algún servicio necesita almacenar o consultar información, es este componente el encargado de responder ante esas necesidades.

Process Data Service

Este servicio se representa en la aplicación como una Fachada[19] para el proceso de renovación de credenciales. En cuanto a su diseño, se provee una interfaz simple y desacoplada implementación, permitiendo ser modificada sin impactar otros componentes de la solución. Los componentes que forman parte de este servicio cumplen con las siguientes características: independencia, portabilidad y reutilización. En la Figura 18 se presentan todos los elementos en el diseño de este módulo, los cuales fueron especificados con anterioridad.

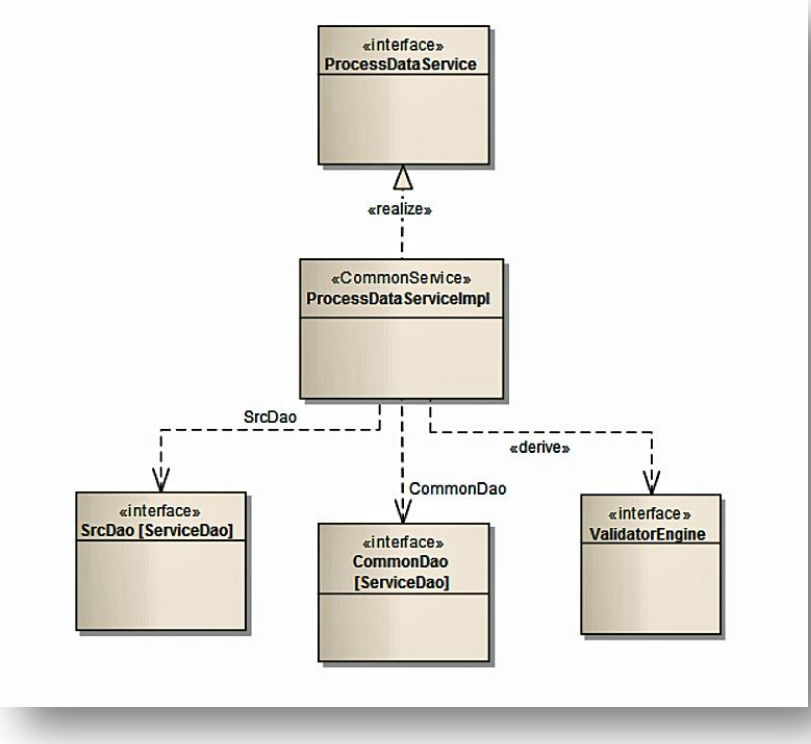

FIGURA 18: IMPLEMENTACIÓN DEL MÓDULO DE ADMINISTRACIÓN DE CONFIGURACIONES.

[19] Fachada: Patrón de diseño que proporciona una interfaz simplificada para un grupo de servicios o un sistema complejo.

En cuanto a la implementación de esta capa, todos los servicios provistos son construidos mediante el Patrón de Diseño llamado "Dependency Injection" utilizado en el Framework Spring en su versión 3.2.2 [19]. La elección de este framework para la construcción de esta capa se justifica por lo siguiente:

- El Framework Spring ofrece la flexibilidad de integrarse con tecnologías de diversos fabricantes. Por ejemplo: Hibernate, MyBatis, JPA, LDAP, etc.
- Es un framework que es constantemente actualizado. En poco tiempo ha integrado un conjunto de nuevas tecnologías y patrones arquitectónicos. Esta característica resulta importante para la construcción, ya que entrega un soporte sustentable que permite enfrentar de mejor forma las periódicas migraciones o actualizaciones que se realizan sobre el sistema.
- Es mantenible y tiene un bajo nivel de acoplamiento. De esta forma, en la aplicación sólo se utilizan servicios en forma de interfaces, aislando su implementación mediante archivos de configuración XML. Esta característica permite actualizar las implementaciones de los servicios sin afectar el comportamiento general del sistema.
- Ofrece la capacidad de poder desplegarse en distintos servidores de aplicaciones. Esto permite abstraerse del fabricante del servidor de aplicaciones para desplegar el middleware NDU, ejemplo: Websphere (IBM) o WebLogic (Oracle).

Considerando cada uno de los componentes antes expuestos en la sección de diseño que a continuación se grafica en la Figura 19 mediante un diagrama de secuencia la interacción entre estos.

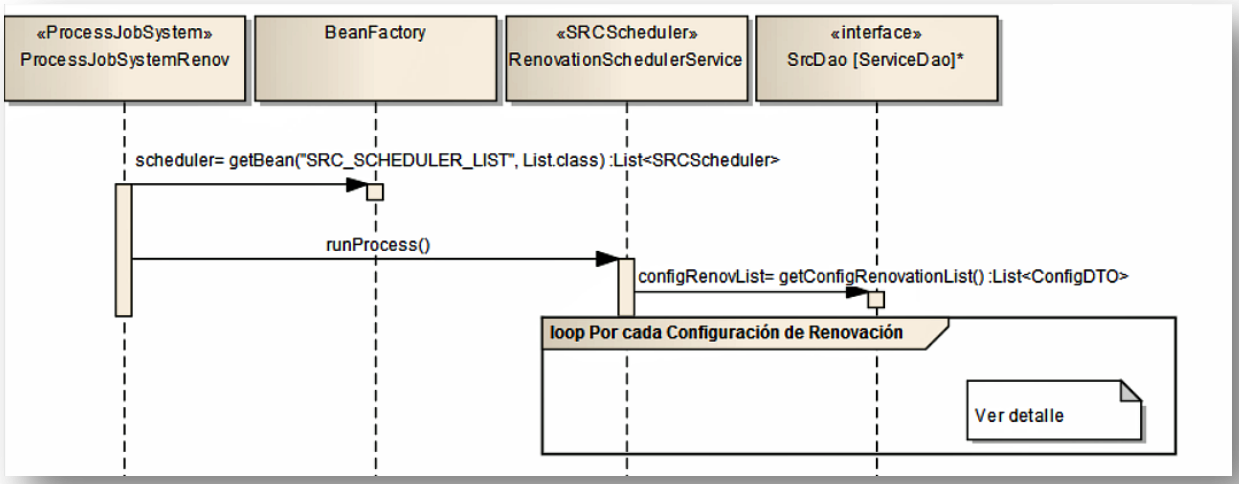

FIGURA 19: DIAGRAMA DE SECUENCIA: INTERACCIÓN ENTRE COMPONENTES. VER DETALLE EN FIGURA 20.

La Figura 19 muestra el cuadro "loop Por cada Configuración de Renovación", lo que hace referencia a que se encontrará en un ciclo que se ejecutará dependiendo de la cantidad de configuraciones que se mantengan en el sistema, el detalle asociado se muestra en la Figura 20.

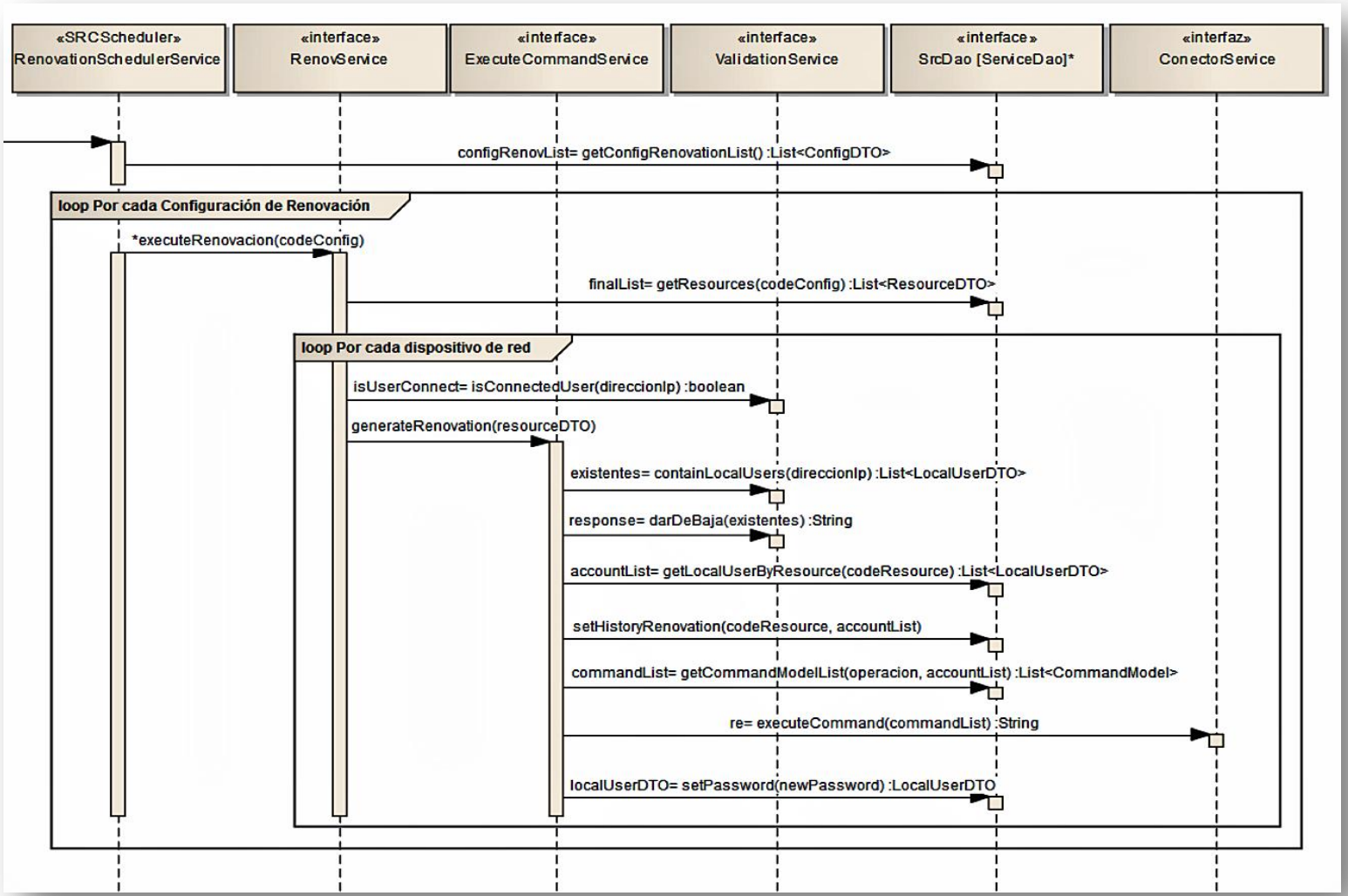

FIGURA 20: DIAGRAMA DE SECUENCIA: VER DETALLE FIGURA ANTERIOR.

Diseño de la Capa de Persistencia

La capa de persistencia de este sistema tiene como principal función almacenar los registros de configuraciones en la respectiva base de dato y disponibilizar de éstos para futuras consultas.

En el diseño de esta capa se usó el Patrón "Objeto de Acceso a Datos"[20] (DAO) el cual suministra una interfaz común entre la aplicación y el dispositivo de almacenamiento de los datos de los usuarios [15]. La mayor ventaja de utilizar este patrón es que cualquier elemento de la capa de negocio puede ser manejado sin necesidad de conocer el origen o destino de la información que provee o almacena. Esta característica permite aislar la capa de persistencia subyacente en la aplicación, la cual puede ser actualizada sin afectar a otros módulos del sistema.

Implementación de la Capa de Persistencia

Se optó por implementar la capa de persistencia, mediante la utilización de una herramienta especializada llamada MyBatis [20]. Este framework tiene la característica de mapear sentencias SQL con Objetos Java a través de archivos XML o anotaciones.

Con esto, se permite utilizar todas las funcionalidades de la base de datos, logrando modificar este mapeo cuando se produzcan las siguientes situaciones:

- Cambios en las estructuras de tablas o vistas.
- Una actualización, cambio o migración de versión de base de datos.

En la Figura 21 se puede apreciar la integración del framework MyBatis en la capa de persistencia del sistema SRC.

[20] DAO: Patrón de Diseño de Arquitectura que aísla la lógica del negocio con el sistema de persistencia que sustenta la aplicación.

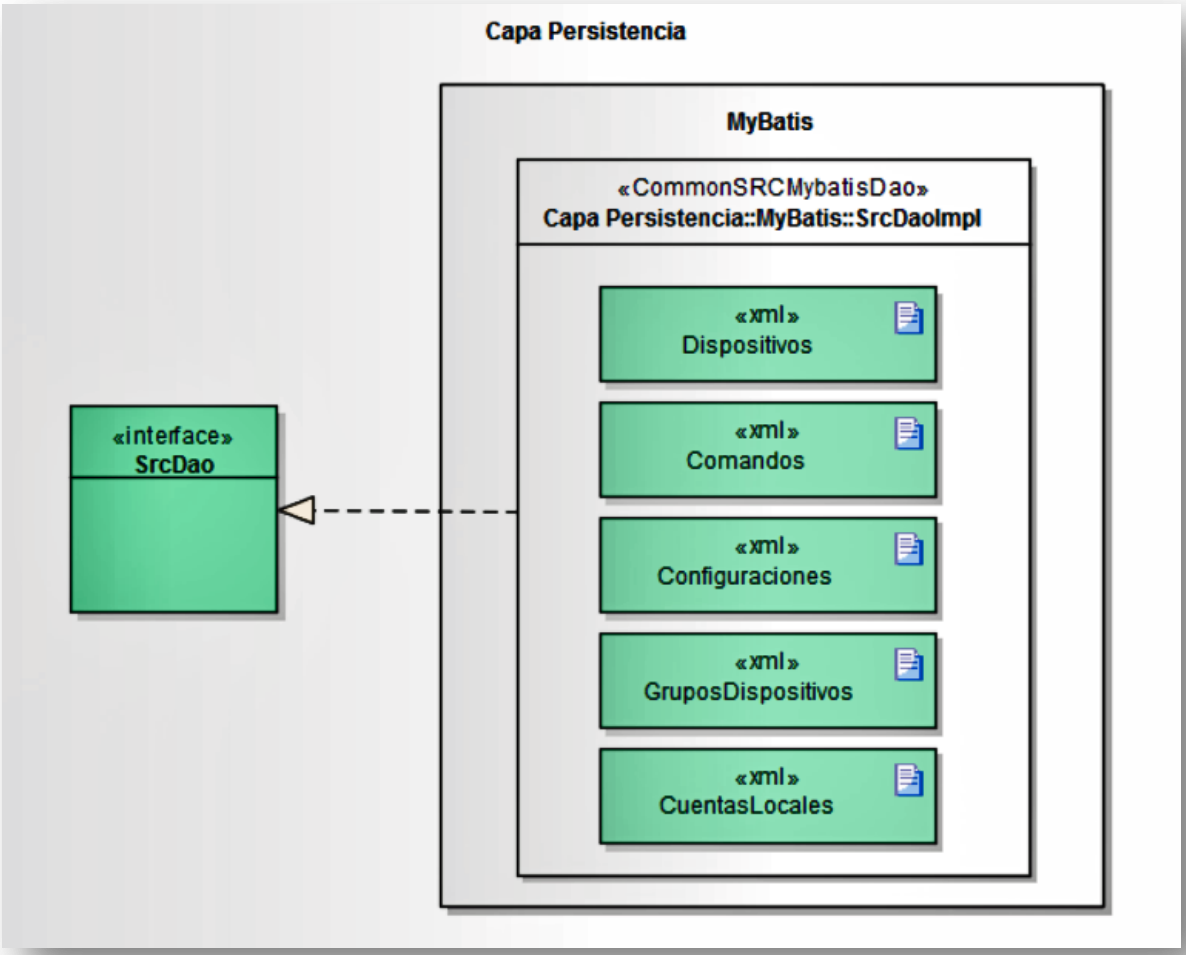

FIGURA 21: INTEGRACIÓN FRAMEWORK MYBATIS.

Como se indicó anteriormente, todas las instrucciones SQL de extracción de registros se encuentran en archivos XML. Estos archivos son interpretados y utilizados por la implementación de la interfaz DAO, la cual puede ser utilizada por cualquier servicio de la capa de negocio.

A continuación se muestra un ejemplo de cómo se obtiene una lista de DTO desde una petición SQL (SELECT). Para esquematizar esto, se toma la funcionalidad de obtener las configuraciones desde la Base de Datos. Así, en la Figura 22 se muestra el mapeo existente entre las columnas de la Query SQL y el objeto DTO, en este caso es un ConfigDTO y el mapper es identificado por configMap).

```
<typeAlias alias="configDTO" type="cl.tis.src.dto.ConfigDTO"/>

<resultMap id="configMap" type="configDTO">
    <result property="code" column="ID_CONF_RENOV"/>
    <result property="nombre" column="NOMBRE"/>
    <result property="descripcion" column="DESCRIPCION"/>
    <result property="tipo_period" column="TIPO_PERIOD"/>
    <result property="fechaCreacion" column="FECHA_CREACION" javaType="java.util.Date" jdbcType="DATE"/>
    <result property="estadoActivacion" column="ESTADO_CONF"/>
    <result property="cron" column="CRON"/>
    <result property="estadoConfiguracion" column="CONFIGURADO"/>
</resultMap>
```

FIGURA 22: MAPPER EN MYBATIS.

Adicionalmente en la Figura 23, se especifica la instrucción SELECT SQL desde donde se proveen los datos referenciados como respuesta a través del mapper de la Figura 22.

```
<select id="selectConfigDtoList" resultType="configDTO" resultMap="configMap">
    select A.ID_CONF_RENOV as "code",
           A.NOMBRE as "nombre",
           A.DESCRIPCION as "descripcion",
           A.TIPO_PERIOD as "tipo_period",
           A.FECHA_CREACION as "fechaCreacion",
           A.ESTADO_CONF as "estadoActivacion",
           A.CRON as "cron",
           A.CONFIGURADO as "estadoConfiguracion"
    from SRC_CONF_RENOVACION A
</select>
```

FIGURA 23: INSTRUCCIÓN DE SELECCIÓN EN MYBATIS.

Las ventajas de utilizar MyBatis con bases de datos existentes son las siguientes:

- Aislamiento de las instrucciones SQL con las implementaciones Java para construir los artefactos DAO. Esto quiere decir que se pueden ocupar un archivo XML para Oracle o SQLServer sin alterar la programación de las clases Java.
- Disminución del tiempo de desarrollo, gracias a que se reduce la cantidad de líneas de código que debe generar el desarrollador. Este

mecanismo permite a los programadores abstraerse de generar rutinas redundantes en la aplicación.

- Al trabajar con archivos XML que contienen instrucciones SQL, se pueden aprovechar las particularidades que ofrece cada fabricante de bases de datos.

Principales Beneficios del Sistema SRC

Dentro de los beneficios que entrega el sistema SRC, podemos mencionar lo siguiente:

- Consolidar y estandarizar las agrupaciones de renovación de credenciales a demanda de los usuarios.
- Contar con historia de las credenciales que posee cada uno de los dispositivos de red involucrados en el proceso de renovación de credenciales.
- La dirección de red, al contar con un repositorio autónomo, mantiene control y estabilidad del proceso de reconciliación.
- Se disminuyen las tareas de configuración de renovación al momento de integrarse con GUIA, dado que sólo se debe interactuar con una única aplicación a ojos del usuario final.
- La tarea de renovación de credenciales disminuye los actuales tiempos de cambios de credenciales por dispositivo, los cuales son manuales uno a uno. Con el SRC se trabaja con agrupaciones de dispositivos y actualizaciones de estas mismas las que son llevadas a cabo de forma simultánea para cada uno de sus integrantes.
- El proceso de renovación de credenciales a cargo del middleware del SRC, puede ser extensible a otros tipos de dispositivos que no hayan sido considerados hasta ahora.
- El sistema al estar construido en capas, permite actualizar cada uno de sus componentes sin afectar el resto de la arquitectura. Esta característica lo hace flexible y adaptable al momento de exportar esta herramienta a otras divisiones de Telefónica en Latinoamérica.
- Dada la estrategia de programación utilizada, se puede activar o desactivar funcionalidades tales como la ejecución concurrente del proceso de renovación de credenciales sobre las distintas configuraciones establecidas. En este sentido, los administradores de sistema pueden escoger el modo de funcionamiento de este proceso según las prestaciones del hardware que soporta el sistema.

Conclusiones y Resultados

La metodología iterativo-incremental utilizada permitió entregar un sistema completamente funcional. Así, el resultado de la primera fase fue totalmente funcional y operativo en un ambiente de producción, mientras se construía el segundo. En este lapso de tiempo los usuarios auditores pudieron identificar las problemáticas existentes en los actuales procesos de operación con los dispositivos de red, ligados de forma muy estrecha con los términos de seguridad especificados para resolver. Respecto a la arquitectura establecida, esta permitió establecer las piedras angulares para futuras integraciones en cuanto a interoperabilidad, mantenibilidad y portabilidad.

Es importante destacar que la aplicación al ser desplegada en un servidor web permitió al alumno abstraerse de problemas generales de una aplicación standalone, como lo son: concurrencia, transaccionalidad, alta disponibilidad y escalabilidad. Esto permitió que el trabajo se centrara en los aspectos propios del negocio y en las necesidades establecidas en un principio.

Uno de los problemas que surgieron en el trabajo de esta tesis fue la necesidad de contar con un hardware de testeo que permitiera evaluar el proceso de renovación en ubicaciones denominadas como extremas. Desde esta perspectiva sólo se pudo evaluar el espectro considerado a la fecha, quedando pendiente de análisis el comportamiento de la aplicación en dispositivos más alejados y/o especificados en redes más lejanas físicamente.

Al diseñar e implementar esta solución, se obtuvieron los siguientes beneficios para la Dirección de Red:

- Se cuenta con un repositorio centralizado de dispositivos de red además de sus respectivas configuraciones de renovación de credenciales. Esto permitió la independencia de otras fuentes de almacenamiento como lo eran los antiguos archivos Excel controladas por otros departamentos dentro de la organización. De esta forma, se mitigó el riesgo de sufrir accesos no autorizados a los dispositivos de red.

- Al unificar todos los dispositivos de red en un solo repositorio, se mejoró el rendimiento operativo provisto desde las mesas tecnológicas hacia los clientes finales. Vale decir, se disminuyeron los tiempos de procesamiento de la actividad de reconciliación. Esto debido a que ahora no se contemplan tareas adicionales de transformaciones ni adaptaciones de datos. Por otro lado, se redujo el trabajo de configuración, puesto que todos los usuarios contienen datos estandarizados y comunes, independiente de la fuente autoritativa que provengan.
- La implementación del sistema SRC, permite mantener historia de las renovaciones de credenciales realizadas sobre los dispositivos de red, con lo que se da cumplimiento a las normativas vigentes del departamento de seguridad de la información y a la necesidad imperiosa de evitar accesos indebidos sobre los dispositivos de red.

Una de las desventajas de la implementación del SRC es la latencia que bajo ciertas redes se puede generar, esto afecta directamente el tiempo de ejecución del proceso de renovación de credenciales. Es decir, el tiempo de renovación está directamente relacionado con la cantidad de credenciales a renovar por dispositivo de red y los tiempos de latencia existentes en las redes utilizadas para este proceso. Esta es una desventaja que se cree se tendrá pero la cual falta contrarrestar con mayor experimentación.

En cuanto al ámbito de seguridad, hoy se puede decir que se ha cumplido con cerrar la ventana de falla que existía en los inicios del proyecto. Hoy las credenciales de acceso a los dispositivos de red solo son accedidas por quienes han sido autorizados para ello.

El sistema SRC al ser una aplicación Web implementada en capas, puede actualizarse o mejorarse sin afectar otros servicios. Con esto se reducen en un 100% las posibles indisponibilidades de GUIA a causa de alguna actualización sobre el sistema SRC, ya que no afecta al resto de elementos vecinos que componen de GUIA.

La lógica de negocio del SRC al ser implementada como un servicio web sobre un protocolo estándar como lo es SOAP, mejoró su interoperabilidad. Esto significó que no solamente GUIA pueda integrarse a este sistema, sino

que queda la ventana abierta para integrarse con otros tipos de sistema o incluso aplicaciones propietarias que necesiten conectarse y requieran de las virtudes del sistema SRC

Como trabajo futuro, se presenta el desafío de extender el sistema a otras divisiones de Telefónica en Latinoamérica, integrando otros tipos de dispositivos de red. En este caso, se necesitaría extender la funcionalidad de renovación de credenciales en donde se deberían reutilizar la mayoría de las funcionalidades y solo construir los necesarios de acuerdo a los nuevos tipos de dispositivos considerados.

Durante la elaboración de esta tesis, se identificaron oportunidades de incluir nuevas líneas de negocio de Telefónica e incluso otras áreas distintas a la dirección de Red a esta plataforma. Las posibles integraciones a esta plataforma son los por ejemplo el área de TI en primera instancia. Esto les permitiría la renovación automática de credenciales de los dispositivos de red que ellos manejan y con los cuales mantienen un trabajo manual a la fecha hoy.

Dados los beneficios obtenidos por el desarrollo de este proyecto, ya se ha considerado el hecho de exportar la solución a nuevas dependencias de telefónica e inclusive se ha considerado la posibilidad de ofrecer a clientes externos una vez finalizada la completa instalación a nivel nacional.

Glosario

API	Application Programming Interface.
ACS	Access Control Server.
ACU	Aprovisionamiento de Cuentas de Usuarios.
AOP	Aspect-Oriented Programming.
AD	Active Directory.
CPD	Centro de Procesamiento de Datos.
CPE	Customer-Premises Equipment.
DAO	Data Access Object.
DI	Dependency Injection.
DN	Distinguished Name.
DTO	Data Transfer Object.
FAI	Fuente Autoritativa de Identidad.
FUAI	Fuente Única Autoritativa de Identidad.
FTP	File Transfer Protocol.
GUIA	Gestión Unificada e Integrada de acceso.
HGM	Herramienta de gestión y Monitoreo.
IDM	Identity Management.
IBM	International Business Machine.
JEE	Java Enterprise Edition.
JPA	Java Persistence API.
JSR	Java Specification Requests.

LATAM	Latino América.
LDAP	Lightweight Directory Access Protocol.
MTO	Make to Order.
MVC	Model View Controller.
NDU	Normalizador de Datos de Usuario.
OID	Oracle Identity Directory.
OIM	Oracle Identity Manager.
PE	Provider Edge.
RFC	Request For Comments.
SOA	Service-Oriented Architecture.
SOAP	Simple Object Access Protocol.
SQL	Structured Query Language.
SRC	Sistema de Renovación de Credenciales
SSO	Single Sign On.
TACACS	Terminal Access Controller Access Control System.
VLAN	Virtual Local Area Network.
VPN	Virtual Private Network.
XHTML	eXtensible HyperText Markup Language.
XML	eXtensible Markup Language.

Bibliografía

[1] Oracle Corporation, «Oracle Information Driven Support» August 2014. [En línea]. Available: http://www.oracle.com/us/support/library/lifetime-support-middleware-069163.pdf.

[2] Oracle Corporation, «Introducción a Oracle Identity Management» 01 June 2008. [En línea]. Available: http://www.oracle.com/ocom/groups/public/@ocom/documents/webcontent/345082.pdf. [Último acceso: 20 09 2015].

[3] Telefónica Chile, Políticas Normativas de Gestión de Identidad, Santiago: Telefónica Chile, 2013.

[4] I. Sommerville, Software_Engineering, Addison-Wesley, 2010.

[5] A. C. a. J. Penn, Identity Management Market Forecast: 2007 To 2014, Forrester Research.

[6] CISCO, «Cisco Secure Access Control Server 4.2,» 2009. [En línea]. Available: http://www.cisco.com/c/en/us/products/collateral/security/secure-access-control-server-windows/data_sheet_c78-453387.pdf. [Último acceso: 23 08 2014].

[7] CISCO, «An Access Control Protocol (TACACS) » 1993. [En línea]. Available: http://tools.ietf.org/html/rfc1492.

[8] Oracle Corporation, «Gestión de Identidades | Oracle Identity Manager,» 01 February 2013. [En línea]. Available: http://www.oracle.com/es/products/middleware/identity-management/index.html?ssSourceSiteId=null. [Último acceso: 22 09 2015].

[9] Telefónica SA, «Especificación de Requerimientos GUIA» Documento Interno de Telefónica SA, 2013-2014.

[10] Ericsson, Telefónica Chile SA., «RFP- Control de Acceso a Elementos de Red» Documento Interno de Telefónica SA, 2013.

[11] Nokia Siemens, Telephonic Chile SA. «Network Access Management Solution of Nakina Systems» Documento Interno de Telefónica SA, 2013.

[12] Dell, «Administración y Control de Acceso de cuentas privilegiadas - PAM» Documento Interno de Telefónica SA, 2013.

[13] T. C. S. Nokia Siemens Network, «Access Control to Network Elements RFQ» Nokia, 2012.

[14] Nokia, «Comparison: Nokia Access Guard vs. Citrix Desktop Virtualisation» 2014.

[15] D. R. Martinez, Aplicaciones Web Un Enfoque Práctico, RA-MA, Ed., Madrid: España, 2010.

[16] PrimeFaces, «Prime Faces» [En línea]. Available: http://www.primefaces.org/whyprimefaces. [Último acceso: 06 06 2015].

[17] «JavaServer Faces 2.0» - - 2014. [En línea]. Available: https://jcp.org/en/jsr/detail?id=314. [Último acceso: 21 06 2014].

[18] DevRate, « Portal de valoraciones de tecnologías según experiencias de los desarrolladores» [En línea]. Available: http://devrates.com/stats/index. [Último acceso: 27 10 2015].

[19] S. S. y. O. Maassen, Patrones de Diseño Aplicados a Java, Pearson, Ed., Madrid: España, 2003.

[20] MyBatis, «The MyBatis Blog» [En línea]. Available: http://blog.mybatis.org/. [Último acceso: 2014-2015].

[21] J. A. M. S. y. Z. R. R., «Gestión de Identidades y Control de Acceso desde una Perspectiva Organizacional,» 01 Junio 2012. [En línea]. Available: http://web.usbmed.edu.co/usbmed/fing/v3n1/v3n1a3.pdf. [Último acceso: 21 09 2014].

[22] A. Kumar, Oracle Identity and Access Manager 11g for Administrators, 1st ed ed., -: Birmingham, 2011.

[23] D. Datta, Oracle Corporation, «Administrator's Guide for Oracle Identity Manager 11gR2» 01 August 2011. [En línea]. Available: http://docs.oracle.com/cd/E25054_01/doc.1111/e14316.pdf. [Último acceso: 21 08 2014].

[24] G. Inc, Magic Quadrant for User Administration and Provisioning, 2012.

[25] Oracle Web Site, «Gestion de Identidades | Oracle Identity Manager» 2 Febrero 2013. [En línea]. Available:

http://www.oracle.com/es/products/middleware/identity-management/index.html?ssSourceSiteId=null. [Último acceso: 16 Abril 2014].

[26] Oracle Corporation, «Introducción a Oracle Identity Management» Junio 2008. [En línea]. Available: http://www.oracle.com/ocom/groups/public/@ocom/documents/webcontent/345082.pdf. [Último acceso: 16 Abril 2014].

[27] Ericsson Corporation, «Web-based configuration management architecture for router networks» 2000. [En línea]. Available: ieeexplore.ieee.org/document/830383/ [Último acceso: 16 Abril 2014].

Anexo

ANEXO A: PATRONES DE DISEÑO [19]

Un patrón de diseño, genéricamente, describe un problema y la solución que se le da al mismo, de modo que si vuelve a suceder el mismo problema se pueda aplicar una solución similar. En el contexto de la ingeniería de software, permiten organizar los objetos según estructuras comunes y probadas, ganando en flexibilidad, reutilización y calidad del software.

Los patrones responden a problemas de diseño de aplicaciones en el marco de la programación orientada a objetos. Se trata de soluciones conocidas y probadas cuyo diseño proviene de la experiencia de los programadores. De esta forma, los patrones de diseño están basados en las buenas prácticas de la programación orientada a objetos.

A continuación se describen los patrones de diseño utilizados en este trabajo de tesis.

A.1 Proxy

Su propósito es proporcionar un representante de otro objeto, por distintas razones como pueden ser el acceso, la velocidad o la seguridad entre otras. Se utiliza cuando se necesite una referencia más elaborada a un objeto que una referencia simple.

Para permitir que el proxy represente al objeto real, el proxy tiene que implementar exactamente la misma interfaz del objeto real. Además, el objeto proxy tiene una referencia del objeto real, esto es necesario para poder llamar a los métodos del objeto real en caso de que sea necesario. De esta manera, los clientes interactúan con el proxy, pero éste puede delegar su ejecución en el objeto real. El proxy implementa la misma interfaz que el objeto real, pero puede ejecutar tareas adicionales que no puede realizar el objeto real, como son la comunicación remota o seguridad.

Se utiliza el patrón proxy en las siguientes situaciones:

1. Cuando se necesite un representante local para un objeto en otro contexto o remoto (otra máquina virtual de java)
2. Cuando se necesite un cache de objetos de costosa instanciación y de frecuente utilización.
3. Cuando se desee establecer políticas de seguridad de creación para el objeto real.

A.2 Factory

Este patrón de diseño también es conocido como "Virtual Builder", puesto que define un método estándar y abstracto para generar objetos, delegando en las subclases su creación efectiva.

Este patrón se denomina Factory porque crea objetos cuando se necesitan. Cuando se empieza a escribir una aplicación, a menudo está claro qué tipo de componentes se utilizarán. Normalmente se tiene una idea general de las operaciones que deben tener ciertos componentes, pero la implementación se realiza en otro momento, por lo que pueden surgir situaciones que no se tuvieron en cuenta.

Esta flexibilidad puede ser alcanzada utilizando interfaces para estos componentes. Pero el problema de programar interfaces es que no se puede crear un objeto a partir de la interfaz. Se necesita una clase que las implemente ara obtener el objeto.

Se puede utilizar este patrón en los siguientes casos:

1. Se desee crear un framework extensible. Esto significa proporcionar flexibilidad delegando las decisiones, como el tipo específico del objeto a crear para un momento posterior.
2. Cuando se sabe cuándo crear un objeto, pero no se conoce el tipo de objeto.

A.3 Fachada

Este patrón tiene como objetivo proporcionar una interfaz simplificada para un grupo de subsistemas de compleja gestión. En otras palabras, agrupa las

funcionalidades de un conjunto de servicios en una interfaz unificada, siendo fácil de usar por parte de los componentes que lo invoquen.

Normalmente, el patrón fachada delegará la mayoría del trabajo en los subsistemas, aunque muchas veces también puede cumplir con alguna función. Como la de ser coordinador de flujo entre los resultados al invocar de manera ordenada a estos sistemas.

Se debe destacar que la intención del patrón fachada no es esconder los subsistemas. Su misión es proporcionar una interfaz más simple para un conjunto de subsistemas, permitiendo que los clientes más avanzados puedan utilizar las opciones más elaboradas y trabajen directamente con los subsistemas.

Se puede utilizar este patrón para:

1. Simplificar el uso de los sistemas complejos proporcionando una interfaz más sencilla sin eliminar las opciones avanzadas.
2. Reducir el acoplamiento entre los clientes y los subsistemas.
3. Introducir capas para grupos de subsistemas, lo que proporciona un alto grado de mantenibilidad del servicio que proporcionan estos subsistemas.

A.4 Singleton

El patrón singleton tiene como objetivo asegurar que una clase sólo posee una instancia y proporcionar un método de clase único que devuelva esta instancia. A la vez permite que todas clases tengan acceso sólo a esa instancia.

Este patrón se utiliza cuando.

1. Se necesita un objeto global, uno que sea accesible desde cualquier parte del sistema, pero que sólo deba ser creado una vez.
2. Cuando no se quiera pasar la referencia de este tipo de objeto a los otros objetos de la aplicación.

3. Cuando se desee dejar de utilizar las variables globales. Dado que sobre las últimas no se tiene control de quién puede acceder a una instancia estática públicamente disponible.

A.5 Front Controller

Patrón de diseño de la capa vista que centraliza el control de las peticiones de los clientes en un sólo punto. Adicionalmente en este punto, se pueden gestionar la seguridad y control de errores. De esta manera, se reduce la cantidad de código embebido en la vista, puesto que se disminuye la lógica de control generado en las vistas.

Este patrón se utiliza cuando:

1. Se desea evitar lógica de control duplicado.
2. Agrupamiento de lógica de control común a múltiples aplicaciones.
3. Separación total de la lógica de control de las vistas.
4. Centralización del acceso a la aplicación en un único punto.

A.6 Double Dispatch

Es un patrón de diseño también conocido como Visitor, su función es proporcionar una forma fácil y sostenible de ejecutar acciones en una familia de clases. Este patrón centraliza los comportamientos y permite que sean modificados o ampliados sin cambiar las clases donde se actúa.

La forma de trabajar del patrón visitor consiste en extraer unas operaciones relacionadas de un grupo de clases y situarlas juntas en una única clase. El motivo principal es la facilidad de mantenimiento del código. En ciertas situaciones, simplemente resulta complicado mantener todas las operaciones en las propias clases. Este patrón es útil para esas situaciones, ya que proporciona un marco genérico para soportar las operaciones sobre un grupo de clases.

Este patrón se aplica cuando:

1. Un sistema contiene un grupo de clases relacionadas.

2. Se tiene que realizar algunas operaciones no triviales sobre algunas o todas las clases relacionadas.
3. Las operaciones deben ejecutarse diferente para las distintas clases.

A.7 Worker

Es un patrón de diseño cuyo propósito es mejorar la productividad y eficiencia de un proceso que implique la ejecución de tareas repetitivas secuencialmente. De esta manera, es utilizado para separar trabajos automáticos de aplicación mediante el multi threading. El modo de funcionamiento de este patrón es el siguiente: un thread worker toma una tarea de una cola y la ejecuta. Cuando finaliza, pasa a la siguiente tarea de la cola.

Con este patrón es sencillo dividir una aplicación en tareas porque solamente se especifica que hay que hacer algo, pero no se concreta cuando. De esta manera el worker ejecuta muchas tareas independientes una tras otra. En vez de crear un nuevo worker cuando hay que llevar a cabo una tarea, se le asigna la tarea a un worker existente.

Este patrón se utiliza cuando:

1. Se quiera mejorar la productividad.
2. Quiera introducir concurrencia.

A.8 Dependency Injection

La Inyección de Dependencias es un patrón de diseño orientado a objetos, en el que se suministran objetos a una clase en lugar de ser la propia clase quien cree el objeto. En una aplicación típica, suelen existir varias clases que conjuntamente realizan algunas tareas y, por lo tanto, se encuentran interrelacionadas de algún modo. En lugar de definir las relaciones en el código de la aplicación, estas dependencias suelen especificarse en archivos XML de configuración para que sea el contenedor el responsable de inyectar las relaciones cuando se creen las entidades llamadas beans.

A.9 DTO

Es un patrón de diseño de la capa lógica cuyo propósito es encapsular y transportar los datos de una entidad por las capas de una aplicación. De este modo, durante el intercambio de información entre los módulos que forman parte del sistema, se evitará el paso de muchos argumentos en los métodos de los objetos. Esto tiene como ventaja que si se produce un cambio en la lista de atributos de dicho objeto, solo cambiamos la definición del DTO y no todas las referencias en los métodos donde participe. Los DTO deben ser objetos serializables para facilitar su transporte entre contextos, deben ser simples y no deben proveer información de la lógica del negocio. Esto quiere decir que su única responsabilidad es almacenar y entregar los datos contenidos en sus atributos.

A.10 DAO

Es un patrón de la capa de persistencia que separa la lógica de la aplicación del acceso a los datos, independizando de esta forma la aplicación de la fuente de datos utilizada. Para esto, se suministra una interfaz común entre aplicación y uno o más medios de almacenamiento. Las ventajas de este patrón son las siguientes:

1. Los objetos de negocios no requieren conocimiento del destino final de la información que gestiona.
2. Centraliza el acceso a datos, escondiendo a los clientes los detalles completos de las fuentes de datos.
3. Facilita la mantenibilidad, puesto que si se produce un cambio produce un impacto mínimo en el resto de la aplicación.
4. Reduce la complejidad de la implementación del acceso a los datos en la lógica de la aplicación

A.11 MVC

El propósito de este patrón es dividir un componente o un sistema en tres partes lógicas “Modelo, Vista y Controlador”, facilitando la modificación o personalización de cada parte. Este patrón responde a la necesidad de

controlar la complejidad y prevenir los efectos de los cambios mediante la división de las siguientes tres partes funcionales:

1. Modelo: se trata del núcleo funcional que gestiona los datos manipulados en la aplicación.
2. Vista: se trata de los componentes destinados a representar la información al usuario. Cada vista está vinculada con un modelo. Un modelo puede estar vinculado a varias vistas.
3. Controlador: un componente de tipo controlador recibe los eventos que provienen del usuario y los traduce en consultas para el modelo o para la vista. Cada vista está asociada a un controlador.

En este patrón la información es generada por un modelo y será el controlador el que cambiará el aspecto de la interfaz de usuario o vista en función de las reglas de negocio y la información generada por el modelo.

Este patrón se utiliza en las situaciones que se enumeran a continuación:

1. Cuando hay que crear componentes que sean flexibles y fáciles de mantener. Normalmente se utiliza para los casos en que se esperan cambios en los componentes y también se esperan que sean reutilizados.
2. Para aplicaciones flexibles e interactivas que distribuyen las funcionalidades de dicha aplicación entre los distintos objetos que la componen, de manera que el grado de acoplamiento entre estos objetos sea mínimo.